Implementing an IDaaS for Azure Active Directory using Azure Conditional Access Policies

MSc Research Project
Cybersecurity

Sujit Mourya
Student ID: x19239343

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Sujit Mourya

Student ID: x19239343

Programme: MSc in Cybersecurity **Year:** 2021-2022

Module: Industry Internship

Supervisor: Prof. Vikas Sahni

Submission Due Date: 07/01/2022

Project Title: Implementing an IDaaS for Azure Active Directory using Azure Conditional Access Policies

Word Count: 5524 **Page Count**: 17

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Sujit Mourya

Date: 06/01/2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on the computer.	□

Assignments submitted to the Programme Coordinator Office must be placed into the assignment box located outside the Office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Implementing an IDaaS for Azure Active Directory using Azure Conditional Access Policies

Sujit Mourya
19239343

Abstract

Microsoft Azure Active Directory is a widely popular identity and access management tool based on a cloud solution that contains various services such as application access management, directory services, and advanced identity protection. The primary objective of this research paper is to use the SensiPass existing service and further develop and extend its existing authentication mechanism. Azure Active Directory integrates with SensiPass's core Identity providing services, and the conditional access of Azure AD contains a security feature for enabling a custom multi-factor authentication. The conditional access feature has enabled SensiPass® for using its existing authentication mechanism. Azure AD is an industry-standard and market leader for IAM services with advanced security and administrative features. SensiPass's founder initialized the requirement to enhance its authentication for IDaaS compatibility. For the research study, extensive research was conducted and evaluation of the existing literature on 3-factor authentication, IAM and AD. The thesis has been written based on the incremental and constructive approach since it is based on developing and testing new features. The evaluation compares cloud service providers and performs further testing, such as unit testing and integration testing. Different test cases have been chosen carefully to cover every edge case and functionality testing. The evaluation output resulted in the successful completion of the integration, and it enabled SensiPass® to lay the foundation of the IDaaS business model.

1 Introduction

Cloud computing has evolved from grid computing to distributed computation, evolutionary technology. There are pools of computers having shared networks, storage resources, applications, and infrastructure within cloud computing technology. At the same time, the essential characteristics of cloud computing contain rapid elasticity, cost management, on-demand provisioning, and pervasive network access (Almorsy *et al.*, 2016). The features provide the benefits through enhanced cost savings, scalability, and ease of usage. While the features add value for business users, security remains the highest priority for such businesses. Identity and Access Management is considered a critical priority in organizations security audits and standardization policies. National Institute of Standards and Technology (NIST) and Cloud Security Alliance (CSA) considers IAM a critical research area and an essential issue for any cloud-based applications or services [1].

Web Engineering discipline defines developing Web-based systems and applications (Murugesan *et al.*, 2001). Every company develop their business applications that require

[1] https://www.nist.gov/identity-access-management

different access control to their services and identity management. Developing in-house solutions sometimes adds overhead costs to these companies, and even more, problems occur when multiple organizations require access to the same applications and services. Such problems can be addressed by building a standardized solution that allows the interoperability of systems in a security context. The core idea of such solutions revolves around using Web service technology by separating authentication and authorization technology from the applications themselves.

The Security Assertion Markup Language (SAML) is considered a secure and standard XML-based protocol for the security exchange and was chosen for the solution. The Azure Active Directory supports SAML based open standard for easy integration with different applications, allowing users to sign in through external federated identity managed by the respective organizations, while Azure AD manages the access to the services. (Yang *et al.*, 2014)

Identity and Access Management is at the core of every app and service in the cybersecurity domain. The current literature is mainly based on the usage of different factors through different technologies, but due to AI, there is a possibility of impersonating the existing technologies in the IAM space. Hence the 3-factor authentication at SensiPass® can mitigate the different attacks related to username and password hack and identity manipulations.

Azure offers a wide range of services, including security, virtual networking, communication mechanisms, and caching tactics, in addition to computation and storage[2]. Azure AD is primarily used as a cloud service providing web application authentication, single sign-on, and user management. Users for Azure Active Directory can originate from a range of locations[3]. The first approach is Azure AD-based users, which manually creates users in the directory. The second option is to use a tool known as Azure AD Connect to synchronize user profiles from on-premises AD or Windows Server AD.

