Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 561487, 15 pages
http://dx.doi.org/10.1155/2014/561487

Research Article

Secure OpenID Authentication Model by Using Trusted Computing

E. Ghazizadeh, Z. S. Shams Dolatabadi, R. Khaleghparast, M. Zamani, A. A. Manaf, and M. S. Abdullah

Advanced Informatics School, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia

Correspondence should be addressed to M. Zamani; mazdak@utm.my

Received 18 June 2014; Revised 9 September 2014; Accepted 15 September 2014; Published 27 November 2014

Academic Editor: Shen Yin

The growth of Internet online services has been very quick in recent years. Each online service requires Internet users to create a new account to use the service. The problem can be seen when each user usually needs more than one service and, consequently, has numerous accounts. These numerous accounts have to be managed in a secure and simple way to be protected against identity theft. Single sign-on (SSO) and OpenID have been used to decrease the complexity of managing numerous accounts required in the Internet identity environment. Trusted Platform Module (TPM) and Trust Multitenancy are great trusted computing-based technologies to solve security concerns in the Internet identity environment. Since trust is one of the pillars of security in the cloud, this paper analyzes the existing cloud identity techniques in order to investigate their strengths and weaknesses. This paper proposes a model in which One Time Password (OTP), TPM, and OpenID are used to provide a solution against phishing as a common identity theft in cloud environment.

1. Introduction

Human dependency on technology has highly increased during the last ten years, and people are in need for computer networks to be up-to-date with news, do electronic shopping, communicate, send and receive emails, and do many of their daily activities. Therefore, it is very important to defend the availability and integrity of all the network systems against various threats [1].

Also, enterprises of all sizes are embracing cloud computing because of the many advantages it provides. These include lower costs, greater business agility, reduced IT administrative overhead, access to best-of-breed applications, and more. Industry analyst firm IDC reports that the SaaS market reached $16.6 billion in revenue in 2010, and it is projected to grow at more than 25% per year between now and 2015.

The cloud contains solutions that address virtually any conceivable business need, including sales, marketing, human resources, collaboration and communication, finance, and so forth. However, this proliferating profusion of solutions has created a daunting operational challenge: how to efficiently manage the profusion of identities that users require—one for each cloud application they access. If you have 1,000 employees, each accessing 10 cloud applications, that will be 10,000 unique identities to manage in total.

In the "Cloud Computing Technology Roadmap," the National Institute of Standards and Technology (NIST) suggests that there is "the need for trusted identities and secure and efficient management of these identities, while users' privacy is protected, is a key element for the successful adoption of any cloud solution." The best way to address these concerns is to deploy strong identity management processes and technologies to ensure that only authorized users have access to cloud applications, which has been attended in this research as well [2–5].

Using the cloud has numerous advantages such as savings, service flexibility, and configurable computing resources; however, privacy and security are the essential concerns to a wide adoption of clouds. Resource sharing, multitenancy, and outsourcing are the new concepts that clouds introduce. Consequently, this creates new challenges to the security community to whom these challenges are addressed:

the ability to promote and tune the security measures developed for traditional computing systems, proposing new models, security policies, and protocols to address the exclusive cloud security challenges [2, 6].

Trusted computing infrastructure systems have expected predictable ways of behaving. Therefore, to enforce these behaviours, hardware and software works are both needed. Also, there is a consistency of behaviors across computing on servers, networks, and storage elements in the data center [7]. Based on the specifications developed by Trusted Computing Group, six key technology notions are included in trusted computing to have a trustable system. These six keys are as follows: secure input and output, remote attestation, memory curtaining/protected execution, sealed storage, Trusted Third Party (TTP) support, and endorsement keys. The goal of these techniques is to provide security for the resources of the system by authenticating the validity of the endpoints of communication and providing guarantee for the integrity of running processes. Also, trusted computing has found uses in distributed firewalls, mobile third party computing, and preventing distributed Denial of Service (DOS) Attack [8]. In practical industrial applications, the key performance indicator- (KPI-) related prediction and diagnosis are quite important for the product quality and economic benefits. To meet these requirements, many advanced prediction and monitoring approaches have been developed which can be classified into model-based or data-driven techniques which should be noticed in the cloud framework [9–13].

In the following sections, some of the requirements to illustrate the proposed method including OpenID, One Time Password, and hardware-based activation are described.

1.1. OpenID. OpenID as a distributed open identity standard is used to identify users and allow them to use services of various relying parties (service of websites) by the use of an identifier that is similar to a URL. This URL is called Personal Identity Portal (PIP) URL which replaces the traditional username and password. An example of a PIP URL is http://david.pip.verisignlabs.com/, which is given to user David by Symantec site as an identity provider. This user can use his OpenID in any website that supports OpenID [14].

Furthermore, OpenID is considered to be a standard which facilitates single sign-on since the identity provider and the relaying party are not required to have a preset relationship. Generally, OpenID is an authentication and recovery system, but it can also be expanded to provide easy registration and attribute exchange. By using the URL typed OpenID, it has become easier for the users to have an account on sites as they do not need to provide their personal information in every website. Also, this URL provides the flexibility to share only the information that the user chooses to share with any relying party [15].

Browser web redirections are used to create communication between the relying party and the OpenID provider. OpenID makes use of query strings of HHTP or creates POST elements to denote different fields rather than use tokens constructed of security elements. In this way, the complexity is decreased, and also there is no need for parsing documents [16].

The operation process of OpenID presented in [17] is described here. This process consists of a user, an identity provider (IDP), and a relying party (RP) which operates as follows.

A user types the OpenID in a relying party site.

(1) The OpenID is confirmed by the RP and connection is created to the IDP.

(2) A request is sent to the IDP from RP to authenticate the user. This communication is done through the user agent.

(3) The IDP requests the user to provide the password which refers to OpenID.

(4) The user provides the password.

(5) The user's password and OpenID are confirmed by the IDP and the authentication process terminates.

(6) The IDP sends a token to RP through the user agent. This is to inform the RP about authentication process. After this process, the user can use the services provided by the relying party.

Using OpenID, as described above, can also result in some security problems. Some of these security concerns are as follows.

Eavesdropping attack: this protocol has a weakness against eavesdropping attack. That is, an eavesdropper can intercept a successful authentication assertion and reuse it if a nonce is not being checked.

Denial of Service Attack: another problem of OpenID is that a rogue relying party is able to launch DOS attack against OpenID provider because of the weaknesses in the OpenID protocol. The OP cannot rapidly determine if a request is genuine or not since the messages of OpenID protocol do not include any such information. Repeated authentication, associations, or signature verification requests made by the relying party can result in this situation.

Man in The Middle (MITM) Attack: generally, changing signed fields by MITM Attack can be prevented by using associations, but there are exceptions which happen during the discovery, association sessions, and direct verification. A compromised DNS will allow an attacker to impersonate an OP and issue associations or make decisions. In this case, the signatures of the messages are not enough anymore. Also, a tampering in the discovery process will allow the attacker to specify any OP, and no impersonation is required. Furthermore, even MITM Attack is not required if integrity of information in the discovery process is violated by changing XRDS document.

Phishing attack: a main issue in using OpenID is phishing attack. In such attacks, the phisher creates a fake RP which is very similar to the real one and directs the user to this fake page. If the user enters the owned OpenID, then s/he would be directed to another fake page related to the OpenID provider that asks for the user's password. If the user provides the required password, then the phisher would obtain the password. In this way, phisher has both the OpenID URL and

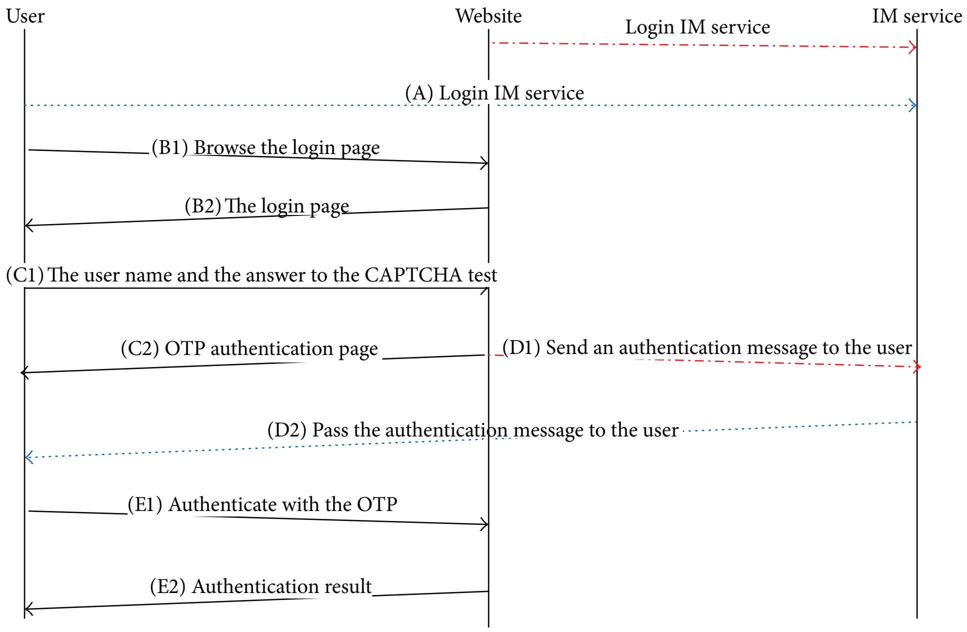
User
Website
IM service
Login IM service
(A) Login IM service
(B1) Browse the login page
(B2) The login page
(C1) The user name and the answer to the CAPTCHA test
(C2) OTP authentication page
(D1) Send an authentication message to the user
(D2) Pass the authentication message to the user
(E1) Authenticate with the OTP
(E2) Authentication result

Figure 1: A login process through OTP [27].

the password to use any RP services instead of the real owner of OpenID.

