

Data Security at Cloud Storage using PGP in conjunction with IPsec VPN

MSc Research Project
Cybersecurity

Regin Raghavan Nair
Student ID: X21173281

School of Computing
National College of Ireland

Supervisor: Evgeniia Jayasekera

National College of Ireland
MSc Project Submission Sheet



School of Computing

Student Name: Regin Raghavan Nair
Student ID: X21173281
Programme: MSc Cybersecurity **Year:** 2023
Module: MSc Research Project
Supervisor: Evgeniia Jayasekera
Submission Due Date: 14 August 2023
Project Title: Data Security at Cloud Storage using PGP in conjunction with IPsec VPN
Word Count: 6144 **Page Count:** 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Regin Raghavan Nair

Date: 12 August 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

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

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

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

Data Security at Cloud Storage using PGP in conjunction with IPsec VPN

Regin Raghavan Nair

X21173281

Abstract

Cloud computing has revolutionized data management and storage, but it has also raised concerns about privacy and control over personal information. FOG computing can help to address these concerns by bringing processing closer to the source of the data, which reduces the risk of data exposure. The storage model relies on cryptography, and PGP can enhance privacy protection when combined with a three-layer storage space framework that supports fog computing. Using a VPN and managing cloud services can further improve data security and ensure authorized access. This study proposes a method to increase the security of cloud storage by combining encryption with PGP and ensuring that the data in transit is also protected by using a VPN as a communication channel. This method fills a security hole in existing cloud storage methods and proposes a safe method that can help businesses use cloud storage for data management.

Key Words: Cloud Computing, Fog Computing, PGP, Edge Computing, Cryptography.

1. Introduction

Cloud computing has brought about a revolutionary change in how we store and manage data. The popularity of cloud storage technology has soared due to the explosive growth of unstructured data, enabling users to access their data from anywhere in the world using any device. However, this convenience also comes with inherent risks, such as the loss of control over personal information and potential privacy invasion. In recent years, edge computing has emerged as a promising solution to address these concerns. It supports a distributed computing infrastructure that brings processing closer to the source data, reducing the risk of sensitive information exposure.

To enhance the security of FOG computing, cryptography plays a vital role in the storage model. Traditional privacy protection solutions, like secret-creating innovation, are susceptible to attacks from cloud servers. To combat this, researchers have proposed a three-layer storage space framework tailored for fog computing. This framework combines the advantages of cloud storage with enhanced privacy protection. The coding formula, Hash-Solomon, is utilized to segment information into different sections, while PGP is employed as an encryption technology combining symmetrical and asymmetrical techniques. This ensures that only authorized individuals have access to the data. To further enhance personal information security, a substantial amount of data is received and processed by local machines and fog servers. This method estimates the distribution proportion retained in cloud, haze, and native devices separately. The framework's performance has been verified through secure learning environments and experimental evaluations.

For access control, end-users receive a token from the solution site after being certified by a third-party authority. By registering with the solution site, individuals can buy and utilize cloud services

from a single service provider. The end-user service website provides secured access control using Virtual Private Network (VPN). It encompasses accessibility control, security plan, key administration, service configuration, accounting management, and digital environments. The integration of FOG computing with cryptography has significantly improved the security of cloud storage. The three-layer storage space framework, along with the use of PGP, represents a substantial advancement over traditional cloud storage approaches. The addition of VPN and cloud service management further enhances the security of personal information and ensures access is granted only to authorized users.