1.1 Motivation and background

"SensiPass® creates a sophisticated digital signature by empowering the user to create a secret interaction they can use to digitally modify their biometric signature, making it impossible for others to steal and imitate." (Mike Hill, 2021). The proposed solution for SensiPass® as Identity-as-a-Service using Azure Active Directory has been proposed due to the requirement that SensiPass set, and there was a need for such solutions as the research gap analysis showed the Azure Active Directory is a market leader in Identity and Access Management. Furthermore, SensiPass® has been the ever-growing IDaaS provider that needed the solution for continuing its foothold as an Identity Service Provider. There is hardly any business providing a 3-factor based authentication service as an IDaaS to their clients, and even the Azure AD integration seems to be an extra feature that would help clients manage their access and permissions of users. Whereas SensiPass® will provide identity management, a robust and secure form of authentication management. The research paper will open a door for different possibilities around IDaaS at SensiPass®, and hopefully, it will have a value proposition to different stakeholders.

[2] https://azure.microsoft.com/en-in/services/active-directory/#overview

[3] https://azure.microsoft.com/en-in/services/active-directory/#overview

2 Related Work

In recent years, the three-factor authentication schemes and protocols have been enhanced to use a combination of password, smartcard, and biometrics, which provides a very high level of security compared to a traditional two-factor authentication that depends upon a password and a security token. A considerable amount of literature exists on three-factor authentication for providing secure authentication with various use cases. The following literature showcases the different schemes for securing IoT-based Networks, Wireless Sensors Networks, and others.

2.1 Three-factor authentication based on IoT-based Networks

In their research study, (Yu, Park and Park, 2019) proposed a scheme that can withstand various attacks such as session key disclosure, impersonation, replay attacks, mutual authentications, and anonymity. To address security issues in cloud computing environments, they presented a secure and lightweight three-factor authentication approach for IoT. To establish secure mutual authentication and Burrows-Abadi-Needham logic analysis, they used secret parameters and biometrics. It was interesting to observe how automated validation of internet security protocols and AVISPA simulation tools were used to combat replay and man-in-the-middle threats. Unfortunately, Yu et al.'s scheme are prone to insider attack because the random nonce for the genuine user can be easily manipulated from the database, and an insider person can easily access and modify it, causing an insider attack. Also, a malicious adversary may obtain a smart card and extract the information stored in the smart card during the user's registration process.

A three-factor mutual authentication system for a multi-gateway IoT environment was proposed in another study at (Lee *et al.*, 2019). Gateway spoofing, session key leakage, offline password guessing, and impersonation attacks were all expected to be addressed in the paper. They proposed their approach to establish secure mutual authentication using BAN logic and AVISPA for automated formal security verification using AVISPA. Unfortunately, it was vulnerable to ephemeral secret leaking, DoS, and privileged insider attacks.

In 2017, (Bae and Kwak, 2020) introduced a multi-factor authentication technique based on a smartcard in a multi-gateway IoT environment that was efficient and trustworthy. The architecture was created to reduce computational and communication costs, but it was shown to be vulnerable to traceability, spoofing, impersonation, and anonymity attacks. This made the scheme insecure for mutual authentication and session key attacks.

For IoT devices, in their work (Alshahrani and Traore, 2019) the authors suggested a lightweight and reliable mutual authentication. To verify the sender's identity, the authors utilize a cumulative key hash chain. Burrows-Abadi-Needham logic is used for validation, and the protocol validates internet security protocols and applications automatically.

(Masud *et al.*, 2021) in 2021 tackled many security flaws in terms of user authentication. They developed a lightweight and anonymity-preserving user authentication to counter DoS, man-in-the-middle attacks, and IoT-related privacy attacks. It provided secure user session management and prevented unauthorized access to IoT sensor nodes. To decrease the node's less computing usage, it contained a small footprint of hash cryptography algorithm, which made it efficient and less costly in terms of computing and communication compared to other protocols.

(Garg *et al.*, 2020) solved the sensitivity related to insecure communication and data exchange developed a scheme using blockchain-enabled authentication key management protocol and elliptic curve called BAKMP-IoMT. The proposed solution included the trusted authority for identity management as an extra layer for secure authentication between two parties for the communication node. Unfortunately, the scheme lacked safety against DoS attacks and communication delays due to high storage costs and heavy computing requirements.

