KASU JOURNAL OF MATHEMATICAL SCIENCES (KJMS) VOL. 1, ISSUE 2, December 2020
ISSN 2734-3839 (Print), ISSN 2735-962X (E-Copy)
http:/www.journal.kasu.edu.ng/index.php/kjms

KASU JOURNAL OF MATHEMATICAL SCIENCES

Improving Cloud Data Security by hybridization of Zero-Knowledge Proof and Time-Based One-Time Password

Aliyu Ahmed Abubakar
Computer Science Department, Kaduna State University

Abstract

Password has been the major vulnerability and challenge in cloud computing environment due to adversarial threat sources. Zero-Knowledge Proof (ZKP) becomes the excellent optional strategy and current anticipation for Cloud Data security and assurance as it gives the server zero knowledge of passwords. However, the VERIFIER, despite having no knowledge of the password must verify that the password entered by the PROVER is correct. A major challenge that occurs is when the user disremembers the password, the cloud environment becomes inaccessible and the cloud service provider has no any replica of the password as a backup. Eavesdropping/snooping is a major threat which provides full access to the cloud environment. Adding Time-based One-time Password (TOTP) will be an elucidation to password delinquency as it expires after a certain splits of seconds or minutes which would render the password unserviceable to the snooped. This research examined cloud services in relation to the security challenges faced by organizations and proposed the hybridization of the Zero-Knowledge Proof and Time-Based One-Time Password algorithms for a better security of the cloud environment.

Keywords: Cloud Computing, Cloud Data Security, Risk Mitigation, Zero-Knowledge Proof, Time-Based One-Time Password

1. Introduction

Cloud computing has the potential to improve the way businesses and Information Technology operate by offering fast start-up, flexibility, scalability and cost efficiency (Carrol, 2016). There has been an intense increase in the popularity of cloud computing systems that rent computing resources on-demand, bill on a pay-as-you-go basis, and many users on the same physical infrastructure. Cloud computing environments provide an illusion of infinite computing resources to cloud users and it allows them to increase or decrease their resource consumption rate according to demands (Rackspace, 2019). June and Yong (2016) explained that there are a number of security threats associated with cloud data services, not only covering traditional security threats, e.g., network eavesdropping, illegal invasion, and denial of service attacks, but also including specific cloud computing threats, e.g., side channel attacks, virtualization vulnerabilities, and abuse of cloud services. The constant concern in cloud environment is data and information security assurance. Password has constantly been the foremost vulnerability and a registered encounter for

cloud computing consequent to user's forgetfulness and supplementary coercion influences. Cloud Service Providers routinely retain copies of user's password and that could consequently, compromise the cloud data and information privacy and security (Jaydip, 2020). Even if data isn't stolen or published, it can still be viewed. Governments can legally request information stored in the cloud, and it's up to the cloud services provider to deny access. Tens of thousands of requests for user data are sent to Google, Microsoft, and other businesses each year by government agencies. A large percentage of the time, these companies hand over at least some kind of data, even if it is not the content in full (Wendy, 2018). Cloud security is tight, but it is not infallible and cybercriminals can get into those files, whether by guessing security questions or bypassing passwords, just like what happened in The Great iCloud Hack of 2014, where nude pictures of celebrities were accessed and published online (Lewis, 2014). An external source e.g. Hacker which is another key concern could acquire an illegal access to the cloud environment. Eavesdropping and snooping give unauthorized organizational personnel illegal access to Cloud Environment using Zero-Knowledge protocols. This makes a traditional password unguaranteed and insufficient for maximum cloud data security obligation even when Zero-Knowledge Encryption is applied. ZKP has relatively less complex computational requirements as compared to the other protocols for authentication as conventional authentication schemes are susceptible to attacks such as MiTM, IP spoofing, DoS, replay and other eavesdropping based attacks, when the data is shared across an untrusted network. (Cherukupalli & Rajesh, 2020; Jitendra & Ankur, 2015). Thus, this research has investigated and identified the vulnerability of password in the cloud data security and proposed a hybrid system which merges Zero Knowledge Proof And Time-Based One Time Password because of its less complex computational requirements as stated in the gap above. In addition to the use of username and password, additional authentication factors such as biometrics, voice and face recognition could also be used to further ascertain the authenticity of the user. Biometric traits, face and voice recognition are easy to provide and difficult to replicate. However, their combination with password requires more computational resources on the server side, and registration of biometrics, face and voice recognition must be done which leads to privacy issue as the clients may not be comfortable with the combination (Lexus, Sim, Ren, Keoh, & Aung, 2017). This is the justification in selecting ZKP and TOTP for the hybridization instead of the biometric, voice or face recognition.