1.1 Pretty Good Privacy (PGP)

PGP functions as a sophisticated cryptographic mechanism that seamlessly blends symmetric and asymmetric encryption methods, ensuring secure communication and data protection with a well-balanced approach. When encrypting plaintext using PGP, the process begins with data compression, which enhances cryptographic security by minimizing vulnerable patterns. Subsequently, a session key is generated from random sources, laying the groundwork for symmetric encryption. During the symmetric encryption phase, the plaintext undergoes encryption using a rapid and secure algorithm, transforming it into ciphertext for robust content protection. PGP ensures a secure key exchange by utilizing the recipient's public key to encrypt the session key, facilitating secure communication. Symmetric encryption is utilized for encrypting the actual plaintext message during the encryption phase. On the other hand, asymmetric encryption is used when encrypting the session key, providing secure key exchange. The consequent encrypted session key and ciphertext are then communicated to the recipient (Kurniawan et al., 2011; "PGP - Pretty Good Privacy," n.d.).

In PGP's public key system variation, users possess both public and private keys. To save resources, PGP employs symmetric encryption to cipher the message and correspondingly encrypts the symmetric encryption key using the receiver's public key. This comprehensive approach guarantees confidentiality and authenticity during data exchange. This cryptographic flexibility makes PGP easily integrate into multiple software applications and systems, demonstrating to be a highly trusted solution for ensuring security and integrity sensitive information(*PGP Explained* | Fortinet, n.d.; *Pretty Good Privacy*, n.d.).

2.1.1 Symmetric encryption technology

Symmetric encryption is a technique that uses the same key to perform both the encryption and decryption process. The use of the same key speeds up the process of encryption and is especially effective in case of large amount of data has to be encrypted. Symmetric algorithms are considered to be highly secure, but it needs to be kept in mind that the sharing of the key should be in a secure way and compromise of the key is nothing but compromise of the entire system.

Some of the most widely used symmetric techniques are.

1. *Advanced Encryption Standard (AES)*: AES is most commonly used symmetric encryption algorithm known for its encryption speed and security. It operates on fixed block sizes of 128, 192, or 256 bits and uses a specific number of encryption cycles based on the key size.
2. *Data Encryption Standard (DES)*: Considered to be less secure than AES, DES operates on 64-bit blocks and uses a 56-bit key. Due to the small key size, DES is now considered vulnerable against modern attacks.
3. *Triple-Data Encryption Standard (3DES)*: 3-DES performs DES three times with two or three different keys on each cycle. Though it is said to provide better security compared to DES, it is comparatively less efficient and not as fast as AES.

2.1.2 Asymmetric Key Encryption

Asymmetric key encryption is also called public key encryption. This method makes use of the different keys but inter-related to each other to perform the encryption, called the private key and the public key. In each key pairs, both the key perform its own role in carrying out the encryption and decryption tasks. The public key is shared with the users and the private key is retained with the owner, undisclosed. Some of the most widely used asymmetric techniques are.

1. *RSA*: A common asymmetric encryption technique is RSA (Rivest-Shamir-Adleman). Information security is achieved by using modular math and large prime integers. To exchange secure keys and generate digital signatures, RSA is frequently employed.
2. *ECC*: Another kind of asymmetric encryption is ECC (Elliptic Curve Cryptography). It provides good security with shorter keys than RSA and is based on elliptic curve mathematics. For devices with constrained resources, this increases the efficiency of ECC.
3. *DSA*: To create and validate digital signatures, the DSA (Digital Signature Algorithm) is utilized. Secure communication and digital certificates both often use it.

2. Research Question and objective

3.1 Research Question

Is there a scope for enhancement of security of data sent from FOG compute environment to cloud storage by utilizing PGP mechanism in conjunction with use of VPN, rather than depending only on Cloud Service Provider (CSP) to protect the data?

3.2 Research Objective

This research focuses on coming up with techniques to ensure security of the data that will be sent by a FOG compute environment to cloud for storage(Karra et al., 2019). The primarily focuses on developing a method that can be easily integrated to existing infrastructure without adding additional cost to the organization and also not adding a very significant overhead to the compute and memory resources of the organization. The focus of the project is not limited to ensuring the security of the data at rest, but also focuses on security of data in transit by configuring an industry standard secure VPN channel instead of using an open-source VPN tools such as OPEN-VPN.