2.2 Three-factor authentication based on Wireless Sensors Networks

A study conducted in 2017 (Jiang *et al.*, 2017) proposed an updated authentication scheme for wireless sensor networks, taken from (Amin *et al.*, 2016). The authors proposed a three-factor mutual authentication protocol for wireless sensor networks. However, it was vulnerable to offline guessing attacks and tracking attacks. Jiang *et al.* improvised the scheme using the Rabin cryptosystem, but it had a relatively high computation cost. Moreover, the authors conducted a formal verification for their proposed protocol using ProVerif for showcasing the fulfilment of the necessary security properties. The protocol was shown to be secure against all types of damaging attacks, including session key disclosure and traceability attacks, after a thorough heuristic security analysis.

(Yu and Park, 2020) in 2020, introduced SLUA-WSN a lightweight three-factor authentication approach that includes a secure user authentication system. It is supposed to be the best in efficiency and outperformed all previous state-of-the-art techniques in mitigating the attacks related to sensor node capture, insider attack, impersonation attack, un-traceability, and replay attack. Although one of its shortcomings was weakness, a shared secret key could have been easily guessed since the parameters stored in the smartcard were easily retrievable. Also, the generated random number did not have a proper validity check during the initial session between GE and FN.

A three-factor multi-gateway WSN-based user authentication technique was also introduced in 2017 (Wu et al., 2017). According to Wu et al., generic WSNs may provide a large overhead to the gateway, necessitating the deployment of several gateways for WSN. They also showed that the protocol they proposed was resistant to a range of cryptographic attacks, including impersonation and sensor capture attacks. Also, (Saqib, Jasra, and Moon, 2021) discovered that it is vulnerable to user tracking attacks and that the session differed amongst participants.

2.3 Three-factor authentication based on different technologies

In 2009, (Fan and Lin, 2009) proposed a more effective three-factor authentication protocol by devising privacy protection on biometrics. The user first selects a random string and encrypts their biometric template while registering as per their protocol. The outcome is saved on the smart card as a sketch. While the authentication is in process, the user must conform to a server that he or she can decrypt the sketch, which requires accurate biometrics. While the scheme provides privacy-preserving and enhances the three-factor authentication scheme, it has been seen that it lacks support for a contact-less change of a password.

The authors in the paper (Challa *et al.*, 2018) also proposed a three-factor authentication scheme along with the key agreement protocol. The key agreement protocol was based on the elliptic curve function and supported biometrics update and password features, dynamic sensor node addition, and smartcard revocation. The security analysis was performed under ROR (Real-or-Random) model and BAN logic. Also, the simulation was done through a widely accepted AVISPA simulation tool. The critical review on Challa et al. later found out that the scheme was prone to session key leak attacks and sensor node capture attacks because the session key lacked computational complexity.

For smart grid connectivity, in 2019 (Khan, Kumar and Ahmad, 2019), worked on a scheme based on biometric elliptical curve cryptography and ECC-based mutual authentication. Their design covered replay attacks, session key management, non-transferability, non-traceability, and impersonation attacks. Furthermore, the proposed scheme also reduced transmission and computation costs more than other innovative grid protocols.

(Roy *et al.*, 2018) proposed a three-factor user authentication scheme based on the Chebyshev chaotic map and the cryptographic has function. The symmetric key encryption and decryption were used for three-factor, and the lightweight and efficiency was achieved due to the fact by avoiding elliptic functions, heavy computation, or modular exponentiation. Nevertheless, it was observed that their scheme was prone to offline password guessing attacks as explicit password verifiers failed during leakage of biometric and smart card data.

There has been a more significant enhancement in modern security perimeter, extending way beyond the organization's network. It can include the user and its device identity as a part of identity-driven signals, which has become a part of the organization's access control decisions. Conditional access could bring all the signals together and decide and impose organizational policies. Azure AD Conditional Access is central to the novel identity-control plane. Conditional Access policies can be applied for the proper access controls to secure the organization. [4] Conditional Access policies are in more straightforward terms referred to as if-then statements. Suppose a user wants to have access to any service, they are required to complete an action. For example, a financial transaction involving sensitive data exchange requires a user to do a biometric-based authentication to complete the transaction.[5]