In the present research, the focus is to provide security against phishing among all other attacks.

1.2. One Time Password (OTP). Generally, there are various methods for strong multifactor authentication. These methods are based on hardware token, software token, smart card, RFID, steganography, watermarking, or biometrics [2, 18–26].

In the methods using hardware token, a dedicated device such as RSA secure ID token is used. But there are some difficulties in using this method since the distribution, management, and the installation and configuration of this device are complicated. Also, this method may not be very useful when several cloud applications, using various service providers, are set up. This is because these tokens are usually configured for a single application.

In software-token-based methods, SMS text message, Skype, smart phone app, E-mail, IM, and so forth, can be used to send a password (the OTP). Therefore, the authentication can be performed based on something that the user already possesses and no hardware token is required. Accordingly, this method has low overhead for management as there is no hardware token distribution and management. Also, since the service provider does not keep any seed, the transaction is more secure. Hence, a dedicated hardware token can be easily replaced by flexible software token.

Methods which are based on biometrics perform authentication are based on physical characteristics (fingerprint, palm print, voice, iris, etc.) of the user. Disadvantages of this method are limited portability, inflexibility (can be linked to a single application), and high overhead because of expensive deployment, configuration, and maintenance.

Based on these explanations, OTP is a kind of password that can be used through the hardware or software tokens for authenticating purposes.

Passwords form an important part of information and network security since they are keys to access private data. Thus, managing passwords to keep them safe and secure is very important. Users generally tend to create simple passwords and write them down to be able to remember them. Although there is a great deal of instructions as to how to create strong passwords, memorize them, keep them safe, and so forth, these passwords can still be stolen by various attacks such as phishing. Therefore, other methods are required to prevent misuse of sensitive information through stolen passwords.

One of these methods is to eliminate standard static password and use One Time Password as a replacement. OTP is a password that changes with every login of the user, unlike the static passwords which are changed only when they are forgotten or expired. Also, static passwords are saved on the hard drives of computers and servers which makes them targets of attacks, but since OTPs change for every user login, even storing the passwords does not have such harms.

A login process by use of OTP as stated in [27] is illustrated in Figure 1. The authentication message, which is passed to the user in step D2, includes the OTP generated for a user to be authenticated.

In general, OTPs are generated in two different ways. One way is time synchronized and the other is counter synchronized. In both ways, the OTP is generated by using an algorithm and a hardware owned by the user which is synchronized with a server.

Time synchronized OTPs are deployed broadly and they need the authentication server and user's hardware to be synchronized to produce a correct password in order to carry out user authentication. The produced password should be used by the user within a limited period of time; otherwise, it expires.

Counter synchronized OTPs are generated based on a synchronized counter between user's hardware and a server. For each OTP request from the user, the counter increases and an OTP is generated. Just like the time synchronized OTPs, for each login, the user must enter a currently generated OTP.

A type of OTP which is widely used in Europe for the authentication process of debit and credit cards is called Challenge Based OTP. It usually uses a hardware device. In this type, to obtain an OTP, the user should provide a known value, like a Personal Identification Number.

All the current OTP systems are based on a cryptographic process to make passwords from a synchronization factor (time or counter), a secret key, and sometimes a PIN. An example of such OTP generators is hash based which takes advantage of hashing algorithms in cryptography for password generation. Such systems have a one-way function with input parameters that include synchronization factor, secret key, and PIN, and its output is a fixed length OTP.

Using OTP can enforce higher security for enterprise resources, provide efficient fraud detection and prevention, and allow the use of versatile authentication methods. The OTP solutions can generally support a variety of authentication methods that can even be used simultaneously. Also, the OTPs can be sent in various ways such as E-mail, IM, Skype, and applications on smart phones, among which using a smart phone reduces the costs and complexity of using hardware OTP tokens while providing more flexibility. By using OTP, a dedicated hardware token can be replaced by flexible software token.

Although using OTP has several benefits, attackers can still use drawbacks of OTP to perform attacks. Generally, using social engineering against OTP users to achieve some of their previous OTPs is one of the attacks performed by phishers during recent years. There are examples of phishing attacks which have been successful to trick online bank customers and steal their previous OTPs. Also, the time synchronized OTPs can be vulnerable to phishing attacks. Such an attack occurs if an attacker can rapidly obtain the OTP in plaintext and use it before the legitimate user has used it. The other type of such attacks can happen when a phisher obtains some previous OTPs (those that are not valid any more) and predict the next OTP that will be generated for a user. In this type of attack if the OTP is generated by a pseudo random generator rather than a true random generator, the OTP will most probably be compromised. A true random generator to generate OTPs can only be used when the OTP is generated by the authenticator and sent to the user; if the OTP is generated by every party, then only a pseudo random generator can be used.

In case the user's computer is already compromised by a malware that can store whatever the user types through the keyboard (key logger), even the OTP method will fail and the attacker can easily achieve the required password.

1.3. Hardware-Based Activation. Activating software, as a part of licensing it, is the process that protects both the user's and the software developer's rights by ensuring that a genuine product is used by the user and is activated only on one specific computer.

One of the most common and secure protection options to activate software is using hardware-based activation base on modern industry and robust data-driven [12, 28]. By incorporating hardware-based activation code to activate software on a user's computer, the software binds to the computer, where it is going to be used. Therefore, this software cannot be copied to other devices. In this way, when a user installs an application on his/her computer, the application gets activated based on the hardware dependent code. This activation code just works on the computer that it is generated for and includes information about the application and the period that it will remain activated. For example, an Autodesk application uses various information such as HDD Serial, Network Adapter MAC Address, information about OS, and CPU information to generate an activation code based on its algorithm, and only by entering a valid code, the Autodesk application gets activated. Therefore, because of the specific properties and advantages of the hardware-based activation, it is used in the proposed security architecture.

1.4. Phishing and Pharming. Phishing can be considered a social engineering attack to gain sensitive information or access permissions (such as usernames, passwords, account numbers, etc.) by forging an entity and pretending to be a trustworthy and legitimate entity during an electronic communication. It is usually performed through email or instant messaging when the victims are directed to a forged website and encouraged to provide their sensitive information. As an example, banking websites are very common targets to be forged [3].

Pharming is another attack, designed to pass the traffic of a website to a forged one. Exploiting vulnerabilities of DNS server software and altering victim's computer host files are some ways of performing this attack. Also, an unprotected access is necessary to target a computer, like changing a customer's home computer instead of a corporate business server. Pharming is considered a serious threat to those who are hosting electronic commerce and banking websites.

Recently, both phishing and pharming attacks have been used to do online identity theft and get access to private information of people. To prevent these attacks different security measures should be incorporated such as technical measures, user training, and public awareness.

In this part we have explained some of the technical ways to prevent these attacks. Usually, these techniques are classified in two categories: list based or heuristic based. List-based techniques create a black or white list or even both and perform the filtering based on these lists. The black list is the most common method which lists the URL of phishing sites through user reports or found URLs by crawlers. Generally, the efficiency of list-based methods depends on how accurate and updated these lists are. However, the common problem in such methods is the false negative problem.