3. Report Structure.

The section describes the sequence of different divisions that follows in this report. At this point the report has already provided the introduction to the topic and also stated the research question along with explaining the objective. The sections following are *Related work and literature review*, *Research Methodology*, *Design and implementations*, *Evaluation* and finally *conclusions* and *Discussions* where outcome of research and future scope of work will be suggested which was not achievable due to unforeseen reasons.

4. Literature Survey

5.1 Introduction.

With the increasing adoption of cloud infrastructure, security concerns have become a crucial issue that cannot be ignored. Fog computing has emerged as a promising solution to tackle these challenges. Unlike cloud computing, fog computing allows cloud services to be utilized within a local network, processing data before storing it there. However, this system heavily relies on Cloud Service Providers (CSPs) to safeguard the data from potential attacks. Unfortunately, this reliance makes it more vulnerable to cyberattacks, potentially leading to data breaches and the loss of critical information. A thorough examination of prior research in these fields will be conducted in order to comprehend the shortcomings and limitations of present methodologies. This literature study primarily aims to investigate how we may enhance data security while transferring data from Fog nodes to cloud devices in a Fog computing environment. In order to develop a comprehensive grasp of the topic, the examination will also go into other pertinent elements and address connected issues.

5.2 Background Context

As per (Shruti & Rani, 2022) (Mukherjee et al., 2017), the advent of Fog computing has provided businesses with the capability to manage the increasing need for real-time data processing and storage. Fog computing boasts reduced latency and accelerated data processing when compared to cloud computing. However, concerns have been raised regarding the security of these systems due to their heavy reliance on wireless communication channels and the distributed nature of Fog computing setups. Security risks such as data breaches, unauthorized access, and data tampering have been identified. To address these challenges, the study proposes the implementation of multiple security layers as a means to enhance the security of Fog computing environments.

According to (Rezapour et al., 2021) (T. Wang et al., 2018) research in 2021, security stands as a significant concern for Fog computing systems, demanding serious attention from both businesses and researchers. Given their decentralized structure and dependence on wireless communication channels, fog computing systems are exposed to various security risks, such as data tampering, unauthorized access, and data breaches. To address these security challenges, employing secure data sharing protocols emerges as a viable approach for fog computing environments.

Edge computing is another critical aspect of fog computing, involving data processing at the network's edge rather than a central location. This approach reduces data transport latency, potentially improving fog computing system performance. However, Aljumah et al. (2017) (P. Y. Zhang et al., 2018) point out that edge computing introduces security challenges that must be addressed. Edge computing devices often reside in unprotected areas, increasing their vulnerability to cyber-attacks. Moreover, their small size and limited processing capabilities make them more susceptible to security breaches. Implementing security protocols like authentication, authorization, and encryption becomes essential to safeguard edge computing devices from potential threats.

In the context of fog computing, access control implementation relies heavily on access control policies. These policies define who can access data, when, and under what conditions. Ensuring appropriate access controls are in place is vital to maintain data security and privacy in fog computing environments(Aljumah & Ahanger, 2018).

According to Rezapour et al.'s research in 2021, cryptography plays a critical role in ensuring security within fog computing. By employing mathematical techniques to encrypt data, fog computing achieves secure communication and information storage, safeguarding data privacy and preventing unauthorized access to sensitive information. Additionally, cryptography in fog computing systems helps ensure data integrity and consistency. However, it also introduces security challenges that must be addressed to ensure the success of the fog computing system(Rezapour et al., 2021).

Another security concern highlighted by Alhuman et al. (2023) and Kiwelekar et al. (2021) regarding cryptography in fog computing is the vulnerability of encryption keys to attacks. Data encryption and decryption require encryption keys, and if these keys are lost or stolen, attackers may have access to private information. To address this issue, efficient key management techniques are required to protect the confidentiality and integrity of encryption keys. One option is to use key encryption technologies to protect the keys by encrypting them with different keys. Additionally, the use of hardware-based key management systems optimized for fog computing settings is another viable solution(Alhumam et al., 2023; Kiwelekar et al., 2021).

