Article

AAAA: SSO and MFA Implementation in Multi-Cloud to Mitigate Rising Threats and Concerns Related to User Metadata

Muhammad Iftikhar Hussain [1], Jingsha He [1], Nafei Zhu [1,*], Fahad Sabah [1], Zulfiqar Ali Zardari [2], Saqib Hussain [1] and Fahad Razque [1]

[1] Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China; hussain@emails.bjut.edu.cn (M.I.H.); jhe@bjut.edu.cn (J.H.); fahad.sabah@hotmail.com (F.S.); saqibhussain@emails.bjut.edu.cn (S.H.); fahad@emails.bjut.edu.cn (F.R.)

[2] Department of Information & Communication Technologies, Begum Nusrat Bhutto Women University, 65170 Sukkur, Sindh, Pakistan; zulfiqar.ali@bnbwu.edu.pk

* Correspondence: znf@bjut.edu.cn; Tel.: +86-188-1059-9602

Abstract: In the modern digital era, everyone is partially or fully integrated with cloud computing to access numerous cloud models, services, and applications. Multi-cloud is a blend of a well-known cloud model under a single umbrella to accomplish all the distinct nature and realm requirements under one service level agreement (SLA). In current era of cloud paradigm as the flood of services, applications, and data access rise over the Internet, the lack of confidentiality of the end user's credentials is rising to an alarming level. Users typically need to authenticate multiple times to get authority and access the desired services or applications. In this research, we have proposed a completely secure scheme to mitigate multiple authentications usually required from a particular user. In the proposed model, a federated trust is created between two different domains: consumer and provider. All traffic coming towards the service provider is further divided into three phases based on the concerned user's data risks. Single sign-on (SSO) and multifactor authentication (MFA) are deployed to get authentication, authorization, accountability, and availability (AAAA) to ensure the security and confidentiality of the end user's credentials. The proposed solution exploits the finding that MFA achieves a better AAAA pattern as compared to SSO.

Keywords: multi-cloud security; federated trust in multi-cloud; multifactor authentication; single sign on; AAAA in multi-cloud

1. Introduction

Improvement in computational techniques over the last few decades and the flood of data over the Internet has led to a complex structure called multi-cloud. Multi-cloud is an advanced form of fog or heterogeneous cloud computing. A blend of various techniques, services, deployment models, cloud infrastructure, and vendor-based cloud architecture all are combined under the umbrella of multi-cloud infrastructure. Multi-cloud ensures the scaling and processing of on-demand services of cloud consumers from various environments. With its extensive controls, multi-cloud also provides security and access control transparency of data or services over the web to end users. Multi-cloud deployment helps to mitigate cybersecurity threats and energy demands on cloud service provider's end through transparent access control over the hosted services and applications [1].

Multi-cloud is an efficient way to reduce ossification of cloud services by providing all essential cloud facilities using a shared space model. It may require the allocation of both virtual and physical resources to overcome a particular need. Due to the robustness

and distributed nature of multi-cloud, security related issues regarding resources allocation are still unexplored. In our research, we provide a novel approach to address these data security and access control risks.

Because the cloud paradigm has very dense and diverse usability in the current information technology era, therefore reliance on username/password identity alone is not enough to ensure security on a shared pool of resources. Due to the diverse nature of the multi-cloud infrastructure and deployment model on vendors, security has been enhanced as multifactor authentication (MFA) [2]. Multifactor authentication has different forms depending on the infrastructure and pattern on which it has been deployed. The main factors emphasized by MFA are inheritance-based, possession-based, and biological-based metrics [3]. In MFA deployment, an attacker has the lowest chance to access the required services, as achieving access may have to clear various security measures [4].

MFA requires two or more factors in a transaction to complete a security check. One factor the user knows (e.g., a phrase, ID, or password). The second factor is one that the user possesses (e.g., token, key, or one-time password (OTP)). In the previous deployment model, two factors were enough to verify a specific user's identification. Still, as the model's diversity expands toward multi-cloud, we need to include bio-medical aspects as a third step of verification [5]. After authentication, the next step is authorization, leading to ensure user access to specific parameters, services, and models. MFA also provides accountability at every login, on each device, and for access and utilization of all resources. It also leads toward availability at the time of configuration and deployment in terms of services. We have plenty of methods to configure MFA, e.g., if a user was unable to get an OTP via short message service (SMS), we add a call option through which the user can prove his/her authentic identity. On the other hand, we have application authenticator or biometric authentication options as well [6]. MFA also ensures data confidentiality, security, access control, and trust related to specific vendors or organizations.

Single sign-on (SSO) raises overlaps with other prominent and unified techniques, including the escalation of multi-cloud, password lethargy, new developer techniques, enterprise agility, web access, and multi-cloud intuitive applications and services [7,8]. Multi-cloud architecture must be considered while integrating cloud computing with another paradigm concerning feature and advanced deployment. Multi-cloud computing is a trending evolution of cloud computing with a diverse nature, enhanced end-user experience, and extensive end-user controls. However, some security risks concerning authentication, authorization, and accountability arise on the provider end when it comes to comprehensive management and the blending of services and technologies. Various vendor services often use the same shared environment under the roof of a single service provider [9].

A federated trust at the vendor level can be configured between multiple vendors to address consumer needs under a single service level agreement (SLA). Various cross-domain access control (CDAC) protocols such as Shibboleth manage identity-based federation services in a multi-cloud environment. Shibboleth is an open-source platform designed by Howard University for testing and educational purposes. Shibboleth has an identity provider end managed by the university and used to deploy and implement single sign-on (SSO) in different federations. It is used to justify attributes designed by the identity provider (idp) and rules based on those attributes designed by the service provider. Shibboleth access for an internal database is maintained and handled by security assertion and markup language (SAML). To maintain trust between different federated domains, a Shibboleth digital signature model is used [10].

In the current digital era, end-users need to prove their authentic identities and be authorized to access various Internet services by using a user ID and password. Consequently, one password is not enough and can be compromised easily over the Internet, where we access plenty of services with the same password on different service accounts. Password managers are also used to protect access, but they are still accessible using a password over the Internet. In the research, we have learned that about 73% of people

have trouble to remembering their password, 72% use the same password for all accounts over the internet, and 50% do not care about their password security and concerns [11,12].

Dual factor authentication was introduced to verify and validate user authentication details for service accounts. Dual factor authentication uses email to send an access code, but email accounts need to use a password to fetch that access code over the Internet. On the other hand, multifactor authentication uses alternative options, e.g., SMS, call, and app authentication, for user authentication as a secondary method rather than a password. When a particular user has to access multiple services under a single user identity, users need to provide multiple passwords over the Internet. Passwords can be compromised over the web or an app access portal through various threats and attacks. In SSO, a single source of thought is used to manage multiple passwords, and hence users get verification and authorization using a single password. With an SSO managing passwords on one end, if that password is compromised then all that is required is to change only that one password; the rest of the passwords are revoked. Secondly, password requirements vary on different service accounts, e.g., length of a password, complex character, and password expiration duration. These requirements raise the level of complexity for an ordinary end user who needs to remember multiple passwords. In SSO, we can access various services from a single password. When we use a combination of SSO and MFA, if a user's single password is compromised, then the user's identity and other information related to user's attributes will not be comprised. Thus, the current research will be beneficial in terms of security compliance and end-user usability with minimum cost.