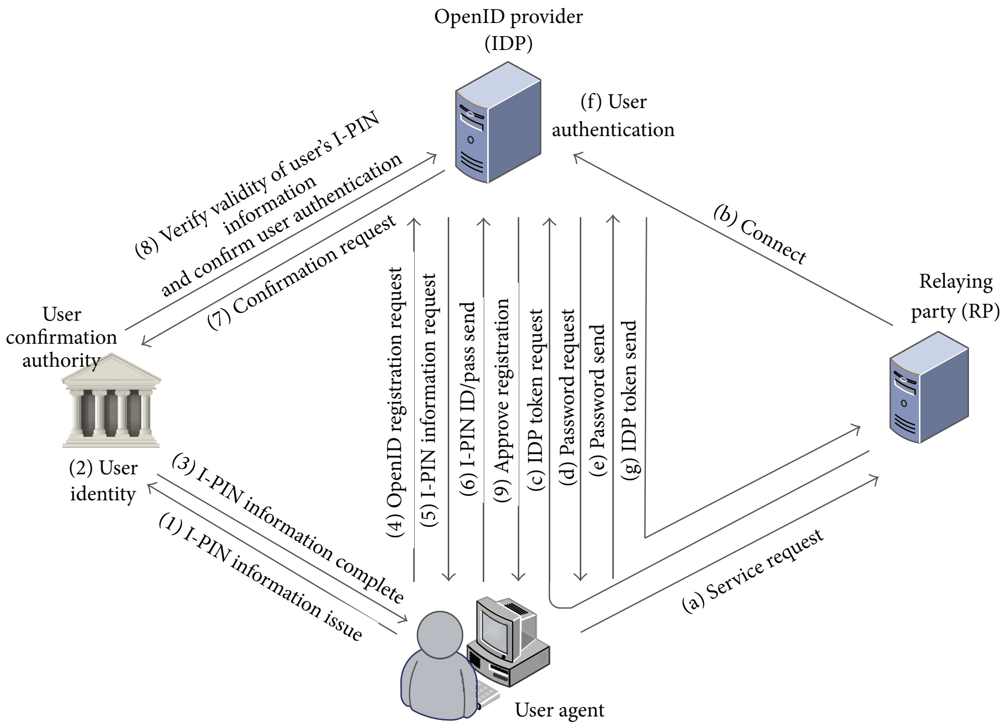

Figure 2: User authentication based on I-PIN [17].

In the other category (heuristic-based), various methods can be used to determine if a site is committing phishing or not. Some of these methods are the following.

(1) URL: in some methods phisher directs users to a fake site by adding @ character to a URL. In such cases the URL must be checked to insure that it does not contain any special sign or character.

(2) Input fields: generally, phishing sites ask for user's password, user name, credit card number, and so forth; therefore, they include various input fields which can help in detecting such sites.

(3) Domain name: in many cases a phishing site has a name similar to the target site. A method to determine such sites is to use metrics to measure the distance between strings. Some metrics that measure the distance are proposed in [29, 30]. The calculated distance can be used as an index to find phishing sites.

(4) Similarity between images: phishing site can also be discovered by checking the visual similarities. This can be done by both comparing the image files' hash values [31] or by comparing colour distribution in images [32, 33].

(5) Frequently used terms: identifying the frequently used words or sentences in a site and then submitting them to a search engine can result in finding the phishing sites. The technique proposed in [34] can be used to find the frequent terms in a site.

The heuristic-based approaches can use one of the above-mentioned methods or a combination of them to assess sites. Finally, it is necessary to mention that the list-based and heuristic-based techniques cannot determine all the phishing sites [35]. Thus, a better way to prevent phishing can be authenticating every user on the web.

1.5. Paper Organization. The rest of the paper is organized as follows: Section 2 includes the problem statement. In Section 3 some of the related works are explained. Section 4 explains the proposed trusted base model to mitigate identity theft. In Section 5 the analysis of the proposed model is presented based on some attacks. Finally, in Section 6, the content of this paper is reviewed and the conclusion is presented. In this paper, the authors tried to make the structure clear as their previous published papers [36–44].

2. Problem Statement

You and Jun in [17] have proposed a technique to strengthen OpenID authentication by use of Internet Personal Identification Number (I-PIN). This number is a unique number issued by a user confirmation authority to authenticate users without using their personal information and has been used in Korea.

Based on Figure 2, the process of requesting and obtaining an I-PIN is as follows.

(1) First, the user requests a trusted third party (Principal Confirmation Authority) to issue I-PIN with his or her name and resident number.

(2) Principal Confirmation Authority confirms identity with user's name and resident number, and then further confirmation is done through certificate

of authentication, credit card information, mobile phone, face, and so forth (one of these).

(3) Once user confirmation is done, Principal Confirmation Authority issues I-PIN for the user.

The generated I-PIN is later used for authenticating the user during the OpenID registration. Whenever this I-PIN is used, the Principal Confirmation Authority should verify the validity of the user's I-PIN information and confirm user authentication by sending verification result and principal confirmation information to the identity who has asked for user authentication.

The User Confirmation Authority which is an entity in the above process is considered to be a TTP. A TTP, as a renowned object used in cryptography, is an entity that assists the interactions among parties by reviewing all their critical transactions according to the ease of creating fake digital contents. The parties in this interaction rely on the third party, and based on the existing trust in this model, the relying parties secure their interactions.

In addition to digital cryptographic transactions, TTPs are common in any commercial digital transactions. An example of trusted third party is Certificate Authority that issues Digital Identity Certificate for a party in an interaction. Transactions that need a third party recordation would also need a kind of third party repository service. Using a TTP has some disadvantages as mentioned below.

(i) Creating, using, and maintaining TTPs are very costly. For example, in case of digital certificates, various issues should be considered such as integration between the existing software platform and tools, easiness in using tools, strength of the algorithms which are incorporated (level of security), flexibility, creation of a platform that accepts all the certificates, and so forth. Responding to all these factors and the creation and maintenance of such TTPs is very costly.

(ii) TTPs such as Certification Authorities, which issue certificates for various users, ask for certain amounts of money for services that they provide. Hence, taking subscription to these services and having multiple certificates for different purposes can be very costly.

(iii) There are limitations in the integration between an external Certification Authority and the infrastructure of any organization.

(iv) To configure, expand, and manage the certificates, there are flexibility issues.

Based on the above explanations, any of the disadvantages of TTPs can be considered as a drawback for models which include a TTP as an entity.

Also, in [27] a model is proposed which uses OTP to prevent phishing. In their proposed model, the integrity checking of the user's platform has been ignored since an instant message service is used (the application of which can be installed by any user). This is while integrity checking is a factor that should be considered for security model environment.

Furthermore, Madsen et al. in [45] discussed federated identity and indicated the benefits of federated identity management based on standards as follows.

(i) It permits and simplifies the processes used by federated organizations in terms of sharing user identity attributes.

(ii) It simplifies authentication and accessing permission using service access requirements.

Also, Madsen et al. illustrated some active problems and concerns in an FIM as follows:

(i) misuse of user identity information through SSO capability in SPs and IDPs;

(ii) user's identity theft;

(iii) trustworthiness of the user.

The above explanations highlight the existing OpenID authentication problems in [17, 27, 45]. Therefore, the aim of this study is to overcome these problems and drawbacks by proposing a secure OpenID model.

3. Related Works

Security of federated identity has become an interesting research area in the last few years and has been appealed by huge companies like IBM. Security concern in federated environment has been addressed by Huang et al. in [46]. They proposed an identity federation broker that introduced a trusted third party between SP and the IDP.

According to Rodriguez et al. study in [47], there are several different formations of identity management regarding ensuring access control in cloud computing environment which is named In-house, IDaaS. The users with In-house identity configuration are able to manage and issue their identity. If identity is configured and issued by outsource company, it is called identity as a service or IDaaS. IDaaS is divided into three categories which have been commercially offered in the market. Complete management, pseudonyms implementation, and independently IDaaS implementation are three configuration parts of IDaaS. Furthermore, the wide area of security via security guidance for critical areas of focus in cloud computing has been discussed by cloud security alliance [48].

Yeluri and Castro-Leon in [7] presented the concept of trusted clouds and also discussed the challenges of cloud security and compliance. In this study, the necessities of trusted clouds are argued. Furthermore, four usage models are introduced in order to enable a trusted computing infrastructure.

Ege in [49] explores the capabilities available to the mobile smartphone platforms to secure such participation and describes an architecture for adding trust management to the exchange of media to and from a smartphone user.

Ghazizadeh et al. in [3] suggested a model in order to solve identity theft in the cloud. This model incorporates trusted computing, Federated Identity Management, and OpenID Web SSO. This proposed model is evaluated through BLP confidential model, security analysing, and simulation.