3 Research Methodology

The research methodology section explains the different procedures performed for implementing the design. The research study has been conducted based on the technology decision at SensiPass® and based on the comparison of different cloud IAM providers as the current market trend recommends using Azure Active Directory and the requirement at SensiPass® needed a solution for Azure Active directory. The research gap analysis was also conducted to find the current market leaders in the cloud-based IAM solution domain. Furthermore, the side-to-side comparison with the different cloud-based IAM providers showed the features and various IAM technologies and their strengths and weaknesses. The implementation primarily follows three steps, first, evaluating the current technology at SensiPass® and Microsoft Azure AD technology. Second, development and deployment of the changed codebase and API. Third, the testing and evaluation of the implemented system.

3.1 Current technologies evaluation at SensiPass® and Microsoft Azure AD technology

It is crucial to evaluate the technologies to be implemented and the infrastructure that must be researched. Microsoft Azure is a market leader in identity and access management through its single sign-on, User provisioning and Azure Active Directory features. The research study has evaluated the IDaaS solution at SensiPass® with Azure Active Directory integration. The evaluation process revolved around the test cases carried out in isolation and different modules as integration testing.

3.2 Implementation and Configuration of the Architecture

[4] https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/overview

[5] https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/overview

Research implementation has been completed based on Microsoft Active Directory services available with Microsoft Azure Cloud. The integration documentation provided by Azure for integrating SAML application in Azure AD provided enough references for understanding Single-Sign-On, SAML protocol, Federated Identity Provider, and X509 Certificates. The Azure Active Directory dashboard contains various roles and administration settings, Enterprise applications registrations, and security. Conditional access was a primary requirement for implementing 3-factor authentication utilizing SensiPass® technology at its core. The current architecture at SensiPass® requires a modification of their API endpoints and the creation of new endpoints that would be used to perform a SAML based authentication and requests from Azure AD.

3.3 Evaluation

Implementing the Azure AD with SensiPass® required thorough testing, dependent on different types of test cases such as Unit and Integration test cases. Also, the testing was performed at the API endpoints and checking user sessions manually after successful authentication. The user registration and authentication were checked along with validations of each mandatory field. Finally, the SensiPass® app was tested with its current technology to test the 3-factor authentication triggered by conditional access on Azure AD.

4 Design Specification

The Design Specification focuses on describing the different modules that support the integration of SensiPass and Azure AD.

4.1 Application User

The Application User is the one who needs access to the application hosted on Azure AD. The users would need the SensiPass application installed on their android mobile to perform 3-factor authentication as multi-factor conditional access triggered on Azure AD. The user should also be registered on Azure AD and belong to the required group to whom the multi-factor access is to be checked.

4.2 Azure AD Subscription and Conditional Access

The Azure AD tenancy was simple to set up and was completely free too. Anyone who has a Microsoft account can easily set up their tenant. Following the formation of the tenant, the application registration required a premium Azure AD membership. Premium P1 subscription prices are $6 per user per month with a yearly agreement, and P2 subscription prices are $9. (Microsoft: Azure AD Pricing 2021). The P2 membership includes similar features as the P1 subscription. However, it also includes additional identity protection and management capabilities. Fortunately, it is possible to access premium services without a paid subscription, even a free tier user account.

Azure AD Conditional Access is used to create policies that analyze Azure Active Directory user access requests to apps and provide access only when the request meets specific criteria, such as user group membership, access device geolocation, or successful multi-factor authentication.

4.3 SensiPass API Service (SPM)

The SensiPass API Service is the endpoint created on the existing SensiPass core architecture to extend the functionality for enabling the SAML based notification service when the conditional access gets triggered from the Azure AD for multi-factor authentication.

4.4 SensiPass Android Application

The user who requires application access needs to have a SensiPass android application for authenticating them through multi-factor authentication[6]. The current application of SensiPass supports only the Android platform, and hence only android users will be able to validate their identity.

5 Implementation

This section of the research paper shows the implementation of the Azure AD and SensiPass core authentication API service. The changes to the core services were first performed on the local environment and eventually uploaded to the staging environment. The Azure AD trial subscription was used to configure the security parameters for multi-factor authentication. The trial subscription came with full access to the Azure AD conditional access.