In this paper, we created a federated trust between an xyz.com hosted site and an abc.com client site, with all incoming traffic from all over the internet divided into three phases. The federated client domain user places an order as a service on the hosted or service provider end. A low risk was defined as one against the client federated user that authenticating outside the hosted domain over the web, an SSO deployed with the help of Shibboleth idp. An MFA was deployed in the case of hosted site users, as they are considered as a medium risk due to internal security attacks and user behavior. High risk users other than those in the federated domains must pass both SSO and MFA; otherwise, they are blocked under defined rules. The research exploits authentication, authorization, accountability, and availability (AAAA) achieved in both deployment cases, assuring all known user's data confidentiality and security. The proposed model secures the user's metadata from well-known modern threats and attacks, such as honeypot, brute force, social engineering, dictionary, key logger, and thwarting password attacks multi-cloud paradigm. The novel contributions of our work are:

- Deployment of a federated trust domain in a multi-cloud environment for internal and external connectivity.
- Definition of the risk model to separate user metadata used to access multi-cloud services.
- Third-party app Shibboleth IDP used to deploy a hybrid SSO-based model in a multi-cloud environment.
- Deployment of a mixed vendor-based MFA model to overcome rising security risks and threats.
- Services attack model to mitigate rising threats related to consumer metadata in the multi-cloud paradigm.

Further, this paper presents related work in Section 2, the proposed work in Section 3, methodology in Section 4, a service attack model in Section 5, and conclusions in Section 6.

2. Related Work

Honhbing Cheng et al. [13] presented an identity-based model to preserve acceptability and privacy of cloud computing by defining a simple logic to retain privacy in terms of accountability. The authors stated that the mechanism for accountability and auditing

to maintain security against adversary attacks was by comparing results and statistics using identity-based encryption in cloud computing.

Subramani Jegadeesan et al. [14] proposed an authentication-based mechanism that delivers a diverse mutual authentication model that ensures authentication of mobile consumers and service providers by secure sockets layer (SSL)-based encryption system. The presented model was helpful in measuring and controlling the computational cost and provides an effective mechanism to change the session key on both ends.

Muhammad Faheem Mushtaq et al. [15] illustrated reviews to elaborate on current security concerns and changes related to the multi-cloud computing paradigm in a review form. The authors presented an overview of the recent deployment in the multi-cloud services architecture. The proposed scheme related to security concerns and issues, which discussed the essential characteristics and services of the cloud deployment paradigm.

Vijaya Kumar Veerabathiran et al. [16] introduced a homomorphic proxy-based re-encryption novel practice for ID-based multi-cloud computing to share cloud user information remotely from one end to the other. This model presented an access control mechanism for cloud end users to ensure resource management with confidential data access and control.

Blesson Varghese et al. [17] defined some new and future trends and research directions related to modern cloud computing using edge paradigms, leading to a multi-cloud infrastructure. Multi-cloud generates multi-tier application access for the consumer with the deployment of social and technical aspects that are secure and sustainable.

Sandip Roy et al. [18] presented a demonstrable authentication-based technique for remote cloud users to access required services. In the proposed approach, third parties or IDPs were not involved during the authentication of remote users. Once a user could access the third party authenticator, it was required to provide multiple credentials depending on configurational behavior. Their proposed technique was cost-effective in terms of computation.

Yuan Zhang et al. [19] presented a secure inception password-based SSO for the authentication and identity of providers in mobile-based cloud computing. Users were grouped so that each group could utilize the server-side key once. Users could generate token requests for a limited time, configured on the IDP end. This model helped the user to save computational cost and helped the server to avoid additional storage costs.

Vanga Odelu et al. [20] designed a scheme using cryptosystem techniques to secure authentication of the mutual and anonymous user without using SSL. A smart card was used to register consumers and service providers; using a one-time single key, users could access mobile services on the cloud with minimum computational head.

Samman Zahra et al. [21] presented multi-cloud computing over internet of things (IoT) to overcome concerns and issues related to formal authentication methods. Shibboleth was used as an IDP in the middle layer between cloud consumers and services providers.

Aleksandr Ometov et al. [22] delivered an extensive report in survey form, which elaborated the factors currently applied for multifactor authentication based on industrial and educational aspects. This survey presented challenges related to adopting MFA in cases of both user and technical perspective as well as presenting a vision of future utilization of MFA in industry.

Shyamala Ramachandran et al. [23] elaborated a two-way authentication method using bilinear map functions in wireless sensor networks with time-efficient and stream-loss impact. The proposed secure grouping algorithms provide better energy awareness.

Liming Fang et al. [24] introduced a fuzzy set-based physiological and behavioral feature authentication system for the medical cloud, which was much better than fingerprint and face detection systems. A vector machine was used to classify and categorize a fuzzy data set based on re-encryption and a conditional proxy solved the data outflow issue.

Lu Zhou et al. [25] established a novel security protocol to enhance authentication in cloud-based IoT systems. The proposed scheme provides robustness in security with limited computational demands for smart devices to ensure a better authentication model.

Azeem Irshad et al. [26] introduced a multi-server authentication scheme for mobile cloud computing. The proposed system was used to mitigate vulnerability such as distributed denial of service (DDOS), spoofing, and de-centralization attacks in wireless sensor networks by using a bilinear pairing mechanism. Preeti Soni et al. [27] used three-factor authentication on behavioral and biomedical logic in remote sensor networks for patient data health care safety and security.

Subhash Chandra Patel et al. [28] presented an access control-based framework focusing on the characteristics and biological behavior needed to access cloud computing services. A mobile phone, software token, and one-time password OTP-based techniques were used to control specific applications' access.

1. A.S. Anakath et al. [29] presented a privacy-based multifactor authentication mechanism to achieve trust in the de-centralization of remote locations in cloud computing. As the end user has no transparency or access to their key-vault data store in remote areas. User access control depends on the cloud-deployment model on both the consumer and provider end.

2. Charanjeet Singh et al. [30] presented three-level MFA for cloud computing, where the first level involves OTP, the second as out-of-band (OOB) policy, and the third one screen clicks on a given image. The proposed technique is helpful for mitigating man-in-the-middle attacks (MIM).

J.K. Mohsin et al. [31] discussed the battle between two factors authentication and MFA in mobile cloud computing in a data set ranging from 2012 to 2017. This survey highlighted issues and concerns such as privacy, security, efficacy, and trust-related to mobile cloud computing.

Mohit Kumar et al. [32] categorized scheduling algorithms for their design metrics, resources measure, and implementation on particular systems, according to their designed nature and implementation in a hybrid environment. The authors also highlighted each type of scheduling algorithm with pros and cons. The research focused on the implementation of natural designed scheduling algorithms in multi-cloud deployment.

Aeyoung Kim et al. [33] proposed a secret key prevention technique for any third party using multivariate biometric key functions. A short key is generated for a specific use only with principal component analysis and confidence interval analysis techniques. The designed biometric signature scheme emphasizes storage size and CPU cycle for signature authentication and verification. A cryptographic technique was used to resist an attack on the current blockchain model. The proposed scheme had a brief method of post-quantum computing techniques and the blockchain deployment model to resist rising blockchain deployment attacks. Digital signature and public key encryption techniques were used to generate a secure communication method in quantum-based blockhead systems [34,35]. The proposed secure key generation techniques were further utilized in multi-cloud deployment to ensure multifactor authentication and authorization.