2. Cloud data Security

2.1 Cloud Risk

Risk management standard of the Institute of Risk Management (2002) has defined as the combination of the probability of an event and its consequences (whether positive or negative). In general, in all types of businesses there are events which represent opportunities for benefit or threats to success, i.e. positive and negative aspects of risks, respectively. Risk in itself is not bad, risk is essential to progress, and failure is often a key part of learning. But we must learn to balance the possible negative consequences of risk against the potential benefits of its associated

opportunity (Van & Roger, 2013). Drissi, Houmani, & Medromi, (2013) said risk assessment is the process of identifying the security risks to a system and determining their probability of occurrence, their impact, and the safeguards that would mitigate that impact. The main objective of risk assessment is hence to define appropriate controls for reducing or eliminating those risks. Risk management however, refers to a coordinated set of activities and methods that is used to direct an organization and to control the many risks that can affect its ability to achieve objectives. We can therefore, study and analyse risk from different perspectives regarding its types, assessment steps and methods in order to find a better mitigation strategy.

Based on Harms-Ringdahl (2001) risk analysis, the method can be divided into qualitative analysis and quantitative analysis:

i. Quantitative Risk Methodologies: which is not commonly used in information technology.
ii. Qualitative Risk Assessments: approach describes likelihood of consequences in detail. This approach is used in events where it is difficult to express numerical measure of risk. It is, for example, the occurrence without adequate information and numerical data. Such analysis can be used as an initial assessment to recognize risk.

Generally there are four steps of risk assessment. The four steps are as follow:

1) Threat Identification 2) Vulnerability Identification 3) Risk Determination 4) Control Recommendation.

The most important risks introduced by Cloud computing are: SLAs breaches, ability to adequately assess risks of a Cloud provider, responsibility to protect sensitive data, virtualization-related risks, loss of direct control of resources and software, compliance risks, and decreased reliability since service providers may go out of business, among others (Singhal, Motahari-Nezhad, & Stephenson, 2009). On the contrary, there are some traditional risks that must be re-evaluated. For instance, the risk of network breaks is now more critical for Cloud organizations since they are totally based on the network. Furthermore, other risks, such as natural disasters, must be considered in a different way (because of the constantly use of external resources) for ensuring high-availability of Cloud services.

2.2 Various methods on Risk Mitigation

2.2.1 Risk Assessment Modelling

Djemame, Armstrong, & Guitart (2016) have explained the elements of risk and the risk inventory which is populated with:

• Assets: Virtual Machine (VM), physical host, Service Level Agreements (SLA), with a description of their characteristics. Risk events are assessed in terms of these.
• Incidents/Risk Scenarios: aim to describe any event, condition or their combination that has the potential to reduce the capacity or availability of an asset. Incidents are composed of:

• Vulnerabilities: describe inherent weaknesses of the asset (e.g. a faulty hardware) and their impact reflects the possibility of a risk incident, e.g. violations of the Quality of Service (QoS), and SLA indicators, inherent to the assets.
• Threats: represent the other side of the risk which depends on factors independent to the asset, e.g. loss of connectivity of a physical host.
• Adaptive capacity: description of the mitigating strategies in place for the specific asset, e.g. server replication.

• Impact/Consequence of a risk incident, e.g. failure of a physical hosts, and is defined using as degraded performance, loss of data, or unavailability. The evaluation is performed according to the indicators selected to describe the asset as well as associated costs, e.g. of not meeting predefined service levels.

Once risks have been identified, evaluated, and reported, the Risk Mitigation process takes place. This involves the definition of potential risk-aware actions, controls, and policies to conduct an appropriate Risk Mitigation methodology, which aims to move risks on the negligible or profitable levels. In this sense, there are four possible responses to effectively deal with each risk. Avoid the risk, by eliminating its cause(s). Reduce the risk by taking steps to cut down its probability, its impact, or both. Accept the risk and its related consequences. Transfer or delegate the risk to external organizations. Obviously, risks with a positive risk level will always be 'accepted'. On the contrary, for negative risks, the organization has the other three alternative responses. Future risks' impacts will depend immeasurably on these decisions taken. Furthermore, there are various methods of risks mitigation, but firstly, the most important risks to which the Cloud Service Provider (CSP) is usually exposed to will be identified. In particular, these risks are also classified into the following groups based on (Guitart & Fit´o, 2009) which is also similar to (Djemame, Armstrong, & Guitart, 2016)'s classification:
1) Cloud capacity provisioning. We distinguish between two types of risk, which are over and under-provisioning a given Cloud. Regarding the CSP, this risk affects assets constituting its own private Cloud.
2) Service Level Agreements (SLA), such as the risks of accepting new Cloud service's SLA, its violations due to poor performance and service disruptions, etc.
3) Virtualization, i.e. those related with the underlying technology of Clouds, such as the risks of virtual machine isolation or virtualization performance overhead.