Horsch et al. in [50] proposed the TrustID architecture and protocols. In their architecture, in order to store some context specific identities in a secure way in a mobile device, a secure element is used. Protocols are introduced to derive identities from a strong root identity to the secure element in the mobile phone in a secure way and also to utilize the newly derived IDs. A reliable smartphone operating system or a Trusted Execution Environment is not needed for these protocols. For this matter, a secure combined PIN entry mechanism for user authentication is included in the concept that prevents attacks even on a malicious device. The implementation is done utilizing a Samsung Galaxy SIII mobile phone using a microSD card SE. Also, the German identity card nPA is used as root identity to derive context-specific identities.

Ahmad et al. in [51] presented a scenario related to identity theft, tracking, and illegitimate information gathering as the first step for identity theft which hackers named fingerprint. Besides, they illustrated the essential problems of lack of platform trust in platforms encompassing in federated systems and also discussed the significances of respective threats on them. Finally, they proposed a user requirement model including core issues for federated identities.

In [16] Urueña et al. discuss about the privacy risks for the users of OpenID and Facebook Connect as two famous single sign-on platforms for web-based content access. In this study the authors provide a very detailed explanation on privacy vulnerability of the OpenID authentication protocol. They discussed how the unique OpenID identifiers of the users are leaked to third parties by OpenID agents, which is a real and well-known privacy risk for OpenID users. Also, the privacy of Facebook Connect (proprietary single sign-on platform, which has become famous lately) is analysed and is concluded that it does not suffer the same vulnerability but there are some other important privacy issues. In order to overcome these problems, the authors propose three solutions; a long-term solution to overcome the root cause of the vulnerability and two short-term mitigations.

Guenane et al. in [52] propose a strong hybrid cloud based firewalling authentication architecture using EAP-TLS Smartcards which delivers identification and authentication of every element of the hybrid cloud-based firewalling services. In this study, the authors propose a central authentication server that keeps all necessary information about a virtual firewall. They think that in this way the problem of considering the virtual firewall in the authentication model is manageable.

Vincent et al. in [53] present the Trusted Identity Module (TIM), a local smartphone module that enables the user to log into application using the newly proposed OpenID Connect protocol. An active cardlet mounted on a secure element like the SIM/UICC of the mobile phone to store long-lived tokens and run cryptographic operations is used in TIM. The authors argue that their TIM solution improves usability, security, and privacy protection for the user while it has few impacts on the emerging OpenID Connect protocol.

Wang et al. in [54] reported their extensive security study on commercial web SSO systems. Their study showed that there are some critical logic flaws in SSO system, which can be discovered from browser relayed messages and can be exploited potentially, even without access to the source code or other insider knowledge of these systems. They have studied and analysed practical steps that attackers might take on commercial systems and the ways to detect these kinds of attacks. Attackers can attack and exploit the vulnerabilities in detection flaw to sign in as the victim. They had exposed new failure in web SSO systems and had highlighted the dreadful need to enhance the security of SSO community.

Security analysis of three commonly available SSOs, which include Microsoft Passport, OpenID 2.0, and SAML 2.0, has been performed by Wang in [55]. He highlighted some vulnerabilities and security issues for each system with their applications and analysed Privacy Aware Identity Management and Authentication for the Web as two alternative solutions for SSOs as well.

Rodriguez et al. in [47] argued in their study that there are some difficulties in digital identity. They focused on Federated Identity Architecture (FIA) and analysed some of the problems related to it. In addition, they explored industrial FIA solutions and investigated security and privacy issues and other challenges. Besides, Yan et al. in [56] proposed a cryptography based federated identity with some desirable features to adapt with cloud computing. They harmonized hierarchical identity-based cryptography with federated identity management in the cloud environment.

OpenID in comparison with Security Assertion Mark-Up Language (SAML) is authentication exchange protocol for identity management in the Internet, but SAML is designed for limited or small scale identity management, and also OpenID is much easier to be deployed and implemented. SAML's parties are based on trust while the parties in OpenID basically trust on DNS system to find the address of IDP and rely on it at any time is called. Therefore, DNS cache poisoning and DNS hijack are common impersonation attacks in OpenID environment [57].

Feng et al. in [58] introduce existing antiphishing methods and put forward a new method. In the new method, two types of passwords are constructed which are fixed password and temporary password. The fixed password is the only one that is used solely on binding PCs. It is believed that it is more safe, suitable, and convenient for users to use OpenID on several fixed PCs.

An in-depth analysis of 14 major SAML frameworks has been explained by Somorovsky et al. in [59]. They specified that 11 of 14 SAML frameworks, such as IBM XS40, Shibboleth, and Salesforce, have critical XML signature wrapping (XSW) weaknesses. On the basis of their analysis, an automated penetration testing tool for XSW in SAML frameworks has been developed. They proposed a framework based on the flow of information between two RP's components, in order to analyze such kinds of attacks. Unexpectedly, practical and efficient countermeasures have been returned by their analysis.

Leandro et al. in [60] proposed a multitenancy authorization system using Shibboleth for cloud-based environments. The main idea of their proposed model is to demonstrate how an organization can use Shibboleth as a base to practically implement a system of access control in a cloud computing

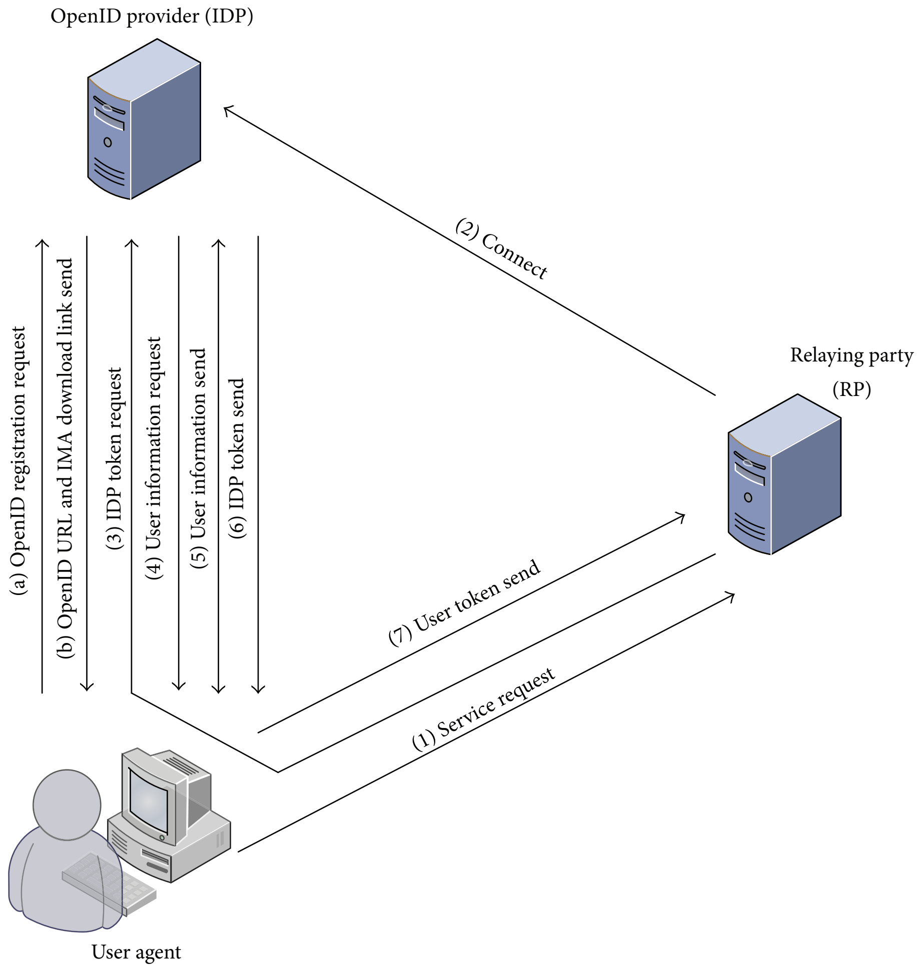

Figure 3: The proposed model.

environment without a trusted third party. Because Shibboleth is based on SAML and it means Shibboleth is compatible with international standards, it can ensure interoperability.

The technique in which trust is expanded is over and done with an open authentication standard called OAuth. Sun in [61] contended that he did not find any new threat by using formal methods to examine security and privacy of OAuth protocol. He explored possible security threats used by three main IDPs such as Microsoft, Facebook, and Google, to appreciate its implementation in real-world settings. He found numerous methodical weaknesses that permit an attacker to have unauthorized access to a user's profile, which opens up possibility of impersonating the victim on the supporting website. He proposed a method for relying parties to use server-flow every time possible and protect the authenticity and confidentiality of SSO credentials [61].

Kim in [62] illustrated the step by step OAuth authentication. He proposed the use of card space (which is a self-issued card) for user to log into IDP. Since self-issued cards are used in place of usernames and passwords, it will prevent identity theft. The system employs public key cryptography and generates dissimilar keys for each site the user visits. For example, even if an Evil Scooper is used, it will not expose anything at all.