In "Table 1" we illustrate the most recently published related work, which match to our proposed methodology from various paradigms of distributed computing. We have extended these works in term of AAAA implementation in the deployment of multi-cloud architecture.

Table 1. Most recent work related to single sign-on (SSO) and multifactor authentication (MFA) implementation.

Author(s)	Concerns	Impact Vector	Proposed Solution	Method
Cheng et al. [13] IEEE Access (2018)	Privacy, adversary attacks, security	Accountability, audit, logical statistic model	Identity-based model to preserve acceptability and privacy of cloud computing	SSO

Jegadeesan [14] Sustainable Cities and Society (2019)	Authentication, SSL-based encryption	Cost, session key exchange	Diverse mutual authentication model	MFA
Veerabathiran et al. [16] Soft Computing (2020)	Remote information sharing	access control mechanism, confidentially data	ID-based homomorphic proxy and re-encryption	SSO
Zhang et al. [19] IEEE Transactions on Mobile Computing (2020)	Identity provider (IDP) authentication	Computational cost, storage cost	Inception password-based SSO model	SSO
Ometov et al. [22] Cryptography (2018)	Industrial and educational aspects	Adopting MFA	Survey based on factors currently applied for multifactor authentication	MFA
Fang et al. [24] Information Sciences (2020)	Medical cloud MFA technique	Vector machine is used to classify and categorize a fuzzy data in medical cloud	Fuzzy set-based physiological and behavioral feature authentication system	MFA
Zhou et al. [25] Future Generation Computer Systems (2019)	Authentication in IoT-based smart devices	security and computational limited smart device	security protocol to enhance authentication in cloud-based IoT	MFA
Soni et al. [27] Computer methods and programs in biomedicine (2019)	MFA in remote sensor network	Patient data's health care safety and security	Three-factors authentication on behavioral and biomedical logic	MFA
Patel et al. [28] International Journal of Green Computing (2018)	Mobile phone, software token, and one-time password	Characteristic and biological behavior to access cloud	Access control-based framework focusing on the characteristic and biological behavior	SSO
Anakath et al. [29] Cluster Computing (2019)	De-centralization of remote locations	Transparency and access on key-vault	Privacy-based multifactor authentication mechanism to achieve trust	MFA
Charanjeet et al. [30] International Journal of Computer Engineering and Technology (2019)	Mitigating man-in-the-middle (MIM) attacks	One-time password (OTP), out-of-band (OOB), and image click	Three-level MFA for cloud computing	MFA
Kim et al. [33] KSII Transactions on Internet and Information Systems (2020)	Storage size and CPU cycle for the signature authentication	Post-quantum computing methods and blockchain authentication	Secret key prevention technique for any third party using the multivariate biometric key functions	SSO and MFA
Fernandez et al. [34] IEEE Access (2020)	Privacy and authentication in quantum computing	Post-quantum cryptosystems	Public-key cryptography and hash functions-based model	SSO and MFA

3. Proposed Work

In an I.T. based organization, end-users always complain about putting so many passwords on various applications and having to remember so many passwords. On the other hand, it is not the elementary job for an I.T. management team to handle so many password repositories. Once a user changes his/her password, it raises another difficulty to manage the password in all the other related directories. SSO provides a solution to get rid of all these difficulties. MFA provides alternative authentication mechanisms to ensure the authenticity of a particular user in multiple dimensions for multiple times. The combination of SSO and MFA provides AAAA to ensure a trustworthy and healthy communication between users of a single domain and different domains on a single site.

In our proposed work, we created an organization named "xyz.com" to act as a hosted party domain, which has the role of a services provider end in a multi-cloud environment. This provider end, known as the first party, has hosted SSO- and MFA-based solutions and other cloud services in the multi-cloud services paradigm. The second party organization we created is "abc.com," known as the client organization. In our scenario, the client's party puts orders on the hosted organization site. The hosted organization does not allow these orders to be authenticated on their site using the hosted active directory. Consequently, a federated trust was deployed between xyz.com and abc.com [36].

Therefore, we are using a third-party solution, which is very readily available and specially designed for study purposes "Shibboleth" by Howard University. Shibboleth

is used to authenticate user orders outside of the hosted site on a designated idp. Following Figure 1 steps define our proposed work.
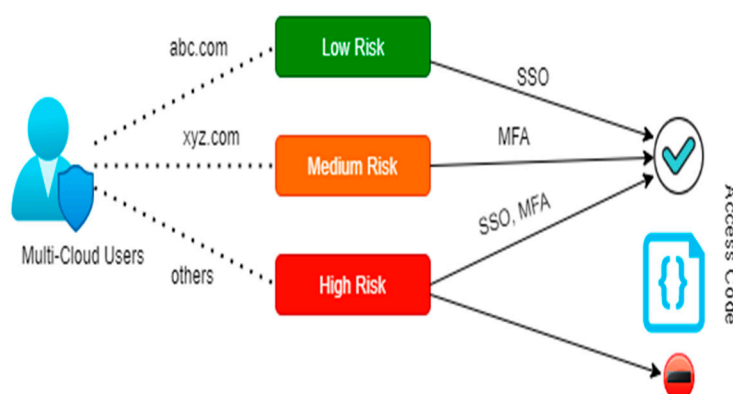

Figure 1. Risk level assessment and proposed mitigation techniques.

Step 1: Users for different domains log in to abc.com to place their order. This order is hosted on xyz.com. Orders from abc.com authenticate on the web portal as the order's data are filtered and authenticity is proven on the web login. If the data are categorized as low risk, different processes can be accessed by the single sign-on. We cannot deploy additional security because we do not have the metadata needed to access the data. Once the users provide login information, IDP will give an authority certificate to complete their order.

Step 2: The users from xyz.com can access services by providing multifactor authentication on their access. This multifactor authentication will provide authenticity and per user accountability. The users from xyz.com are considered a medium risk as their authenticity depends on the hosted services environment. Therefore, we deployed multifactor authentication (MFA) to make it more secure and accountable.

Step 3: Other users from outside the federated trust environment will be declined if they do not prove both SSO and MFA conditions. We mark them as "high-risk" because they do not belong to any federated or hosted domain. Therefore, they are restricted in their ability to access and complete their order.

4. Methodology

Users from abc.com provide authentication on web login while going to place an order on the hosted site. The SSO operation will be performed when the order metadata arrive on xyz.com. Since the first party and third party are interconnected with each other via the domain level as a federated trust, SSO provides authentication and authorization. The SSO workflow is presented in the following diagram Figure 2.
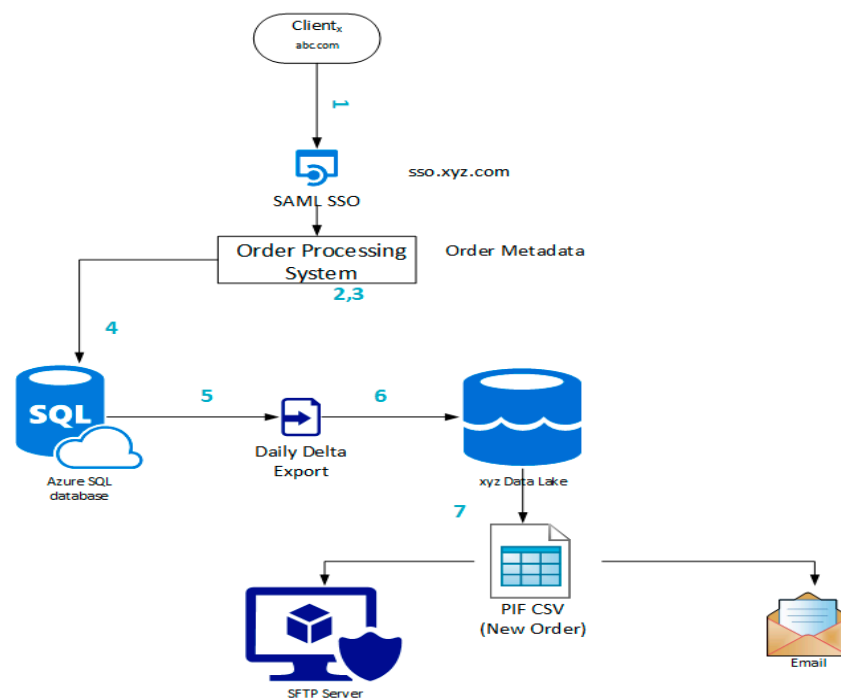
Figure 2. Medium risk mitigation workflow.