5.3 Related Work.

As per (Sadkhan, 2021; Zaw et al., 2019) while analyzing Elliptic Curve Cryptography (ECC) and its applications, classifying previous works based on their objectives. ECC's advantages, like high security with smaller keys, and its suitability for bandwidth-limited devices are highlighted. However, the paper acknowledges challenges in ECC, such as the need for strong randomness generation and difficulties in protecting hash functions. Scalability issues in cloud servers are also addressed, proposing a blooming filter as a potential solution. The author argues that ECC alone is not sufficient for cloud data security due to challenges related to the quality of randomness generation, difficulties in protecting hash functions, potential weaknesses in certain ECC schemes, and vulnerability to specific attacks. These factors can compromise the overall security of ECC in

cloud environments, necessitating additional measures and complementary cryptographic techniques to ensure robust protection for sensitive data stored and transmitted in the cloud.

The research by (C. Wang et al., 2018)(Samydurai et al., 2015) focuses on cryptographic cloud storage using Ciphertext-Policy Attribute-Based Encryption (CP-ABE). CP-ABE allows fine-grained access control over encrypted data in the cloud, but it faces challenges with access policy revocation. The authors critically examine existing revocation schemes and discuss a method which divides data into slices for more efficient revocation. However, they identify limitations such as the method's one-cycle all-or-nothing property, potential attacks by revoked users, inefficiency in the revocation process, scalability issues, and the possibility of data recovery even after revocation. In summary, CP-ABE alone is not sufficient for cloud storage because of the difficulties in efficiently revoking access when access policies change. The limitations in current revocation schemes can lead to security vulnerabilities and hamper the overall efficiency and scalability of access policy revocation in cryptographic cloud storage systems(Guo et al., 2017).

(L. Zhang et al., 2016) (Acar et al., 2018)conducted an extensive study on Homomorphic Encryption and its diverse applications, enabling secure computations on encrypted data without decryption, safeguarding privacy in IoT and cloud computing. THE paper thoroughly explores various homomorphic cryptosystems, including Partially HE (PHE) and Fully HE (FHE), elaborating on their strengths, drawbacks, and potential use cases. Homomorphic Encryption's privacy-preserving applications in cloud computing, genomic data storage, and participatory sensing are meticulously examined. Nevertheless, the study uncovers several significant limitations: computational and communication inefficiencies in both PHE and FHE, challenges in large-scale implementation, high interaction costs in Secure Multiparty Computation (SMC), storage overheads, and complexities in supporting complex aggregations while preserving anonymity.

In their research work,(Rehman et al., 2021) (Zaw et al., 2019)describe AES-ECC, a novel method for enhancing cloud data security. To overcome the drawbacks of typical encryption systems, notably in terms of computing complexity and time, author combines AES and ECC techniques. The paper adopts a conventional research process, examining existing plans, highlighting their shortcomings, and suggesting the AES-ECC hybrid model. The research covers materials, methodologies, software, and system requirements while providing a full explanation of the use and analysis of AES-ECC. The authors emphasize how well AES-ECC works to mitigate the noted drawbacks. Key drawbacks of existing cloud data security techniques are identified in the article, including computation complexity, encryption key size, and the absence of user private key data in the CSP. Overall, the report acknowledges the need for more study and improvement of AESECC hybrid approach.

(Soman & Natarajan, 2017)(Sood, 2012). present a research paper discussing a novel hybrid data security algorithm, which combines ECDSA, SHA256, and AES to ensure secure data communication within cloud environments. The primary objective is to enhance data security in cloud computing platforms. This algorithm involves generating SHA256 message digests and

ECDSA digital signatures on the client machine, followed by AES encryption using a public key for data or message protection. However, the study highlights a crucial limitation of the hybrid approach. While it proves successful in enhancing data security, the algorithm's performance suffers when handling large-scale data in cloud computing. The computational resources required for encrypting and decrypting extensive datasets using sophisticated cryptographic techniques make real-time applications less feasible.