Hwang and Li in [63] proposed data colouring and software watermarking techniques to protect common data objects and enormously distributed software modules. They found that trust and security are two issues in cloud businesses for completely security compliant cloud platforms. They focused on virtualization's security to gain trust environment in the cloud. They recommended using a trust-overlay network over multiple data centres to implement a reputation system to establish trust between service providers and data owners. Finally, they showed a Cyber trust and privacy in the cloud that are entirely based on cloud layers and deployments.

4. Proposed Model

In this proposed solution, first, we introduce existing parties and then some assumptions that are incorporated.

As shown in Figures 3 and 4, this method includes three parties which are user agent, relying party (RP), and identity provider (IDP). To use the OpenID services and to be registered as an OpenID user, the user should have a computer system which has TPM hardware, a vTPM, or a TPM emulator. One of the benefits of using TPM is to guarantee the integrity of the user's frameworks. Also in

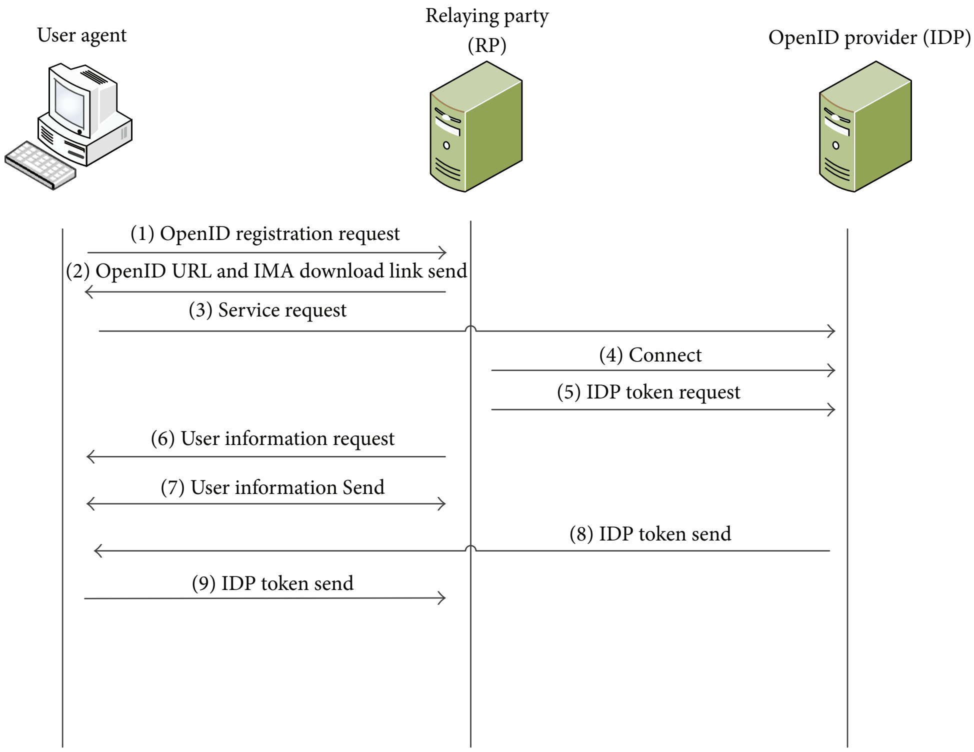

Figure 4: Data flow of the proposed model.

our scenario, we have assumed that the communications are taking place in public cloud. In case of migrating to private cloud, the users who are registered in the IDP database (according to the request of their organization), based on their TPM hardware, can perform the OpenID registration and achieve the OpenID. Moreover, to perform an authentication process by using OTP, an application that receives the produced OTP is used. We name this application Instant Message Application (IMA). The IMA will be provided by IDP for all the users who have successfully registered for an OpenID. This application gets activated by using the user's TPM hardware and receives the OTP which is sent by the IDP (the whole process is discussed in the proposed method steps).

(a) User Agent Requests OpenID. User agent requests for a PIP account and sends his/her registration request by providing a unique username, password, TPM key, security message, and personal contact information (address, email address, etc.). Therefore, the user will be registered as an OpenID user to have a unique OpenID URL.

Also, an automated public Turing test (CAPTCHA) is included in the registration process before registering the user to receive an OpenID. This test is to ensure that the response is generated by a human being.

(b) IDP Sends OpenID URL and IMA Download Link to the User. IDP creates an OpenID URL for the user and sends it to him/her. Also, an IMA download link is sent to the user. The user receives the OpenID and the IMA download link then s/he downloads and installs the IMA.

Also in this scenario, we assume that both the user and IDP have their own TPM public key. They send encrypted data by the use of their own public key and TPM can decrypt this data through its private key.

After installing the IMA, it should be activated. This activation process includes verification of the product key (serial key) and the TPM hardware key for IMA product, and also licensing the product to run only on the specific system that has been verified by the IDP during the registration process.

The activation code is produced based on a pair of codes (computer code, product code). The first code which is the computer code identifies the user's computer. In our scenario, the IMA uses the TPM hardware of the user's computer (for those systems that do not have TPM, the vTPM or TPM emulator can be used) and then generates the computer code through its own algorithm. An advantage of this hardware-based method is that there is no need to keep this string in secret because of the TPM's specifications.

This activation process is automatically performed for the first time that the IMA is launched when the user's computer is connected to the Internet. Processes (a) and (b) are only performed for the first time when the user requests OpenID from IDP. Other steps of the process are as follows.

(1) User Agent Requests Service from RP. User enters the owned URL typed OpenID in the login page of RP to use the provided Internet services. Through this URL the user is able to sign in to any website that supports the OpenID services.

(2) RP Connects to IDP. RP confirms the received OpenID from the user and connects to IDP who has provided

the user's OpenID. In the OpenID Authentication version 2.0, the IDP and RP establish an association based on their TPM's public and private keys and it could be an optional step.

(3) RP Requests Token from IDP through User Agent (Redirection for User Authentication). In this step, the user's location is located and the authentication token is created by RP. RP requests the IDP to authenticate the user to prove that s/he is who he/she claims to be. For this reason, after getting the RP's request, the browser performs the next step. The browser uses the Security Assertion Mark-Up Language (SAML) protocol to continue the token exchange.

(4) IDP Requests User Information for Authentication (IDP Sends Security Message and Requests for OTP and Username). IDP requests the user to provide some information through a request page. This request page includes some fields to be filled and one field to be approved by the user. The included fields are username, password (OTP), and secure message. In the user name field, the user should enter his/her username which is set during the registration process. The OTP field should be filled by the user, based on the OTP that he/she receives through the installed IMA. The secure message field is already filled by IDP and should show the secure message that the user has defined during his/her registration in first step. Thus, based on this secure message, the user can determine whether the provided request page is a phishing page or not.

(5) User Agent Sends the Required Information (User Agent Sends the Username and OTP). This step is the most critical part of our proposed OpenID based on IMA. In our environment we have assumed that IDP can collude with RP to connect user alliance and collect user's behaviours. Also, we have supposed that there is no Privacy Enhancing Technology (PET) organized in our cloud environment. User's browser, RP, and IDP must prove their identity based on mutual attestation process using their TPM-enabled platforms. We have assumed that the IDP is trustworthy and authorized by all participating parties in the Federated Identity Architecture. In this scenario, IDP is the attester that sends the challenge for the attested user to check their integrity.

After checking the integrity, the user must fill in the provided fields and check the provided secure message. Therefore, first, the user should compare the received secure message with the one that s/he has provided during the registration process. If the shown secure message is the same as the one already defined, the user should approve it and continue to provide the required information; the user must enter his/her registered username and then fill in the password field by the OTP that IDP has generated and sent it to the user through his/her IMA. If the secure message is not the same as the one already defined by the user in the registration process, the user must not enter his/her username and OTP since the requested page can be a fake page.

(6) IDP Sends Token to User Agent (Authorizing User's Browser for Further Requests). After insuring about a successful mutual attestation process, that is, the one that IDP and the user have confidence in one another, the IDP will deliver SAML token to the user's browser. Nowadays, some IDPs such as Facebook, IBM, and Google use SAML token in order to deliver their users access rights. For future work, we may consider other mechanisms.

(7) User Agent Sends Token to RP. In this step, user puts more confidence on the RP to get a service based on the SAML token. IDP sends an encrypt token by the user's public key that shows IDP is legitimated and verified by a trusted authority. User decrypts the token by his/her private key. Finally, the user uses the token to get more services from RP.

5. Security Analysis

In this section, we examine the strength of the proposed solution in terms of security and consider possible improvements.