Step 1: In the first step, users outside of abc.com log on through the web to place their order. This order file is forwarded to the second-party federated domain in the form of security assertion markup language (SAML) [37].

Steps 2, 3: The xyz.com domain fetches this SAML file and starts the ordering process. SSO is performed here by an open-source and readily available third part application, Shibboleth. Howard University designed this application for testing and study purposes [38]. On the back end, the university has its identity provider IDP to ensure the authenticity and prove the authority of any order. In order, metadata have a private key of the order; on the IDP end, the public access is stored in trusted domain order metadata. The IDP compares private and public keys and generates a transport layer security (TLS) certificate to the order.

Step 4: On the hosted site, order data are stored in the Azure cloud-based SQL database temporarily. Daily databases are exported from Azure SQL databases in CSV format.

Step 5: Daily delta export process fetch the per day, data from Azure SQL databases and store it on XYZ side in Data Lake Storage.

Step 6: Data Lake Storage is a particular type of storage without retention policies and limited storage time. On the other hand, Data Lake Storage does not provide fully secure protocols for the stored data [39].

Step 7: A query fetches the data from Data Lake Storage and stores a copy of the order metadata in secure file transfer protocol (SFTP). SFTP is used to transfer large files over the web with standard secure shell (SSH) security. One copy of these metadata are emailed to a specific group to maintain easy reliability.

4.1. Example of the Order Process

As an example of a multi-cloud environment, we tried to create a blend of Azure and active amazon web services (AWS) cloud. In this example, we present an order data workflow with the AWS model. The order placement process shown in the Figure 3 below. Thus, how an order can be placed from AWS is as follows.

- Order initiated on clientx site "abc.com" and the SAML assertion issuer generates a SAML file to shift it to AWS Route 53 services for the next step.

- AWS Route 53 provides a cloud domain name service to translate the name into a specified IP address. Therefore, it will decode SAML files and fetch the required information from metadata in digits format and redirect it to a load balancer for the next process.
- The application's load balancer provides content-based and specific protocol-based routing and delivery of the content. The content is monitored in this section and separated into the required content in CSV format. This sorted content is redirected toward the application target group.
- A target group informs the load balancer where the traffic is routed to the load balancer with a specific amazon elastic compute cloud (EC2) container, lambda functions, and specific IP address. Content with name xyz-sso-v1-target-group-443 is redirected on port 443.
- AWS Fargate works as "elastic container service" (ECS) and makes it very simple to deploy AWS container services. In this step, AWS container services run with the help of AWS Fargate. The container cluster is configured as EC2ContainerService-xyz-cluster-www-main.
- Running container services performs two tasks. Firstly, the data are stored in the AWS database and required data are emailed to the specific email group. Container 1 (Task definition): xyz-sso-v1-task: 3 (Container Port: 443). Secondly, it transfers the data to Shibboleth IDP for further processing and assigns an authority certificate.
- Container 2 (Task definition): xyz-sso-v1-idp-task: 5 (Container Port: 443). Shibboleth IDP matches the private key of metadata with the public key stored and assigns the TLS certificate as an authority.
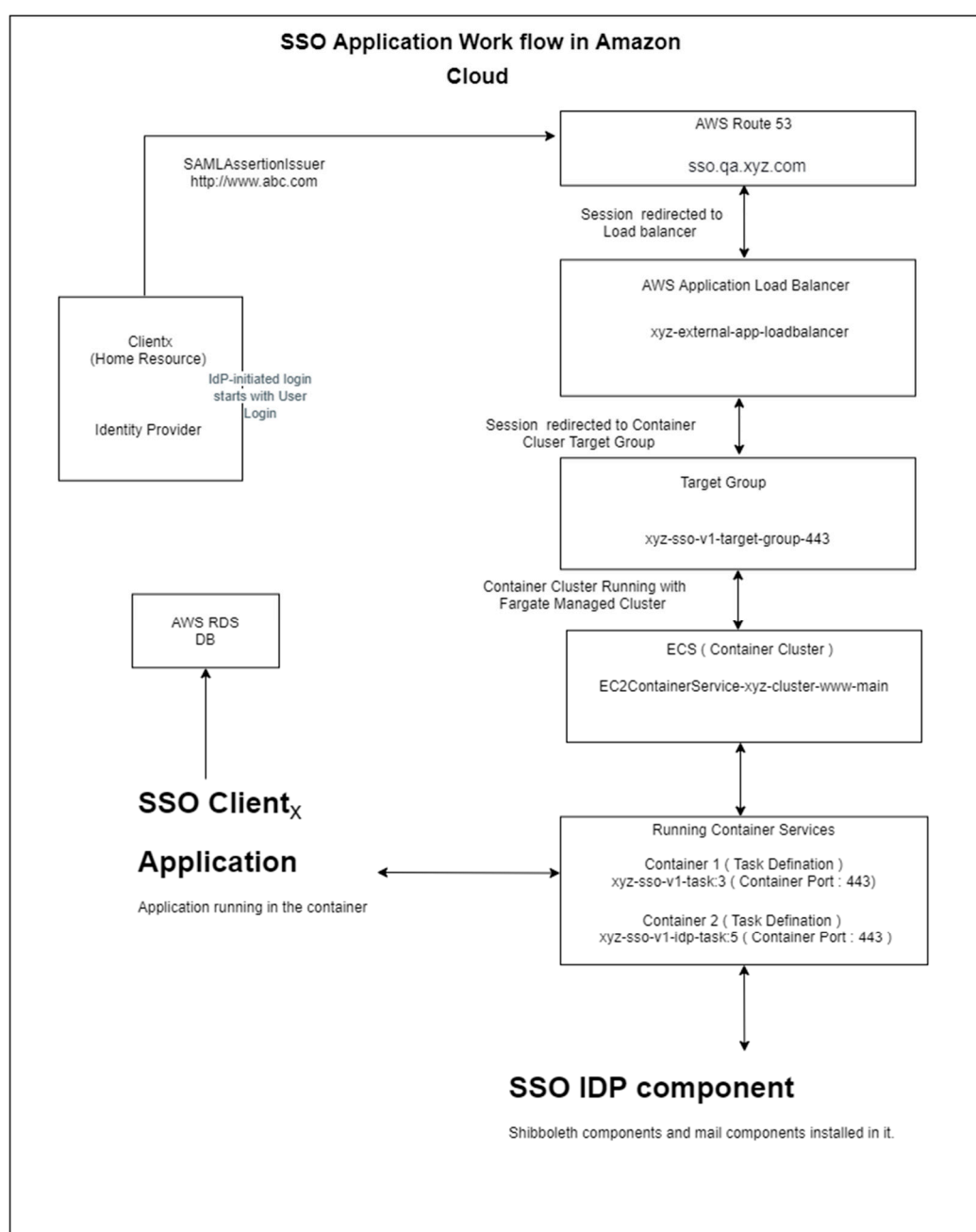
Figure 3. SSO implementation with AWS and Shibboleth.