[6] https://play.google.com/store/apps/details?id=ie.cit.nimbus.sensipass.sensipassandroid198eval&hl=en_US&gl=US

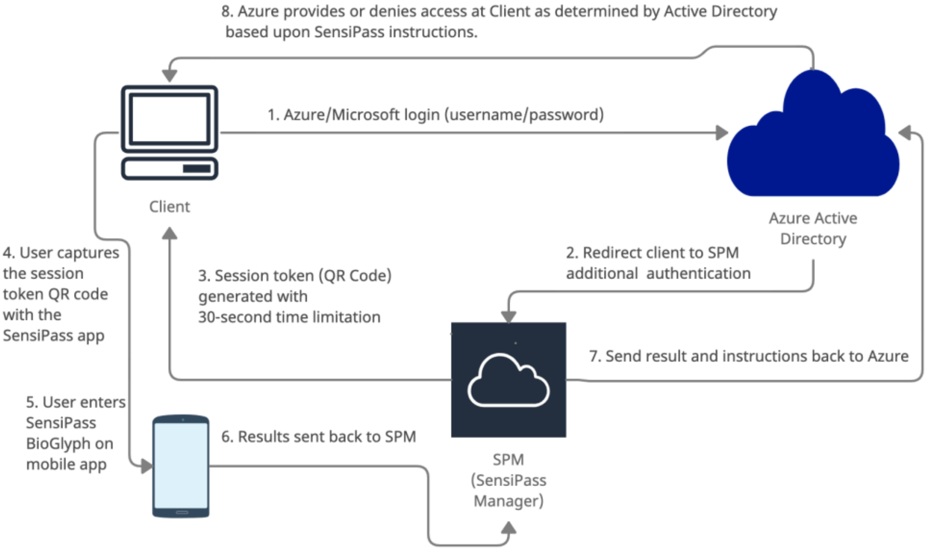

8. Azure provides or denies access at Client as determined by Active Directory based upon SensiPass instructions.
1. Azure/Microsoft login (username/password)
Client
Azure Active Directory
4. User captures the session token QR code with the SensiPass app
3. Session token (QR Code) generated with 30-second time limitation
2. Redirect client to SPM additional authentication
7. Send result and instructions back to Azure
5. User enters SensiPass BioGlyph on mobile app
6. Results sent back to SPM
SPM (SensiPass Manager)

Figure 1. Proposed Architecture diagram for SensiPass® and Azure AD

Figure 1. shows the proposed architecture diagram from SensiPass to Azure AD and a user's login process flow. The Azure user login through the existing Azure/Microsoft credentials for accessing the application on Azure. The Azure AD checks for the existing session. If the user is already logged in, they will access the application or be redirected to authenticate in the Azure AD login page. Once the user is login to Azure AD and a session is created, the conditional access checks if multi-factor authentication is required for the user. If the user requires multi-factor authentication, the user is taken to the SensiPass page, where the user needs to scan the QR code to trigger the 3-factor authentication on the SensiPass app. The user opens the front camera in the SensiPass android application, and the request goes to the API, which identifies the user based on the biometric, tokens and knowledge factor and fuses all three into a single, highly dynamic digital signature (Hill, Ruddy and SIROTA, 2012). The result is then processed for the verification of the user, and the response is sent back to Azure for confirming the final authentication on Azure AD. On The final step, the user is taken to the application landing page.

The implementation primarily took place in two steps. First, changes were done on the SensiPass authentication service. Second, the configuration changes and settings at the Azure AD end. For this part of the research paper implementation, the focus was only on the configuration at the Azure AD end. The following steps show the configuration and creation of different pre-requisites for integrating Azure AD and SensiPass core services.

5.1 Creation of a Tenant

The first step required for implementing Azure AD is the creation of a Tenant. Tenant represents the organization created in Microsoft Azure AD. Azure Active Directory's primary responsibility is to organize all the applications and users into a single domain called a group. These groups are also known as tenants—the tenant act as a dedicated instance for app developers in the Azure AD environment. The relationship between different Microsoft cloud services is generated through tenants. Tenants id is used as sign-in credentials to external services and Azure[7].

The tenant creation page takes different steps and starts with the basic steps asking for the tenant's name. The tenant name should be unique as it creates an organization subdomain on Microsoft's primary domain. The next step reviews the selection of server country location, and finally, it takes a while to create an AD space on the Azure cloud. A unique tenant id is generated after the completion of tenant creation.