5.1. Mutual Authentication. Mutual authentication, also called two-way authentication, is a process or technology in which both entities in a communication link authenticate each other. In our proposed model, IDP authenticates an OpenID user by asking him/her to input the user name and OTP password. Besides, OpenID user, using public and private keys of the TPM (vTPM) verifies the integrity of the IDP which is only known to the user. Other users cannot guess the correct TPM (vTPM) keys because of the TPM (vTPM) characteristics. Therefore, the user authenticates the IDP and deals with it, and at the same time, IDP deals with the verified user. We assume that an attacker can send the users to the fake IDP; thus, the fake IDP cannot decrypt the end user information by the fake TPM keys. The absence of a TPM will cause authentication of the OpenID to fail.

5.2. Compromising the IMA. The strength of the proposed model lies in TPM (vTPM), which is considered as a secure replacement for certificates since it is difficult to compromise its components. The attacker is assumed to call TPM (vTPM) commands without bounds and without knowing the TPM (vTPM) root key, expecting to obtain or replace the user key. The analysis goal in TPM (vTPM) study is to guarantee the corresponding property of IDP and the user [64, 65]. Also, the overall aim of the proposed model is verifying the IMA. Therefore, to compromise the solution, an attacker must at least know vTPM content and then target the components accordingly.

An essential axiom is that TPM (vTPM) is bound to one and only one platform and because of this reason it has been used in this study to check the integrity. In [66] Black Hat shows how one TPM could be physically compromised to gain access to the secrets stored inside it. But launching this attack requires physical possession of the PC and needs someone with specialized equipment, to intimate the knowledge of semiconductor design, and advanced skills. Thus, Microsoft believes that using a TPM is still an effective means to help protect sensitive information and accordingly takes advantage of a TPM.

While the abovementioned attack is interesting, these methods are difficult to duplicate and, as a result, pose a very low risk in our proposed model. Furthermore, IDP asks the user his/her credentials to gain security assurance. As a result, an attacker must not only be able to retrieve the appropriate secret from TPM (or vTPM), but also find the user credentials (user name and OTP password). If the credentials are sufficiently complex, this poses a hard, if not intractable, problem to solve in order to obtain the required key to phishing attack in OpenID environment.

5.3. Authentication in an Untrustworthy Environment. Sometimes users have to sign into their web accounts in an untrustworthy environment, for example, accessing a credit card account using a public Internet at university, on a shared computer or a pervasive environment. Our solution is also applicable to such cases.

In this case, the user must sign into his/her OpenID account via the trusted device and then just follow the OpenID login instructions. Our proposed model requires integrity of the trusted device for each authentication, regardless of cookies.

Because of using the TPM and the trusted device, the user can read the authentication information from the device and then log into the website on the untrustworthy environment.

Khiabani et al. [64] argue that pervasive systems are weaving themselves in our daily life, making it possible to collect user information invisibly and in an unobtrusive manner even by unknown parties. Therefore, OpenID as a security activity would be a major issue in these environments. The huge number of interactions between users and pervasive devices necessitates a comprehensive trust model which unifies different trust factors like context, recommendation, and history to calculate precisely the trust level of each party. Trusted computing enables effective solutions to verify the trustworthiness of computing platforms in untrustworthy environment.

5.4. Insider Attack. Client's weak password or server secret key stored in server side is vulnerable to any insider who has access to the server. Thus, in the event that this information is exposed, the insider is able to impersonate either party. The strength of our proposed model is that TPM key is the essential part in the trusted OpenID model and IDP stores TPM database with its TPM key which cannot be accessed by attackers. Therefore, our scheme can prevent the insiders from stealing sensitive authentication information.

5.5. The Man in the Middle (MITM) Attack. In the Man in the Middle Attack, a malicious user located between two communication devices can monitor or record all messages exchanged between the two devices. Suppose an attacker tries to launch an MITM Attack on a user and a website and that the attacker can monitor all messages sent to or received by the user.

The idea of making conventional phishing, pharming, and MITM Attacks concerns private users who are not usually connected to a well-configured network. Furthermore, private users often administrate their computers by themselves. Using Public Key Infrastructures (PKI), stronger mutual authentications such as secret keys and passwords, latency examination, second channel verification, and One Time Passwords are some ways which have been introduced to prevent MITM in the network area.

Table 1: Comparative analysis.

Title	Insider attack	MITM	Phishing attacks	DNS poisoning
[68]			*	*
[17]			*	*
[58]		*	*	
[69]			*	
[70]		*	*	
[71]		*		
[72]		*	*	
[27]	*		*	
[73]			*	
[60]	*			
[74]			*	*
Secured OpenID model (our proposed model)	*	*	*	*

Mat Nor et al. in [67] asserted that many security measures have been implemented to prevent MITM attacks such as Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocol, while adversaries have come out with a new variant of MITM attack which is known as the Man In The Browser (MITB) attack. This attack attempts to manipulate the information between a user and a browser and is much harder to detect due to its nature.

Also, trust relationship between interacting platforms has become a major element in increasing the user confidence while dealing with Internet transactions, especially in online banking and electronic commerce. Therefore, in our proposed model, in order to ensure the validity of the integrity measurement from the genuine TPM (vTPM), the Attestation Identity Key (AIK) is used to sign the integrity measurement. AIK is an asymmetric key and is derived from the unique endorsement key (EK) certified by its manufacturer which can identify the TPM identity and represent the Certificate Authority role against MITM attacks. Besides, in our proposed model, strong and efficient OTP has been utilized against MITM, which is one of the sufficient ways to mitigate identity theft and MITM attack.

6. Discussion

In comparison to other OpenID, other approaches are heavy weight protocols. In particular, SSL Certificate and secure channel are some of the requirements which OpenID users and cloud users need to install and prepare before authentication. To summarize all approaches, trust, OTP, and hardware authentication are influenced by endorsement and industrial company. However, none of the existing researches give a unified and common deliberation on all attacks that influence the confidentiality of the OpenID authentication. Table 1

Table 2: Feature comparison of existing approaches.

Method	Login on any PC	Security	Price	Trust
My OpenID personal icon	Own	Safe	Inexpensive	Medium
VeriSign and IE	Any	Safe	Expensive	Low
VeriSign and Firefox	Own	Safe	Expensive	Medium
Videoop	Any	Not safe	Inexpensive	Low
Jobber's by SMS	Any	Not safe	Inexpensive	Low
Feng's proposed model	Any	safe	Inexpensive	medium
Secured OpenID model (our proposed model)	Any	Safe	Inexpensive	High

summarizes this study of OpenID models based on the attacks which are considered in the current proposed model.

Feng et al. in [58] conducted a study on existing antiphishing approaches which cannot be used on any other system and they should be used on the user's system such as seatbelt and personal icon. Some of these approaches use certificates that are expensive, such as VeriSign's certificate, and they are not suitable for a personal system.

In Table 2, a detailed comparison between our proposed model and their models is shown.

This table shows that our proposed model can be used with any system because, as mentioned before, user can use vTPM or TPM emulator. Also, Secured OpenID is safe because of the trust characteristic which insures that the user's PC and the user agent integrity are unaffected by any malicious software and it behaves in the predictable way for the intended purpose. TPM is not expensive and as it has been mentioned it is a built-in equipment on the PC's motherboard. The most important part is trust which has been mentioned by the TCG and it has been recognised that the most important feature of the TPM is trust.

7. Conclusion

Trusted computing and multitenancy have the potential to solve trust and security concerns in a federated environment. We have presented the concept of using hardware-based activation, OTP, OpenID Web SSO, trusted computing, and federated identity management to solve identity theft in the cloud. The novelty lies in the fact that this research combines the OTP, OpenID, and hardware-based activation and finally adopts these novel technologies to propose a new secure SSO authentication model. A nexus of trusted computing, cloud computing, and Federated Identity provided by the model is assumed to be the study contribution enhancing security and privacy of cloud computing. Trusted and secure identities and efficient management of these identities, while users' privacy is protected, are a key element for the successful adoption of any cloud solution.

We presented advantages and disadvantages of OpenID, TTP, OTP, and trusted computing. Also, we presented the related work on this area, and besides, we stated the problems. Next, a proposed architecture for federated identity management based on trusted computing and multitenancy was offered to mitigate identity theft in the cloud. This will enable us to have more trust-based relationships amongst users, infrastructure components, and providers. This will also enable the enforcement of security, trust, and privacy policies for individual users, RPs, and IDPs. While the main aim is to mitigate identity theft, we can also extend this model to cater for other concerns in federated identity management. Finally, we analysed the security issues of the proposed model.