4) Cloud applications data. In this case we consider the risks of data integrity loss, destruction of data, etc.
5) Cloud resources outsourcing. Risks associated with the loss of governance and hidden costs, among others.
6) Others. Here we group the risks of power loss to the IT systems, natural disasters, fire, etc.VMs, SLAs, and entire infrastructure as CSP's main assets, a risk mitigation strategy can be implemented and also considering the vulnerabilities and threats associated with physical nodes,

ISSN 2734-3839 (Print), ISSN 2735-962X (E-Copy)
http:/www.journal.kasu.edu.ng/index.php/kjms

VMs, SLAs, and entire infrastructure. As part of implementation of the risk mitigation, a risk inventory plays a critical role in the output of a risk model and what assets, threats, vulnerabilities and impacts are considered. The following defines the elements of the risk inventory used as input to the risk model in its design and implementation.

3. Zero-Knowledge Proof (ZKP)

Ryan (2014) wrote: Privacy-Enhancing Technologies (PETs) are a class of services, applications, and mechanisms that help mitigate online privacy threats by empowering users with control over the collection, dissemination, and use of information about themselves and about their day-to-day activities. According to Zero-knowledge proof was invented by Shafi et al. (GMR89), and since then, it has been used in a number of cryptographic applications. ZKE normally works in interactive proof systems and it has two parties which are:

i. An (all powerful) Prover, frequently called Peggy (a randomized algorithm using a private random number generator)

ii. A (little (polynomial) powerful) Verifier, frequently called Vic (a polynomial time randomized algorithm using a private random number generator).

Prover knows some secret, or knowledge, or a fact about a specific object, and wishes to convince Vic, through a communication with him, that he has the above knowledge. Very informally, a zero-knowledge proof protocol allows one party, usually called prover/Peggy, to convince another party, called verifier/Vic, that prover knows some facts without revealing to the verifier ANY information about his knowledge. The purpose of zero knowledge protocols is to allow verification to be performed accurately without revealing the witness, or in fact leaking any other information. It has two major components as explained, which are the Prover and the Verifier. It is another effective way of securing data on the cloud as the cloud service providers and the servers will have no any knowledge or a copy of the user's password. In this case, even if an identity thief hacks the server, no any knowledge or information of the User's password shall be found.

For zero-knowledge proof to work, it must satisfy three properties:

I. Completeness: if the statement is true, the honest verifier (that is, one following the protocol properly) will be convinced of this fact by an honest prover.
II. Soundness: if the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability.
III. Zero-knowledge: if the statement is true, no verifier learns anything other than the fact that the statement is true. In other words, just knowing the statement (not the secret) is sufficient to imagine a scenario showing that the prover knows the secret. This is formalized by showing that every verifier has some simulator that, given only the statement to be proved (and no access to the prover), can produce a transcript that "looks like" an interaction between the honest prover and the verifier in question.

4. One-time Passwords (OTPs)

OTP techniques were used to reduce the damage of passwords compromised through many attacks like spyware and replay attacks (Abukeshipa & Amna, 2014). Traditional user authentication mechanisms like public/private key pairs suffer from a common weakness which is the users need to choose a good password that contains strong passphrases to protect their privacy and credentials. While users access systems from remote hosts, their credentials are stored at untrusted hosts. While these hosts are infected by malware or phishing and spyware attacks that harvest untrusted key or password as they are entered, these credentials can be reused by an attacker. OTP generation algorithms typically make use of randomness, making prediction of successor OTPs by an attacker difficult, and also hash functions, which can be used to derive a value but are hard to reverse and therefore difficult for an attacker to obtain the data that was used for the hash. This is necessary because otherwise it would be easy to predict future OTPs by observing previous ones. Concrete OTP algorithms vary greatly in their details. Various approaches for the generation of OTPs can be presented and listed below:

- Based on time-synchronization between the authentication server and the client providing the password (OTPs are valid only for a short period of time)
- Using a mathematical algorithm to generate a new password based on the previous password (OTPs are effectively a chain and must be used in a predefined order).
- Using a mathematical algorithm where the new password is based on a challenge (e.g., a random number chosen by the authentication server or transaction details) and/or a counter.