4.2. Coding Example

In the section below we describe the functionality of a daily order processed between two federation sites in terms of code. With the presented code format, it is very easy to understand the development and design of the application-programming interface (API) used for the functionality of the whole working process.

Here is the pseudo-code above the AWS order workflow.

AWS order workflow algorithm

Function lambda handler
    Pass In: event, context

    Connect to database and store connection
    Create a temporary table in database to store the changed rows

```
                    Open file handler to csv

        FOR every row in temporary table
                        Store row data in csv file
                ENDFOR

                Save file on file system
                Remove temporary table
                Close database connection

                IF csv file not empty
                        Connect to email client
                        Compose, attach file and send email to specified address
                        PRINT "Email Sent the required address"
                ELSE
                        PRINT "No records to notify. No email sent"

                Pass out: nothing
        ENDFUNCTION
```

4.3. Multifactor Authentication

Multifactor authentication is a security system that requires more than one-time user credential information to prove authenticity and user accountability. User accountability is in terms of per device login and multi-login on different devices. MFA provides authorization after approving the multi-time credential of a single user [40]. MFA has a separate makeup as a password first and then has various secondary options such as SMS to a cell number, a call on a cell phone, email recovery through a one-time password, and application authentication to approve individual application access by a particular user.

MFA provides a multi-layer fence that makes it more complicated for unauthorized users to access specific applications and services. In a multi-cloud domain where the same resources are shared with consumers from various fields and regions, MFA plays a vital role to ensure the AAAA security model. Should attackers break one barrier such as acquiring a user's password from various masquerading attacks, the attacker still has to clear one more fence to prove authenticity in our proposed model. We deployed a real-time MFA model with the combination of Azure and Cisco cloud service. We used the latest authentication scenarios in both cases. In the case of Azure, for the first authentication, multi-cloud users authenticated from Azure active directory via a radius server and then for secondary option used Cisco Meraki firewall based authentication

The second step of our AAAA multi-cloud design model is the combination of SSO and MFA. We used MFA for the internal use of the federated domain of xyz.com, on which we could use our designed Azure active directory for services. The following diagram presents the MFA model and its implication in terms of high risk. We term it high risk because we are using the internal environment and our home directory and services for these particular users.

4.4. Authentication Metrics

An authentication metric is a type of user metadata used for proof of identity. MFA is used to prove who you are, what you have, and where you got the access. The most commonly used three metrics are as follows [41].

Knowledge metric: It is based on knowledge-based authentication (KBA). Here, consumers need to answer some secret questions that only the authentic user should know.

Possession metric: Here, the user needs something special to prove authenticity, such as a one-time password (OTP) token, verification code, SMS code, etc.

Inheritance metric: Users are required to prove their authority via biometrics, such as iris detection, face detection, fingerprint, etc.

4.5. Designed Model MFA Methods

In our designed model, there are three main methods configured to prove authentication [42]. Multifactor authentication process defined in Figure 4 below.

SMS: The user receives an SMS on a given cell number and put this SMS OTP into a required field. We do not use this option as Meraki Client firewall servers no longer support SMS-based authentication.

Call on Phone: Users receive a call on their given number and prove authorization by pressing the numeric digit on a cell phone keypad. It is a beneficial and effective method in our case.

App approval: Users need to configure the Microsoft authenticator on their cell phones, and authority will be approved by assigning access on request.
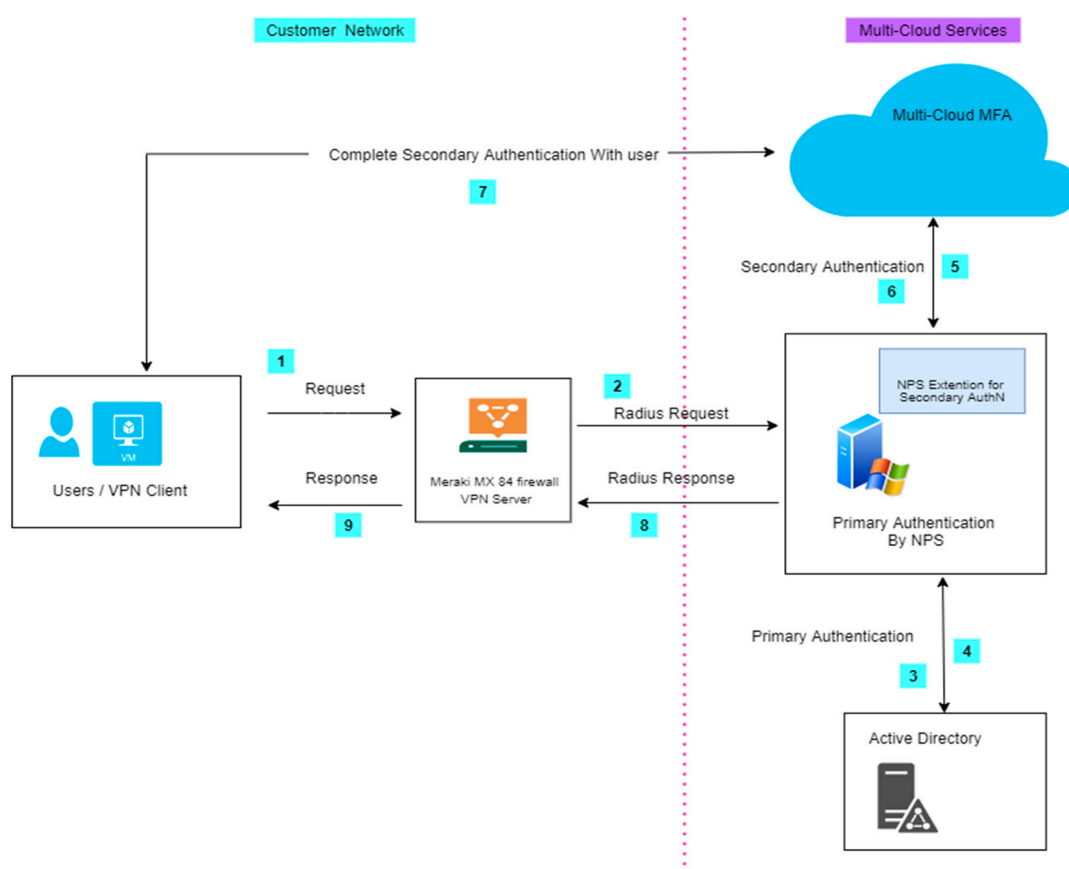

Figure 4. MFA implementation with azure and cisco. NPS, network policy server.

In this figure, we divided the topology into two sections: customer network and multi-cloud services zone. Complete configuration steps of the given model are as follows:

Step 1: When a user tries to utilize services, an automatic message request is sent to the Meraki MX 84 firewall virtual private network (VPN) server. Users must connect to the VPN client via a username and password.