Serial No	Title of the Paper Reviewed	Authors	Advantages	Disadvantages
1)	Elliptic Curve Cryptography- Status, Challenges and Future trends	Sattar B. Sadkhan	Suitability in limited bandwidth, Suitability for pervasive computing, High speed in low bandwidth, Better performance than older algorithms.	Weakness in ECSchnorr, Vulnerability to Certain Attacks, Dependence on Quality of Randomness, Difficulty in Hash Functions.
2)	efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage	Changji Wang, Jing Liu, Yuan Yuan, Jiayuan Wu.	Fine-Grained Access Control.	Unauthorized data restoration after revocation, Security issues in cloud storage, Access policy revocation problems, Vulnerability to attacks.
3)	A Review of Homomorphic Encryption and its Applications	Lifang Zhang, Yan Zheng, Raimo Kantola	IoT Interconnectivity, Efficient PIR, Participatory Sensing, Multi-party Sensing, Versatile Homomorphism.	Challenges in Large-scale Implementation, Dependence on Non-standard Assumptions, Message Expansion and Overhead, Complexity of Implementing Complex Aggregations.
4)	Hybrid AES-ECC Model for the Security of Data over Cloud Storage	Saba Rehman ,Nida Talat Bajwa ,Munam Ali Shah ,Ahmad O. Aseeri Adeel Anjum.	Optimized public key encryption, Enhanced performance and efficiency, Effective use of AES and ECC, Better attack prevention.	Larger computation overhead cost, Increased computation time, Larger key size for encryption, Increased size of encrypted messages.

5)	An enhanced hybrid data security algorithm for cloud	Soman, Vikas K. Natarajan, V.	Scalability, Enhanced Security, Confidentiality and Integrity, Metering and Pay-as-You-Use.	Massive Data Handling difficulty, high Computational Resources requirement, less Real-time Feasibility
----	------------------------------------------------------	-------------------------------	---------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------

5.4 Literature gap

1) Most of the discussed techniques in the above studies does not focus on compatibility of the proposed techniques in an existing environment, and computational overhead in terms of processing power or large /complicated key generation when using techniques like CP-ABE and ECC.

Proposed solution: By use of combination like AES+RSA which some of predominantly know encryption technique and combination of these techniques with increase the security exponentially compared to when used individually.

2) All the above research is narrowed down only to ensure security of the data by encrypting it with a wide range of technology, The has been very less focus towards developing a system that can also ensure a secure medium for transport of this encrypted data. Security of data in transit is not considered with equal priority.

Proposed solution: implement a IPsec Site to Site VPN, which would provide an encrypted medium which only allows the designated traffic to pass through it. This ensure data at transit is protected, adding additional layer of security.

5. Research Methodology

The methodology approached in this paper is focused on identifying a solution which is suitable to be integrated into an existing infrastructure of an organization to add an extra layer of security to critical data owned by an organization, stored in a public cloud storage. The cloud technology is one of the most widely used technology in the industry due to its scalability, availability, and cost effectiveness, but it comes with other downfall such a concern of security of the critical client data and dependency on cloud provider to ensure data security. To overcome this problem the research focuses on coming up with a solution using a combination of Symmetric and Asymmetric key encryption method. While every other work we referred to has focused on security and integrity of data at rest, this research will focus on contributing more by adding third layer security by ensuring data in transit is also secured using a secure channel.

At the first stage we focus on processing the message. here we work towards encrypting the message using a combination of symmetric encryption and asymmetric encryption method. we use a combination of AES (Advanced Encryption Standard) and RSA(Rivest–Shamir–Adleman) to

encrypt the message and encrypt the encryption key used in the first stage adding two layers of security.