5.2 Creation of an Active Directory User

The creation of an AD user contains two options. First, the create user form contains a different field requiring identity, password, groups, roles, settings, and job info. In contrast, the Invite user option allows users to collaborate with the organization in AD. The invite link gets through the email, and the user can accept the invitation to start the collaboration within the organization.

5.3 Creation of an Active Directory User Group

This step deals with creating the user group when the multi-factor authentication is exposed for groups. The pre-requisite to conditional access is group creation. The groups can be targeted instead of every user asked for multi-factor authentication. The group contains two types, Security and Microsoft 365 groups. Basic information such as group names and descriptions are required for creating a group.

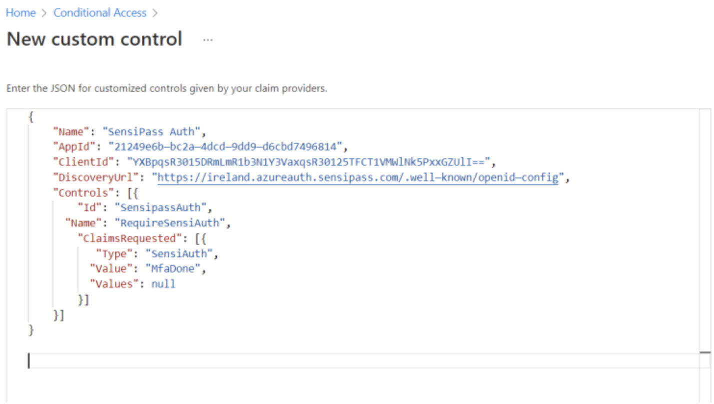
5.4 Creation of conditional access policies

This phase requires the creation of a new policy for conditional access. There are two methods of creating a conditional access policy. First, the policy is created from scratch and second, the policy is created from templates. The user can be targeted based on the device specifications and parameters, sign risk and user risks from a 'what-if' policies. The conditional access policy creates signals together, enforces organizational policies, and makes decisions based on the conditions. [8] The preview templates contain the predefined commonly used policies across different locations and types of customers. Different combinations of users, apps, conditions, sessions, and access controls can create a conditional access policy.

[7] https://www.educba.com/azure-tenant/

[8] https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/concept-conditional-access-policies

5.5 Creation of Custom Controls in Conditional Access



Enter the JSON for customized controls given by your claim providers.

```
{
    "Name": "SensiPass Auth",
    "AppId": "21249e6b-bc2a-4dcd-9dd9-d6cbd7496814",
    "ClientId": "YXBpqsR3015DRmLmR1b3N1Y3VaxqsR30125TFCT1VMWlNk5PxxGZUlI==",
    "DiscoveryUrl": "https://ireland.azureauth.sensipass.com/.well-known/openid-config",
    "Controls": [{
        "Id": "SensipassAuth",
      "Name": "RequireSensiAuth",
        "ClaimsRequested": [{
           "Type": "SensiAuth",
          "Value": "MfaDone",
          "Values": null
        }]
    }]
}
```

Figure. 2 Custom Control Configuration in Azure AD Conditional Access

The custom control configuration shown in Figure. 4 shows the JSON data that contains the configuration details for the SensiPass® Service endpoint. The DiscoveryUrl contains the SensiPass® service endpoint URL triggered when a condition is met in Azure AD. Name, AppId, ClientId contains the details specific to the application, and it should be unique for each Azure AD implementation. The "Controls" contains the ID and Name that will be sent to SensiPass® SAML based service and verified once the request is received. The ClaimsRequested contains the Type and Value for every request. There can be multiple values in ClaimsRequested based on different implementations, as it is an array-based key.

6 Evaluation

This section of the research paper focuses on evaluating the implementation that was performed in the previous section. The evaluation is done through the testing of each test cases and the test output is used to evaluate the outcome and the success of the implementation. The integration of the two systems is also tested separately with the end-to-end functionality.

The current analysis and evaluation were performed to understand the findings and results of each test case, comparing against the cybersecurity framework like OWASP. The main goal of the evaluation is also to find out the integration testing and unit test cases written for each module independently and altogether. The evaluation methodology revolves around the test cases as the implementation of the research is based on apps, software programming and APIs.