Step 2: The Meraki firewall VPN server is directly connected to the central radius server called a network policy server (NPS). Therefore, it forwards the client request to the NPS. The NPS has central position radius servers connected on both ends for primary authentication via Azure active directory and secondarily via multi-cloud MFA. The NPS is also responsible for replying to the required request code to the Meraki firewall server.

Step 3, 4: In these steps, the NPS will shake hands with the Azure active directory to prove primary authentication. In return, the Azure active directory sends a token back to the NPS server. Hence, the first authentication will be confirmed; it will also count attempts per device and per user as accountability as per login based.

Step 5, 6: For the secondary authentication, the NPS forwards the token request to multi-cloud MFA, where users need to configure their desired authentication choice, e.g., call, SMS, or app authentication. In case in which we are using Meraki, we are limited to using the call or app authentication option. The SMS option is not supported by Meraki firewall.

Step 7: In this step, the user will receive any of his/her configuration options and receive a call or approve access from the Microsoft authenticator application. Hence, secondary authentication is proved, and authorization and availability are confirmed—availability in the sense that users have multiple options to verify their secondary authentication.

Step 8, 9: These steps involve handshaking to ensure two-way connectivity between the user VPN client, the Meraki firewall server, and the NPS.

5. Results and Discussions

In the case of SSO, the implementation code and its workflow should be in mathematical notation form. Mathematical notations are used to calculate the cost in terms of computational behavior. We have used SSO with multifactor authentication to achieve AAAA authentication, authorization, accountability, and availability. The whole code has been divided into three parts: (1) running once per day, (2) loop for a specific time, and (3) use for built-in functions and libraries.

The placed orders from the abc.com federated domain process execution and metadata processing occur once per day; hence, the algorithm cost for these constant functions will be $O(1)$. At the same time, the tasks that are running once will be $\Omega(1)$. Here the value is constant or running once, so the total cost will remain the same during the whole algorithm for these functions.

In the case of exporting data to CSV, the format loop will run as per the number of orders in a whole day; hence, the algorithm cost will change to $O(n)$, representing the constant evaluation of the required functional loops. For the built-in library function, inner function, and Shibboleth functionality, the algorithm's cost depends on $o(n)$ and $w(n)$. As much as the size of input or required function increases, it will be increased in parallel.

The following diagram presents the AAAA comparison as per the number of requests between SSO and MFA. In our presented scenario, the number of federated orders via SSO varied per day in 15,000–20,000. Internal user access to the system via MFA was about 5000 per day. We present the AAAA for both the SSO and MFA, respectively. Figure 5 shows the plotting of the highest range rule to avoid complexity and to make it easy to understand.
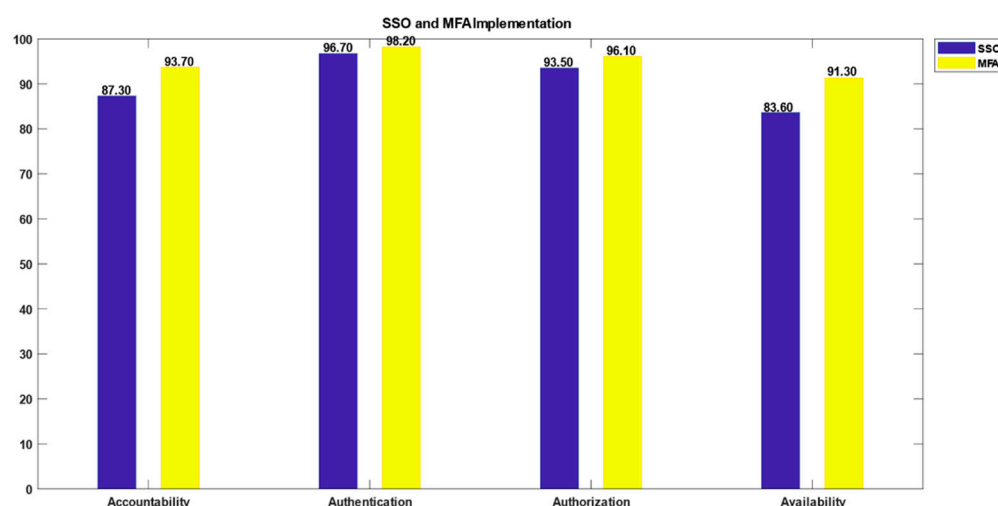
Figure 5. Authentication, authorization, accountability, and availability (AAAA) comparison of SSO and MFA implementation.

The Figure 5 shows that the percentage of accountability and availability create a massive difference between SSO and MFA deployment in prosed work. Results were obtained by implementing a test-running environment in the various need-based scenarios. We obtained quite similar results in the case of authentication and authorization. Hence, it can be concluded that in our design model, MFA proves better AAAA as compared to SSO.

Services Attack Model

In this section we discuss the rising threats and concerns related to multi-cloud and how our proposed technique can overcome these increasing threats and attacks in this domain.

Honeypot attacks: A honey pot is a security breach system projected to mimic a target in a cybersecurity system. A loophole is created in front of the original approach to trap malicious attacker activity; a particular action is taken when an attacker is wholly involved in this copycat system [43]. Our designed system can create these types of honeypots and take appropriate action before accessing the original approach to ensure AAAA in a multi-cloud domain.

Dictionary attacks: Hackers use a list of common words or a dictionary with different combinations multiple times to access the environment. This password-breaking request is useless, as our designed MFA- and SSO-based topology had multifactor and IDP authentication.

Brute force attacks: A list or string of password phrases is applied multiple times to access a designed automated tool with random character sets. MFA and SSO based implementation has a complete barrier to stop these automatic password attacks to avoid stealing access.

Traffic interception attack: Cybercriminal users use automated software such as packet sniffers to access the password between the access and service providers [44]. MFA deployment gets rid of this type of attack as we have multiple authentication and authorization scenarios.

Man-in-the-middle attack: These are sniffers using spoofing tools to access user credentials in an intermediary act. Our designed scheme provides a complete barrier against this type of attack to ensure AAAA in a multi-cloud environment.

Key logger attack: In these attacks, an attacker has software to access keystrokes and log files and to gather useful information from it. Still, SSO implementation ensures avoiding this type of attack due to IDP-based authentication and authorization.

Social engineering attacks: Our proposed scheme helps overcome well-known social engineering attacks like phishing, baiting, and quid quo pro [45], as we have deployed an AAAA environment with multifactor and IDP-based authentication and web authentication is the federated domain's responsibility before entering the host network.

Thwarting password attack: Firm password policy or alphanumeric passwords are not enough these days due to rapid access and the high performance tools available. A single time password is not secure enough to mitigate rising concerns and our current age of such issues [46]. Our designed system ensures multifactor authentication, authorization, accountability, and availability. It provides surety to reduce this increasing type of threat of the modern age over the Internet.

Table 2 describes the complete mechanism of proposed work, the attack surface, and designed metric, and how much it will help to reduce the above listed attacks in a multi-cloud paradigm. Definitions of the terms used in Table 2 are as follows:

- Attack surface is defined according to the proposed model in multi-cloud.
- Impact in terms of severity level, which causes damage in the environment.
- Designed metrics define the parameters used by the particular technique to overcome the attack.
- Proposed solution is defined as the MFA, SSO, or combination of MFA and SSO deployment.
- The efficacy of the proposed technique to avoid the particular attack in multi-cloud paradigm is defined in terms of the efficiency and cost effectiveness. Efficacy is measured in terms of how the proposed technique justifies and overcome attacks in terms of percentage.