The future work will involve the development of a prototype of the proposed system for cloud computing and testing it for diverse real-world scenarios. The goal is to prove effectiveness of the proposed privacy and identity management system, as well as its potential to become a standard for privacy and identity management in the cloud computing.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] M. Yildiz, J. Abawajy, T. Ercan, and A. Bernoth, "A layered security approach for cloud computing infrastructure," in *Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN '09)*, pp. 763–767, December 2009.

[2] M. Alizadeh, W. H. Hassan, M. Zamani, S. Karamizadeh, and E. Ghazizadeh, "Implementation and evaluation of lightweight encryption algorithms suitable for RFID," *Journal of Next Generation Information Technology*, vol. 4, no. 1, pp. 65–77, 2013.

[3] E. Ghazizadeh, M. Zamani, J.-L. Ab Manan, and M. Alizadeh, "Trusted computing strengthens cloud authentication," *The Scientific World Journal*, vol. 2014, Article ID 260187, 17 pages, 2014.

[4] E. Ghazizadeh, M. Zamani, J.-L. Ab Manan, R. Khaleghparast, and A. Taherian, "A trust based model for federated identity architecture to mitigate identity theft," in *Proceedings of the 7th International Conference for Internet Technology and Secured Transactions (ICITST '12)*, pp. 376–381, December 2012.

[5] E. Ghazizadeh, M. Zamani, J.-L. Ab Manan, and A. Pashang, "A survey on security issues of federated identity in the cloud computing," in *Proceedings of the 4th IEEE International Conference on Cloud Computing Technology and Science (CloudCom '12)*, pp. 562–565, December 2012.

[6] I. M. Khalil, A. Khreishah, and M. Azeem, "Cloud computing security: a survey," *Computers*, vol. 3, pp. 1–35, 2014.

[7] R. Yeluri and E. Castro-Leon, "The trusted cloud: addressing security and compliance," in *Building the Infrastructure for Cloud Security*, pp. 19–36, Springer, 2014.

[8] M. MacKay, T. Baker, and A. Al-Yasiri, "Security-oriented cloud computing platform for critical infrastructures," *Computer Law and Security Review*, vol. 28, no. 6, pp. 679–686, 2012.

[9] S. Yin, G. Wang, and X. Yang, "Robust PLS approach for KPI-related prediction and diagnosis against outliers and missing data," *International Journal of Systems Science*, vol. 45, no. 7, pp. 1–8, 2014.

[10] B. Huang and G. Yu, "Research and application of personalized modeling based on individual interest in mining," *Abstract and Applied Analysis*, vol. 2014, Article ID 514295, 8 pages, 2014.

[11] S. Yin, X. Yang, and H. R. Karimi, "Data-driven adaptive observer for fault diagnosis," *Mathematical Problems in Engineering*, vol. 2012, Article ID 832836, 21 pages, 2012.

[12] S. Yin, G. Wang, and H. R. Karimi, "Data-driven design of robust fault detection system for wind turbines," *Mechatronics*, vol. 24, no. 4, pp. 298–306, 2014.

[13] S. Yin, S. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 11, pp. 6418–6428, 2014.

[14] I. Khalil, A. Khreishah, and M. Azeem, "Consolidated Identity Management System for secure mobile cloud computing," *Computer Networks*, vol. 65, pp. 99–110, 2014.

[15] G. Dólera Tormo, F. Gómez Mármol, and G. Martínez Pérez, "Towards the integration of reputation management in OpenID," *Computer Standards and Interfaces*, vol. 36, no. 3, pp. 438–453, 2014.

[16] M. Urueña, A. Muñoz, and D. Larrabeiti, "Analysis of privacy vulnerabilities in single sign-on mechanisms for multimedia websites," *Multimedia Tools and Applications*, vol. 68, no. 1, pp. 159–176, 2014.

[17] J.-H. You and M.-S. Jun, "A mechanism to prevent RP phishing in OpenID system," in *Proceedings of the 9th IEEE/ACIS International Conference on Computer and Information Science (ICIS '10)*, pp. 876–880, August 2010.

[18] M. Gharooni, M. Zamani, M. Mansourizadeh, and S. Abdullah, "A confidential RFID model to prevent unauthorized access," in *Proceedings of the 5th International Conference on Application of Information and Communication Technologies*, pp. 1–5, October 2011.

[19] H. Taherdoost, M. Zamani, and M. Namayandeh, "Study of smart card technology and probe user awareness about it: a case study of middle eastern students," in *Proceedings of the 2nd IEEE International Conference on Computer Science and Information Technology (ICCSIT '09)*, pp. 334–338, August 2009.

[20] A. A. J. Altaay, S. B. Sahib, and M. Zamani, "An introduction to image steganography techniques," in *Proceedings of the International Conference on Advanced Computer Science Applications and Technologies (ACSAT '12)*, pp. 122–126, Kuala Lumpur, Malaysia, November 2012.

[21] M. Zamani, A. A. Manaf, and R. Ahmad, "Knots of substitution techniques of audio steganography," in *Proceedings of the International Conference on Telecom Technology and Applications*, pp. 415–419, 2009.

[22] M. Zamani, H. Taherdoost, A. A. Manaf, R. B. Ahmad, and A. M. Zeki, "Robust audio steganography via genetic algorithm," in *Proceedings of the International Conference on Information and Communication Technologies (ICICT '09)*, pp. 149–154, August 2009.

[23] Z. Huang, S. Yin, and H. R. Karimi, "Residual generator-based controller design via process measurements," *Mathematical Problems in Engineering*, vol. 2014, Article ID 290371, 8 pages, 2014.

[24] Y. Xu, S. Yin, J. Yu, and H. R. Karimi, "Design of a TFT-LCD based digital automobile instrument," *Mathematical Problems in Engineering*, vol. 2014, Article ID 549790, 8 pages, 2014.

[25] J. You, S. Yin, and H. R. Karimi, "Filtering for discrete fuzzy stochastic time-delay systems with sensor saturation," *Mathematical Problems in Engineering*, vol. 2013, Article ID 146325, 10 pages, 2013.

[26] J. You, S. Yin, and Z. Yu, "Robust estimation for discrete time-delay Markov jump systems with sensor non-linearity and missing measurements," *IET Control Theory & Applications*, vol. 8, no. 5, pp. 330–337, 2014.

[27] C.-Y. Huang, S.-P. Ma, and K.-T. Chen, "Using one-time passwords to prevent password phishing attacks," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1292–1301, 2011.

[28] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: an overview," *IEEE Transactions on Industrial Electronics*, 2014.

[29] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443–453, 1970.

[30] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, 1981.

[31] R. Venkatesan, S.-M. Koon, M. H. Jakubowski, and P. Moulin, "Robust image hashing," in *Proceedings of the International Conference on Image Processing (ICIP '00)*, pp. 664–666, September 2000.

[32] K. T. Chen, J. Y. Chen, C. R. Huang, and C. S. Chen, "Fighting phishing with discriminative keypoint features," *IEEE Internet Computing*, vol. 13, no. 3, pp. 56–63, 2009.

[33] C. Y. Huang, S. P. Ma, W. L. Yeh, C. Y. Lin, and C. T. Liu, "Mitigate web phishing using site signatures," in *Proceedings of the IEEE Region 10th Conference (TENCON '10)*, pp. 803–808, November 2010.

[34] Y. Zhang, J. I. Hong, and L. F. Cranor, "Cantina: a content-based approach to detecting phishing web sites," in *Proceedings of the 16th International Conference World Wide Web (WWW '07)*, pp. 639–648, May 2007.

[35] L. Cranor, S. Egelman, J. Hong, and Y. Zhang, "Phinding phish: an evaluation of anti-phishing toolbars," Tech. Rep., CyLab, Carnegie Mellon University, 2006.

[36] M. Alizadeh, J. Shayan, M. Zamani, and T. Khodadadi, "Code analysis of lightweight encryption algorithms using in RFID systems to improve cipher performance," in *Proceedings of the IEEE Conference on Open Systems (ICOS '12)*, pp. 1–6, Kuala Lumpur, Malaysia, October 2012.

[37] E. Amiri, H. Keshavarz, M. Alizadeh, M. Zamani, and T. Khodadadi, "Energy efficient routing in wireless sensor networks based on fuzzy ant colony optimization," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 768936, 17 pages, 2014.

[38] F. Arab, H. Selamat, and M. Zamani, "An overview of success factors for CRM," in *Proceedings of the 2nd IEEE International Conference on Information and Financial Engineering (ICIFE '10)*, pp. 702–705, Chongqing, China, September 2010.

[39] F. Arab, H. Selamat, S. Ibrahim, and M. Zamani, "A survey of success factors for CRM," in *Proceedings of the International Conference on Computer Science and Applications*, San Francisco, Calif, USA, 2010.