There are also different ways to make the user aware of the next OTP to use. Some systems use special electronic security tokens that the user carries and that generate OTPs and show them using a small display. Other systems consist of software that runs on the user's mobile phone. Yet other systems generate OTPs on the server-side and send them to the user using an out-of-band channel such as SMS messaging. Finally, in some systems, OTPs are printed on paper that the user is required to carry.

A time-synchronized OTP is usually related to a piece of hardware called a security token (e.g., each user is given a personal token that generates a one-time password). It might look like a small calculator or a keychain charm, with an LCD that shows a number that changes occasionally. Inside the token is an accurate clock that has been synchronized with the clock on the proprietary authentication server. On these OTP systems, time is an important part of the password algorithm, since the generation of new passwords is based on the current time rather than, or in addition to, the previous password or a secret key. This token may be a proprietary device, or a mobile phone or similar mobile device which runs software that is proprietary, freeware, or open-source. An example of time-synchronized OTP standard is Time-based One-time Password Algorithm (TOTP)

4.1 OTPs Mathematical Algorithms

(i). A seed (starting value) is chosen.
(ii). A hash function f(s) is applied repeatedly (for example, 1000 times) to the seed, giving a value of: f(f(f(.... f(s) ....))). This value, which we will call f1000(s) is stored on the target system.
(iii). The user's first login uses a password p derived by applying f 999 times to the seed, that is, f999(s). The target system can authenticate that this is the correct password, because f(p) is f1000(s), which is the value stored. The value stored is then replaced by p and the user is allowed to log in.
(iv). The next login, must be accompanied by f998(s). Again, this can be validated because hashing it gives f999(s) which is p, the value stored after the previous login. Again, the new value replaces p and the user is authenticated.
This can be repeated another 997 times, each time the password will be f applied one fewer times, and is validated by checking that when hashed, it gives the value stored during the previous login. Hash functions are designed to be extremely hard to reverse, therefore an attacker would need to know the initial seed s to calculate the possible passwords, while the computer system can confirm the password on any given occasion is valid by checking that, when hashed, it gives the value previously used for login. If an indefinite series of passwords is wanted, a new seed value can be chosen after the set for s is exhausted

4.2 OTPs versus other methods of securing data

One-time passwords are vulnerable to social engineering attacks in which phishers steal OTPs by tricking customers into providing one or more OTPs that they used in the past. The password may be used as quickly by the attacker as the legitimate user, if the attacker can get the OTP in plaintext quickly enough. Another type of attack - which may be defeated by OTP systems implementing the hash chain as discussed above - is for the phisher to use the information gained (past OTP codes which are no longer valid) by this social-engineering method to predict what OTP codes will be used in the future. For example, an OTP password-generator that is pseudo-random rather than truly random might or might not be able to be compromised, because pseudo-random numbers are often predictable once one has the past OTP codes. An OTP system can only use truly random OTPs if the OTP is generated by the authenticator and transmitted (presumably out-of-band) to the user; otherwise, the OTP must be independently generated by each party, necessitating a repeatable, and therefore merely pseudo-random, algorithm.

5. Time-based One Time Password (TOTP)

Network security is a very important issue for organizations in order to protect their sensitive data from attackers. Number of researchers have provided different security solution supported network protocols to enhance data privacy and confidentiality over networks. RADIUS is one of the most popular protocols used in network communication for user authentication. Unfortunately, there is much vulnerability facing the security issue in RADIUS network protocol. One of these vulnerabilities is a replay attack problem which is needed to be prevented. The previous protocols have presented number of techniques to reduce the effects of replay attack in RADIUS protocol.

One Time Password (OTP) technique is one of the most important techniques which are used to enhance the security of user authentication in numerous environments and to close the potential gap in network security. With several OTP techniques, including the Time-based OTP (TOTP), Hash OTP (HOTP) and Challenge Response OTP (CROTP).