6.1 Side by side comparison analysis of cloud identity providers

Cloud identities require cloud-based IDaaS and IAM solutions as adoption for becoming a logical step for its management. Cloud service providers offer different services for managing the identities of administrators and IAM services for managing end users' identities. Table 1 explains significant cloud service providers and the third parties offering IAM services. In the evaluation of different cloud service providers, the Microsoft active directory was our focus point as the proposed solution required the usage of Azure AD and its features such as SSO compatibility, Multifactor authentication, reporting, and monitoring helps to provide a featured-rich option for connecting any external app for federated login through external IdP through Azure AD conditional access.

Table 1: Cloud identity service provider and customer IAM systems

Providers	Overview	Features	Delivery
MS Azure Active Directory	Connects with Active Directory (On-premises)	SSO compatibility, MFA, Identity governance, Identity protection, monitoring and reporting, RBAC	Cloud
IBM Security Identity and Access Assurance	Provides identity and access management governance	RBAC, SSO, MFA, Risk-based authentication, Identity governance and administration,	On-prem, cloud
Oracle Identity Cloud Management	Offers identity and access management (IAM) for employees, partners, and customers in hybrid environments.	MFA, SSO, RBAC, Delegated Authentication, Duo authentication, FIDO security[9]	Cloud
Okta	Identity and access management, Mobility management for users, partners, and firms.	Provisioning, SSO, Active Directory, LDAP integration, MFA, Mobile identity management, platform-independent	Cloud and on-prem
ForgeRock	Integrates identity and access management across	SSO, monitoring, provisioning, reporting	Cloud on-prem

[9] https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/oracle-identity-cloud-service.html#GUID-5300093E-25D9-40E4-BF4B-50A65A3AC055

	cloud, on-prem and mobile.		

6.2 Test Scenarios Execution

The test scenarios are shown below with the actual results and the pass and fail statuses. The test cases have been chosen with all the edge cases in mind and the unit functionality testing and integration testing.

Table 2: Actual results and pass or fail statuses of the test scenarios

Test Scenario	Actual Result	Module	Pass/Fail
User Registration	User registration was successful.	Azure	Pass
Checking user session after login	The user session was created in the browser after the login.	Azure	Pass
Deleting user sessions and checking access	The user session was created.	Azure	Pass
User details entry at Azure AD dashboard	User entry was successfully created on the app.	Azure	Pass
Conditional access check	The conditional access was triggered based on the user groups and the severity score	Azure	Pass

6.3 Discussion

The literature review studied so far have many significant flaws. Firstly, the existing related works focused on improvement over fewer security features, and it only met the partial goals of developing an IAM System. Secondly, there has been little to no contribution in the IAM domain for enhancing the usage of biometric-based 3-factor authentication that can be considered a hundred per cent secure. AI technology and weaponization are becoming common in today's cyberattacks. The requirement of full-proof IAM identity management is a need of an hour. Thirdly, the IAM systems have been coupled with different mandatory features and needed the businesses to use their in-house built solutions for managing their identity access and management services, where there was a need for customized solutions. Microsoft Azure active directory is considered one of the best class Access management services integrated with the SensiPass® built identity management services. Together, they form a secure IAM tool that could be readily available for identity-as-a-service customers.

7 Conclusion and Future Work

To conclude, the discussion above states that the proper IAM solution relies on the powerful Identity management solution, which is future proof and fits any use case. Also, the Access management services like Azure AD are considered obvious solutions for managing the users and their permissions. The implementation solution presented in the research paper gives a clear and precise walkthrough to implement SensiPass® and Azure AD seamlessly through the SAML protocol. The different cloud IAM providers have only focused on the 2-factor authentication, mainly based on SMS, email, and app notification.

The IDaaS tool built with the research work has a practical use case in military-grade applications and requires a financial-based transaction. Moreover, there are other use-cases where there is a need for Identification in a chain of custody of the user who is a part of the logistics supply chain management. The tool can be used in different scenarios which are business-critical and requires robust, secure authentication.

For future work, the proposed solution can be implemented into the Metaverse concept that has been the talk of the town due to Facebook being heavily investing in the technology. Also, it would be worthy of building the whole architecture around a decentralized model using blockchain for further research work.