Table 2. Listed parameters and efficacy of the proposed work in terms of percentage.

	Attack Name	Attack Surface	Impact	Designed Metrics	Proposed Solution	Efficacy
1	Honeypot attacks	Email traps, malware honeypot, decay database, spider honeypot	High	User credentials and identity	Hybrid (MFA and SSO)	95%
2	Dictionary attacks	Auto login, auto fill, remember password, secondary access code	Medium	User metadata and schema	SSO	96%
3	Brute force attacks	Spam ads, tracking malware, website redirection	Medium	Credentials stuffing	MFA	98%
4	Traffic interception attacks	Credentials theft, intrusion traps	Low	Metadata scamming	MFA	93%
5	Man-in-the-middle attacks	Intrusion detection traps, traffic analyzer, scamming	Low	Traffic header, attached metadata, redirection	MFA	96%
6	Key logger attacks	Key stork, login information	Medium	Information estimation	SSO	97%
7	Social engineering attacks	Victim identification, track covering, foothold expanding, story sapping	Medium	Background information gathering, target engagement, data siphoning	Hybrid (MFA and SSO)	94%
8	Thwarting password attacks	Dictionary, brute force, man-in-the-middle, social engineering, key logger, traffic interception	High	User, information, credentials and metadata	Hybrid (MFA and SSO)	96%

6. Conclusions

This paper focused on the current issues related to password protection and user credentials confidentiality, as users have to apply multiple passwords to access a variety of services on multi-cloud. We created a testing environment using federated trust domain techniques to deploy SSO and MFA on a designed scenario. In the case of SSO, we used a hybrid cloud environment with a third-party designed IDP. A multifactor authentication model was deployed in a multi-cloud environment with the help of Microsoft and

CISCO. AAAA achieved the security and confidentiality of the user's metadata with extensive support.

The designed paradigm helps to mitigate current rising threats and attacks related to user's metadata in a multi-cloud environment, e.g., honeypot, brute force, social engineering, and the dictionary key logger, as well as thwarting password attacks in a multi-cloud paradigm. The results show that MFA performs better in AAAA functionality as compared with SSO. Deployment of the designed paradigm in different series and need-based designs should be a future study.

Author Contributions: Data curation, Methodology, M.I.H.; Conceptualization, J.H.; Formal analysis, Funding acquisition, N.Z.; Investigation, F.S.; Software, J.H., F.S.; Validation, Z.A.Z.; Resources, software, S.H.; writing, Z.A.Z., F.R.; review and editing, F.R. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Atat, R.; Liu, L.; Wu, J.; Li, G.; Ye, C.; Yang, Y.; Yi, Y. Big Data Meet Cyber-Physical Systems: A Panoramic Survey. *IEEE Access* **2018**, *6*, 73603–73636, doi:10.1109/access.2018.2878681.
2. Akinrolabu, O.; New, S.; Martin, A. Assessing the Security Risks of Multicloud SaaS Applications: A Real-World Case Study. In Proceedings of the 2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), Paris, France, 21–23 June 2019; pp. 81–88.
3. Akinrolabu, O.; Nurse, J.R.; Martin, A.; New, S. Cyber risk assessment in cloud provider environments: Current models and future needs. *Comput. Secur.* **2019**, *87*, 101600, doi:10.1016/j.cose.2019.101600.
4. Kumar, R.; Goyal, R. On cloud security requirements, threats, vulnerabilities and countermeasures: A survey. *Comput. Sci. Rev.* **2019**, *33*, 1–48, doi:10.1016/j.cosrev.2019.05.002.
5. Nguyen, D.C.; Pathirana, P.N.; Ding, M.; Seneviratne, A. Integration of blockchain and cloud of things: Architecture, applications and challenges. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2521–2549.
6. Singh, M.M.; Ching, K.W.; Manaf, A.A. A novel out-of-band biometrics authentication scheme for wearable devices. *Int. J. Comput. Appl.* **2018**, *42*, 589–601, doi:10.1080/1206212x.2018.1547347.
7. Vehniä, V.J. Implementing Azure Active Directory Integration with an Existing Cloud Service. Master's Thesis, University of VAASA, Vaasa, Finland, 2020.
8. Arunarani, A. R., Manjula, D., & Sugumaran, V. (2019). Task scheduling techniques in cloud computing: A literature survey. Future Generation Computer Systems, 91, 407-415.
9. Bhagyoday, R.; Kamani, C.; Bhojani, D.; Parmar, V. Comprehensive Study of E-Health Security in Cloud Computing. *Int. Res. J. Eng. Technol. (IRJET)*, **2019**, 2395-0056.
10. Bendiab, G.; Shiaeles, S.; Boucherkha, S.; Ghita, B. FCMDT: A novel fuzzy cognitive maps dynamic trust model for cloud federated identity management. *Comput. Secur.* **2019**, *86*, 270–290, doi:10.1016/j.cose.2019.06.011.
11. Yıldırım, M.; Mackie, I. Encouraging users to improve password security and memorability. *Int. J. Inf. Secur.* **2019**, *18*, 741–759, doi:10.1007/s10207-019-00429-y.
12. Pilar, D.R.; Jaeger, A.; Gomes, C.F.A.; Stein, L.M. Passwords Usage and Human Memory Limitations: A Survey across Age and Educational Background. *PLoS ONE* **2012**, *7*, e51067, doi:10.1371/journal.pone.0051067.
13. Cheng, H.; Rong, C.; Qian, M.; Wang, W. Accountable Privacy-Preserving Mechanism for Cloud Computing Based on Identity-Based Encryption. *IEEE Access* **2018**, *6*, 37869–37882, doi:10.1109/access.2018.2851599.
14. Jegadeesan, S.; Azees, M.; Kumar, P.M.; Manogaran, G.; Chilamkurti, N.; Varatharajan, R.; Hsu, C.-H. An efficient anonymous mutual authentication technique for providing secure communication in mobile cloud computing for smart city applications. *Sustain. Cities Soc.* **2019**, *49*, 101522, doi:10.1016/j.scs.2019.101522.
15. Faheem, M.; Akram, U.; Khan, I.; Naqeeb, S.; Shahzad, A.; Ullah, A.; Mushtaq, M.F. Cloud Computing Environment and Security Challenges: A Review. *Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 183–195, doi:10.14569/IJACSA.2017.081025.
16. Veerabathiran, V.K.; Mani, D.; Kuppusamy, S.; Subramaniam, B.; Velayutham, P.; Sengan, S.; Krishnamoorthy, S. Improving secured ID-based authentication for cloud computing through novel hybrid fuzzy-based homomorphic proxy re-encryption. *Soft Comput.* **2020**, *24*, 18893–18908, doi:10.1007/s00500-020-05119-9.
17. Varghese, B.; Buyya, R. Next generation cloud computing: New trends and research directions. *Futur. Gener. Comput. Syst.* **2018**, *79*, 849–861, doi:10.1016/j.future.2017.09.020.
18. Roy, S.; Chatterjee, S.; Das, A.K.; Chattopadhyay, S.; Kumar, N.; Vasilakos, A.V. On the Design of Provably Secure Lightweight Remote User Authentication Scheme for Mobile Cloud Computing Services. *IEEE Access* **2017**, *5*, 25808–25825, doi:10.1109/access.2017.2764913.