According to Lee, Lee, Lee, Choi, & Sung-Jae (2011) TOTP is an instant password, in other words, a code that changes after every time we use it to authenticate. TOTP are passwords that are only valid for a single or small number of transactions. An attacker has a smaller period of time to gain access to resources protected by such password because any previously stolen passwords will likely have become invalid which means adding some uncertain factors in the procedure of authorization. Every time user logins, the information transmitted over network is different, thus the security is improved. TOTP has a characteristic making it impossible to predict the next password from the current password; also they are not vulnerable to replay attacks (Luo, Wu, & Lan, 2012; Madhuri, 2010; Wei & Ang, 2010) TOTPs avoid a number of shortcomings that are associated with traditional (static) passwords. On the downside, TOTP will be difficult for human beings to memorize (Lee et al., 2011) P is based on cryptographic algorithms:
Cryptogram=f(k) while key k a cryptographic is generated
TOTP = HOTP(Secret Key, TC),
TOTP-Value = TOTP mod 10d, where d is the desired number of digits of the one-time password.
The reference implementation is as follows:

- Generate a key, K, which is an arbitrary byte string, and share it securely with the client.
- Agree upon a T0, the Unix time to start counting time steps from, and an interval, TI, which will be used to calculate the value of the counter C (defaults are the Unix epoch as T0 and 30 seconds as TI)
- Agree upon a cryptographic hash method (default is SHA-1)
- Agree upon a token length, N (default is 6)

5.1 TOTP Weaknesses and vulnerabilities

TOTP codes can be phished just as passwords can, though they require phishers to proxy the credentials in real time rather than collect them later on in time. Because TOTP devices have batteries that go flat, clocks that can de-sync, and because software versions are on phones that users can lose or have stolen, all real-world implementations have methods to bypass the protection (e.g.: printed codes, email-resets, etc.), which can cause a considerable support burden for large user-bases, and also gives fraudulent users additional vectors to exploit. TOTP codes are valid for longer than the amount of time they show on the screen (usually two or more times longer). This is a concession that the authenticating and authenticated sides' clocks can be skewed by a large margin. All One Time Password-based authentication schemes (TOTP and HOTP included, among others) are still vulnerable to session hijacking, i.e., commandeering a user's session after they have logged in.

6. Conclusion

This paper discussed the less computational complexity as the justification in selecting zero-knowledge proof. TOTP is also selected for the Hybridization due to requirements complexity of other security measures such as biometric, voice and face recognition. The password will be generated using either a token device or a mobile phone application. Password on Cloud Computing Environment has been the foremost vulnerability and challenge due to adversarial threat sources, forgetfulness of the user and sabotages from within organizations. Zero-Knowledge Proof (ZKP), however, becomes the excellent optional and trending strategy and current anticipation for cloud data security and assurance as it gives the server zero knowledge of passwords. This research, correspondingly, has thoroughly studied additional security measure which is adding a One Time Password to make the cloud environment fully hack proof. A number of OTPs were studied and Time-based OTP is adopted to be merged with ZKP for a more secure cloud platform. The process of the TOTP generation consists of:

i) Input value
ii) TOTP generation
iii) TOTP extraction
iv) Time

The new system for cloud data security using Zero-Knowledge Proof and Time-Based One-Time Password hybridization is been proposed based on Lexus et. al. (2017) conducted novel scheme that does not require the user to present his credentials, and yet is able to prove ownership of access to the cloud service using a variant of zero-knowledge proof. He devised a challenge-response protocol to authenticate the user, requiring the user to compute a One-Time Pad (OTP) to authenticate himself to the server without revealing password to the server. A prototype has been implemented to facilitate the authentication of the user when accessing Dropbox, and the experiment results showed that the overhead incurred is insignificant (Lexus, Sim, Ren, Keoh, & Aung, 2017). The distinction and gap here is that Lexus used OTP while this research proposed TOTP for the Hybridization using the same Dropbox.

The limitation of this research include:

i. Hash OTP (HOTP) and Challenge Response OTP (CROTP) are all types of OTP and are not been applied.
ii. Only a few available cloud computing service providers were used for study and evaluation as there are many cloud service providers today.

Future Work

The future work is to Improve Cloud Data Security by hybridization of Zero-Knowledge Proof and Hash OTP (HOTP) or Challenge Response OTP (CROTP) which are all types of OTP. Another research could also be conducted to hybrid ZKP with either Biometric, Voice or face recognition.

References

Abukeshipa, A., & Amna, S. (2014). *Implementing and Comprising of OTP Techniques (TOTP,HOTP,CROTP) to Prevent Replay Attack in RADIUS Protocol.* Gaza: A Thesis Submitted in Partial Fulfillment of the Requirements forthe Degree of Master in Information Technology, Deanery of Higher Studies, Faculty of Information Technology, Islamic University of Gaza.