References

Almorsy, M., Grundy, J. and Müller, I. (2016) 'An Analysis of the Cloud Computing Security Problem', *arXiv:1609.01107 [cs]* [Preprint]. Available at: http://arxiv.org/abs/1609.01107 (Accessed: 2 January 2022).

Amin, R. *et al.* (2016) 'Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks', *Computer Networks*, 101, pp. 42–62. doi:10.1016/j.comnet.2016.01.006.

Bae, W. and Kwak, J. (2020) 'Smart card-based secure authentication protocol in multi-server IoT environment', *Multimedia Tools and Applications*, 79(23), pp. 15793–15811. doi:10.1007/s11042-017-5548-2.

Challa, S. *et al.* (2018) 'An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks', *Computers & Electrical Engineering*, 69, pp. 534–554. doi:10.1016/j.compeleceng.2017.08.003.

Fan, C.-I. and Lin, Y.-H. (2009) 'Provably Secure Remote Truly Three-Factor Authentication Scheme With Privacy Protection on Biometrics', *IEEE Transactions on Information Forensics and Security*, 4(4), pp. 933–945. doi:10.1109/TIFS.2009.2031942.

Garg, N. *et al.* (2020) 'BAKMP-IoMT: Design of Blockchain Enabled Authenticated Key Management Protocol for Internet of Medical Things Deployment', *IEEE Access*, 8, pp. 95956–95977. doi:10.1109/ACCESS.2020.2995917.

Hill, M.J., Ruddy, T.R. and SIROTA, R. (2012) 'Method and computer program for providing authentication to control access to a computer system'. Available at: https://patents.google.com/patent/WO2012164385A2/en (Accessed: 4 January 2022).

Jiang, Q. *et al.* (2017) 'Lightweight three-factor authentication and key agreement protocol for internet-integrated wireless sensor networks', *IEEE Access*, 5, pp. 3376–3392. doi:10.1109/ACCESS.2017.2673239.

Lee, J. *et al.* (2019) 'Secure Three-Factor Authentication Protocol for Multi-Gateway IoT Environments', *Sensors*, 19(10), p. 2358. doi:10.3390/s19102358.

Masud, M. *et al.* (2021) 'Lightweight and Anonymity-Preserving User Authentication Scheme for IoT-based Healthcare', *IEEE Internet of Things Journal*, pp. 1–1. doi:10.1109/JIOT.2021.3080461.

Murugesan, S. *et al.* (2001) 'Web Engineering: a New Discipline for Development of Web-Based Systems', in Murugesan, S. and Deshpande, Y. (eds) *Web Engineering: Managing Diversity and Complexity of Web Application Development*. Berlin, Heidelberg: Springer (Lecture Notes in Computer Science), pp. 3–13. doi:10.1007/3-540-45144-7_2.

Roy, S. *et al.* (2018) 'Chaotic Map-Based Anonymous User Authentication Scheme With User Biometrics and Fuzzy Extractor for Crowdsourcing Internet of Things', *IEEE Internet of Things Journal*, 5(4), pp. 2884–2895. doi:10.1109/JIOT.2017.2714179.

Saqib, M., Jasra, B. and Moon, A.H. (2021) 'A lightweight three factor authentication framework for IoT based critical applications', *Journal of King Saud University - Computer and Information Sciences* [Preprint]. doi:10.1016/j.jksuci.2021.07.023.

Wu, F. *et al.* (2017) 'An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment', *Journal of Network and Computer Applications*, 89, pp. 72–85. doi:10.1016/j.jnca.2016.12.008.

Yang, Y. *et al.* (2014) 'An Identity and Access Management Architecture in Cloud', in *2014 Seventh International Symposium on Computational Intelligence and Design*. *2014 Seventh International Symposium on Computational Intelligence and Design*, pp. 200–203. doi:10.1109/ISCID.2014.221.

Yu, S., Park, K. and Park, Y. (2019) 'A Secure Lightweight Three-Factor Authentication Scheme for IoT in Cloud Computing Environment', *Sensors*, 19(16), p. 3598. doi:10.3390/s19163598.

Yu, S. and Park, Y. (2020) 'SLUA-WSN: Secure and Lightweight Three-Factor-Based User Authentication Protocol for Wireless Sensor Networks', *Sensors*, 20(15), p. 4143. doi:10.3390/s20154143.