19. Y. Zhang, C. Xu, H. Li, K. Yang, N. Cheng and X. S. Shen, "PROTECT: Efficient Password-based Threshold Single-sign-on Authentication for Mobile Users against Perpetual Leakage," in IEEE Transactions on Mobile Computing, doi: 10.1109/TMC.2020.2975792.

20. Odelu, V.; Das, A.K.; Kumari, S.; Huang, X.; Wazid, M. Provably secure authenticated key agreement scheme for distributed mobile cloud computing services. *Futur. Gener. Comput. Syst.* **2017**, *68*, 74–88, doi:10.1016/j.future.2016.09.009.

21. Zahra, S.; Alam, M.; Javaid, Q.; Wahid, A.; Javaid, N.; Malik, S.U.R.; Khan, M.K. Fog Computing Over IoT: A Secure Deployment and Formal Verification. *IEEE Access* **2017**, *5*, 27132–27144, doi:10.1109/access.2017.2766180.

22. Ometov, A.; Bezzateev, S.; Mäkitalo, N.; Andreev, S.; Mikkonen, T.; Koucheryavy, Y. Multi-Factor Authentication: A Survey. *Cryptography* **2018**, *2*, 1, doi:10.3390/cryptography2010001.

23. Ramachandran, S.; Shanmugam, V. A two way authentication using bilinear mapping function for wireless sensor networks. *Comput. Electr. Eng.* **2017**, *59*, 242–249, doi:10.1016/j.compeleceng.2016.12.009.

24. Fang, L.; Yin, C.; Zhou, L.; Li, Y.; Su, C.; Xia, J. A physiological and behavioral feature authentication scheme for medical cloud based on fuzzy-rough core vector machine. *Inf. Sci.* **2020**, *507*, 143–160, doi:10.1016/j.ins.2019.08.020.

25. Zhou, L.; Li, X.; Yeh, K.H.; Su, C.; Chiu, W. Lightweight IoT-based authentication scheme in cloud computing circumstance. *Futur. Gener. Comput. Syst.* **2019**, *91*, 244–251.

26. Irshad, A.; Sher, M.; Ahmad, H.F.; Alzahrani, B.A.; Chaudhry, S.A.; Kumar, R. An improved Multi-server Authentication Scheme for Distributed Mobile Cloud Computing Services. *TIIS* **2016**, *10*, 5529–5552.

27. Soni, P.; Pal, A.K.; Islam, S.H. An improved three-factor authentication scheme for patient monitoring using WSN in remote health-care system. *Comput. Methods Programs Biomed.* **2019**, *182*, 105054, doi:10.1016/j.cmpb.2019.105054.

28. Patel, S.C.; Jaiswal, S.; Singh, R.S.; Chauhan, J. Access Control Framework Using Multi-Factor Authentication in Cloud Computing. *Int. J. Green Comput.* **2018**, *9*, 1–15, doi:10.4018/ijgc.2018070101.

29. Anakath, A.S.; Rajakumar, S.; Ambika, S. Privacy preserving multi factor authentication using trust management. *Clust. Comput.* **2019**, *22*, 10817–10823, doi:10.1007/s10586-017-1181-0.

30. Singh, C.; Singh, D. A 3-Level Multifactor Authentication Scheme for Cloud Computing. *Int. J. Comput. Eng. Technol.* **2019**, *10*, 184–195.

31. Mohsin, J.K.; Han, L.; Hammoudeh, M.; Hegarty, R. Two Factor Vs Multi-factor, an Authentication Battle in Mobile Cloud Computing Environments. In Proceedings of the International Conference on Big Data and Internet of Thing—BDIOT2017; London, UK, 20–22 December 2017; p. 39.

32. Kumar, M.; Sharma, S.C.; Goel, A.; Singh, S.P. A comprehensive survey for scheduling techniques in cloud computing. *J. Netw. Comput. Appl.* **2019**, *143*, 1–33.

33. Kim, A.; Wang, C.; Seo, S.-H. PCA-CIA Ensemble-based Feature Extraction for Bio-Key Generation. *KSII Trans. Internet Inf. Syst.* **2020**, *14*, 2919–2937, doi:10.3837/tiis.2020.07.011.

34. Fernandez-Caramés, T.M.; Fraga-Lamas, P. Towards Post-Quantum Blockchain: A Review on Blockchain Cryptography Resistant to Quantum Computing Attacks. *IEEE Access* **2020**, *8*, 21091–21116, doi:10.1109/access.2020.2968985.

35. Petrache, A. L., & Suciu, G. (2020). Security in Quantum Computing. *Annals of Disaster Risk Sciences: ADRS*, *3*(1), 43-50.

36. Challagidad, P. S., & Birje, M. N. (2020). Multi-dimensional dynamic trust evaluation scheme for cloud environment. Computers & Security, 91, 101722.

37. Karie, N.M.; Kebande, V.R.; Ikuesan, R.A.; Sookhak, M.; Venter, H.S. Hardening SAML by Integrating SSO and Multi-Factor Authentication (MFA) in the Cloud. In Proceedings of the 3rd International Conference on Networking, Information Systems & Security, Marrakech, Morocco, 15 December 2020; pp. 1–6.

38. Singh, S.; Ra, I.H.; Meng, W.; Kaur, M.; Cho, G.H. SH-BlockCC: A secure and efficient Internet of things smart home architecture based on cloud computing and blockchain technology. *Int. J. Distrib. Sens. Netw.* **2019**, *15*, 1550147719844159.

39. Khine, P.P.; Wang, Z.S. Data lake: A new ideology in big data era. In 4th Annual International Conference on Wireless Communication and Sensor Network; EDP Sciences, Wuhan, China, 15–17 December 2017; Volume 17, p. 03025.

40. Acar, A., Liu, W., Beyah, R., Akkaya, K., & Uluagac, A. S. (2019). A privacy-preserving multifactor authentication system. Security and Privacy, 2(5), e88.

41. Kaleeswari, C.; Maheswari, P.; Kuppusamy, K.; Jeyabalu, M. A brief review on cloud security scenarios. *Int. J. Sci. Res. Sci. Technol.* **2018**, *4*, 46–50.

42. El-Hajj, M., Fadlallah, A., Chamoun, M., & Serhrouchni, A. (2019). A survey of internet of things (IoT) authentication schemes. Sensors, 19(5), 1141.

43. Devi, B.T.; Shitharth, S.; Jabbar, M.A. An Appraisal over Intrusion Detection Systems in Cloud Computing Security Attacks. In Proceedings of the 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, 5–7 March 2020; pp. 722–727.

44. Idhammad, M.; Afdel, K.; Belouch, M. Distributed Intrusion Detection System for Cloud Environments based on Data Mining techniques. *Procedia Comput. Sci.* **2018**, *127*, 35–41, doi:10.1016/j.procs.2018.01.095.

45. Albladi, S.M.; Weir, G.R. User characteristics that influence judgment of social engineering attacks in social networks. *Human-centric Comput. Infor. Sci.* **2018**, *8*, 1–24.

46. Schechter, S.; Tian, Y.; Herley, C. StopGuessing: Using Guessed Passwords to Thwart Online Guessing. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy (EuroS&P); IEEE, 2019; pp. 576–589.