Carrol, M. (2016, Movember 21). *Secure Cloud Computing: Benefits, Risks and Controls.* Retrieved July 10, 2017, from ieeexlore.ieee.org: http://ieeexplore.ieee.org/document/6027519

Chaidos, P., & Ioannis, P. (2017). *Zero Knowledge Protocols and Application (A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, Security & Crime Science).* London: University College press, London.

Cherukupalli, V. V., & Rajesh, K. M. (2020). Zero-Knowledge Proof Based Authentication. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 238.

Djemame, J., Armstrong, K., & Guitart, D. (2016). *A Risk Assessment Framework for Cloud Computing.* IEEE Transactions on Cloud Computing, Availabale: https://doi.org/10.1109/TCC.2014.2344653.

Drissi, S., Houmani, H., & Medromi, H. (2013). Survey: Risk Assessment for Cloud Computing. *(IJACSA) International Journal of Advanced Computer Science and Applications,*, 144.

Guitart, J., & Fit´o, J. O. (2009). *Introducing Risk Management into Cloud Computing.* University of Catalonia , Barcelona Supercomputing Center and Technical, Barcelona.

Harms-Ringdahl, L. (2001). *Safety analysis: Principles and practice in occupational safety.* CRC Press.

IRM. (2002). *Risk Management Standard.* London: " The Institute of Risk Management, (IRM), Website.

Jaydip, S. (2020, December 1). *curity and Security andPrivacy Privacy Privacy Issues in Cloud.* Retrieved from https://arxiv.org/ftp/arxiv/papers/1303/1303.4814.pdf

Jitendra, K., & Ankur, S. (2015). A Survey of Zero-Knowledge Proof for Authentication. *International Journal of Advanced Research in Computer Science and Software Engineering*, 494.

June, T., & Yong, C. (2016, Movember 22). *Ensuring Security and Privacy Presentation for Clound Data Services.* Retrieved September 7, 2017, from ACM Computing Servey: https://en.wikipedia.org/wiki/Cloud_computing_security

Lee, J. S., Lee, M.-K., Lee, S. J., Choi, D.-H., & Sung-Jae, D. K. (2011). "Low-Power Design of Hardware One-Time Password Generators for Card-Type OTPs". *ETRI Journal, 33*.

Lewis, D. (2014, September 2). Retrieved from https://www.forbes.com/sites/davelewis/2014/09/02/icloud-data-breach-hacking-and-nude-celebrity-photos/?sh=14f1c4172de7

Lexus, Sim, J. H., Ren, S. Q., Keoh, S. L., & Aung, K. M. (2017). *A Cloud Authentication Protocol using One-Time Pad.* Retrieved June 6, 2018, from www.eprints.gla.ac.uk: http://eprints.gla.ac.uk/123504/1/123504.pdf

Luo, S. Z., Wu, Y., & Lan, Y. W. (2012). "A Technique for Preventing Replay Attack in Road Networks". *International Conference on Advanced Information Networking and Applications* (pp. page 807-810). IEEE.

Madhuri, R. L. (2010). "attack patterns for detecting and preventing ddos and replay attacks". *international journal of engineering and technology*, *2 (9)*, pp. 4850-4859.

Rackspace. (2019). *Rackspace*. Retrieved February 23, 2018, from www.rackspacecloud.com: http://www.rackspacecloud.com/.79

Ryan, H. (2014). *Efficient Zero-Knowledge (A thesis presented to the University of Waterloo in fulfillment of the thesis requirement for the degree of Doctor of Philosophy in Computer Science).* Waterloo, Ontario, Canada.

Shafi, G., Silvio, M., & Charles, R. (2020, December 4). *The Knowledge Complexity of Interactive Proof-Systems.* Retrieved from citeseerx.ist.psu.edu: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.419.8132&rep=rep1&type=pdf

Singhal, S., Motahari-Nezhad, H., & Stephenson, B. (2009). “Outsourcing Business to Cloud Computing Services: Opportunities and Challenges”. *Report HPL-2009-23.* HP Labs,.

Van, S., & Roger, L. (2013). Software Development Risk: Opportunity, Not Problem. *IJACSA) International Journal of Advanced Computer Science and Applications*, 143.

Wei, G. W., & Ang, W. S. (2010). "Efficient Password-Proven Key Exchange Protocol against Relay Attack on Ad Hoc Networks". *International Conference on Advanced Information Networking and Applications* (p. 8). IEEE.

Wendy, Z. (2018, July 26). Retrieved from https://blog.malwarebytes.com/101/2016/04/should-you-store-your-data-in-the-cloud/.