[40] H. Taherdoost, A. Forghani, N. Jalaliyoon, M. Zamani, and M. Namayandeh, "Adoption framework expansion based on the computer ethics' related research models and ethical scenarios analysis," in *Proceedings of the International Conference on Economics Business and Management*, Manila, Philippines, 2010.

[41] S. Yazdanpanah, S. Shojae Chaeikar, M. Zamani, and R. Kourdi, "Security features comparison of master key and IKM cryptographic key management for researchers and developers," in *Proceedings of the 3rd International Conference on Software Technology and Engineering*, Kuala Lumpur, Malaysia, 2011.

[42] M. Zamani, A. B. A. Manaf, R. B. Ahmad, F. Jaryani, S. S. Chaeikar, and H. R. Zeidanloo, "Genetic audio watermarking," *Communications in Computer and Information Science*, vol. 70, pp. 514–517, 2010.

[43] M. Zamani, A. B. A. Manaf, R. B. Ahmad, A. M. Zeki, and P. Magalingam, "A novel approach for audio watermarking," in *Proceedings of the 5th International Conference on Information Assurance and Security*, pp. 83–86, Xian, China, September 2009.

[44] M. Zamani, R. B. Ahmad, A. B. A. Manaf, and A. M. Zeki, "An approach to improve the robustness of substitution techniques of audio steganography," in *Proceedings of the 2nd IEEE International Conference on Computer Science and Information Technology (ICCSIT '09)*, pp. 5–9, IEEE, Beijing, China, August 2009.

[45] P. Madsen, Y. Koga, and K. Takahashi, "Federated identity management for protecting users from ID theft," in *Proceedings of the Workshop on Digital Identity Management*, pp. 77–83, 2005.

[46] H. Y. Huang, B. Wang, X. X. Liu, and J. M. Xu, "Identity federation broker for service cloud," in *Proceedings of the International Conference on Service Sciences (ICSS '10)*, pp. 115–120, Hangzhou, China, May 2010.

[47] U. F. Rodriguez, M. Laurent-Maknavicius, and J. Incera-Dieguez, *Federated Identity Architectures*, 2006.

[48] D. C. J. Archer, N. Puhlmann, A. Boehme, P. Kurtz, and J. Reavis, "Security guidance for critical areas of focus in cloud computing v3.0," Cloud Security Alliance, 2011.

[49] R. K. Ege, "Secure trust management for mobile platforms," in *Proceedings of the International Conference on Computing, Networking and Communications (ICNC "14)*, pp. 381–385, February 2014.

[50] J. Horsch, K. Böttinger, M. Weiß, S. Wessel, and F. Stumpf, "TrustID: Trustworthy identities for untrusted mobile devices," in *Proceedings of the 4th ACM Conference on Data and Application Security and Privacy (CODASPY '14)*, pp. 281–288, March 2014.

[51] Z. Ahmad, J. L. Ab Manan, and S. Sulaiman, "User requirement model for federated identities threats," in *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE "10)*, pp. V6317–V6321, August 2010.

[52] F. Guenane, M. Nogueira, and G. Pujolle, "Strong hybrid cloud-based firewalling authentication using EAP-TLS smart-cards," in *Proceedings of the 1st Workshop of Mobile applications, Secure Elements and Near Field Communication*, pp. 1–4, Gainesville, Fla, USA, February 2014.

[53] J. Vincent, S. Kale, and V. Frey, "User-Centric Identity Management Using Trusted Identity Module-Binding Mobile Phone Secure Elements to the OpenID Connect Protocol".

[54] R. Wang, S. Chen, and X. Wang, "Signing me onto your accounts through Facebook and Google: a traffic-guided security study of commercially deployed single-sign-on web services," in *Proceedings of the 33rd IEEE Symposium on Security and Privacy (S and P '12)*, pp. 365–379, San Francisco, Calif, USA, May 2012.

[55] S. Wang, *An Analysis of Web Single Sign-On*, 2011.

[56] L. Yan, C. Rong, and G. Zhao, "Strengthen cloud computing security with federal identity management using hierarchical identity-based cryptography," in *Proceedings of the 1st International Conference on Cloud Computing (CloudCom '09)*, pp. 167–177, Beijing, China, December 2009.

[57] J. Jiang, H. Duan, T. Lin, F. Qin, and H. Zhang, "A federated identity management system with centralized trust and unified Single Sign-On," in *Proceedings of the 6th International ICST Conference on Communications and Networking in China (CHINACOM '11)*, pp. 785–789, Harbin, China, August 2011.

[58] Q. Feng, K.-K. Tseng, J.-S. Pan, P. Cheng, and C. Chen, "New anti-phishing method with two types of passwords in OpenID system," in *Proceedings of the 5th International Conference on Genetic and Evolutionary Computing (ICGEC '11)*, pp. 69–72, Xiamen, China, September 2011.

[59] J. Somorovsky, A. Mayer, J. Schwenk, M. Kampmann, and M. Jensen, "On breaking saml: be whoever you want to be," in *Proceedings of the 21st USENIX Security Symposium*, pp. 397–412, USENIX, Bellevue, DC, USA, 2012.

[60] M. A. P. Leandro, T. J. Nascimento, D. R. dos Santos, C. M. Westphall, and C. B. Westphall, "Multi-tenancy authorization system with federated identity for cloud-based environments using Shibboleth," in *Proceedings of the 11th International Conference on Networks (ICN '12)*, pp. 88–93, IARIA, 2012.

[61] S. T. Sun, "Simple But Not Secure: An Empirical Security Analysis of OAuth 2.0-Based Single Sign-On Systems," 2012.

[62] H. J. Lee, I. Jeun, K. Chun, and J. Song, "A new anti-phishing method in openID," in *Proceedings of the 2nd International Conference on Emerging Security Information, Systems and Technologies (SECURWARE '08)*, pp. 243–247, Cap Esterel, France, August 2008.

[63] K. Hwang and D. Li, "Trusted cloud computing with secure resources and data coloring," *IEEE Internet Computing*, vol. 14, no. 5, pp. 14–22, 2010.

[64] H. Khiabani, J.-L. A. Manan, and Z. M. Sidek, "A study of trust & privacy models in pervasive computing approach to trusted computing platforms," in *Proceedings of the International Conference for Technical Postgraduates (TECHPOS '09)*, pp. 1–5, Kuala Lumpur, Malaysia, December 2009.

[65] M. Donovan and E. Visnyak, *Seeding the Cloud with Trust: Real World Trusted Multi-Tenancy Use Cases Emerge*, 2011, http://www.ittoday.info/Articles/Trust/Trust.htm.

[66] P. Cooke, *Black Hat TPM Hack and BitLocker*, 2010, http://blogs.windows.com/windows/b/windowssecurity/archive/2010/02/10/black-hat-tpm-hack-and-bitlocker.aspx.

[67] F. B. Mat Nor, K. Abd Jalil, and J. L. Ab Manan, "Remote user authentication scheme with hardware-based attestation," *Communications in Computer and Information Science*, vol. 180, no. 2, pp. 437–447, 2011.

[68] X. Ding and J. Wei, "A scheme for confidentiality protection of OpenID authentication mechanism," in *Proceedings of the International Conference on Computational Intelligence and Security (CIS '10)*, pp. 310–314, Nanning, China, December 2010.

[69] D. Thibeau and R. Drummond, "Open trust frameworks for open government: Enabling citizen involvement through open

identity technologies," Tech. Rep., OpenID Foudation and Information Card Foudation, 2009.

[70] P. Urien, "An OpenID provider based on SSL smart cards," in *Proceedings of the 7th IEEE Consumer Communications and Networking Conference (CCNC '10)*, pp. 1–2, January 2010.

[71] K. A. Jalil, F. B. M. Nor, and J. A. Manan, "Mitigating man-in-the-browser attacks with hardware-based authentication scheme," *International Journal of Cyber-Security and Digital Forensics*, vol. 1, no. 3, p. 6, 2012.

[72] C. Latze and U. Ultes-Nitsche, "Stronger authentication in e-commerce: how to protect even Naïve user against Phishing, pharming, and MITM attacks," in *Proceedings of the IASTED International Conference on Communication Systems, Networks, and Applications (CSNA '07)*, pp. 111–116, October 2007.

[73] A. Leicher, A. U. Schmidt, and Y. Shah, "Smart OpenID: a smart card based OpenID protocol," *Information Security and Privacy Research*, vol. 376, pp. 75–86, 2012.

[74] H. Hodges and M. Johansson, *Towards Kerberizing Web Identity and Services*, 2008.