In this research we go an extra mile by adding another layer of security by configuring a secure VPN tunnel to transfer the encrypted message and key. VPN tunnel itself uses an AES encrypted channel with Md5 hashing and tunnel is negotiated once all parameters including the pre-shared key matches at both ends. The tunnel is designed to allow only specific traffic, which is only intended to travel through tunnel, every other traffic flows over open internet.

By following the above techniques, the methodology implemented in this research ensures that the data at rest and at transit is secured.

5.1 Pretty Good Privacy

At the initial stage the system generates a session key, the payload to be encrypted is then encrypted using the symmetric key encryption method and session key as the encryption key, we use AES as the encryption mechanism in this case. At the next phase the session key is encrypted using the receiver's public key. At the second phase the asymmetric encryption takes place using the RSA (Rivest-Shamir-Adleman) method. The ciphertext and encrypted key is then sent to the receiver and the receiver decrypts the session key using his private key. Once the session key is obtained the cipher text is decrypted to reveal the original message. Figure 1 shows the flowchart representing operation of PGP mechanism (*How PGP Works*, n.d.).

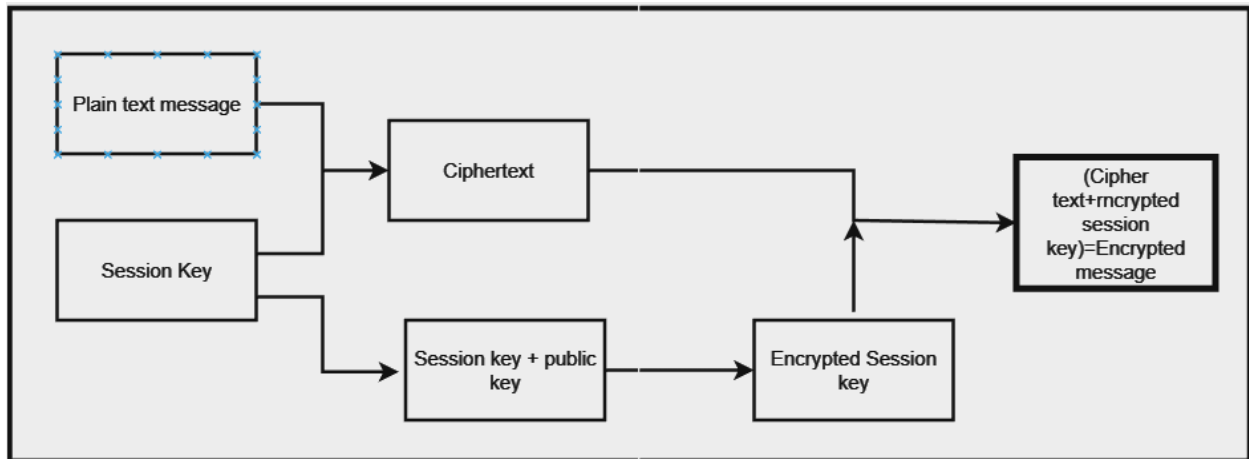


Figure 1 PGP Flowchart.

5.2 Encryption Techniques.

There are two stages of encryption that are performed in this research.

5.2.1 Advanced Encryption Standard.

The Advanced Encryption Standard (AES) is a highly secure symmetric block cipher utilized to encrypt data in fixed 128-bit blocks. AES operates through a substitution-permutation network, performing a sequence of connected operations like Sub-Bytes, Shift-Rows, and Mix-Columns.

The number of rounds varies depending on the key size: 10 rounds for a 128-bit key, 12 rounds for a 192-bit key, and 14 rounds for a 256-bit key. AES has undergone thorough peer review, ensuring its reliability and long-term viability. Being compatible with multiple platforms, AES enjoys widespread adoption and global recognition. The compatibility is the major factor that led to choosing this encryption method.

In the proposed method we will use the AES method to encrypt the file at the Fog node. The steps below explain the process that takes place during the encryption of the data before sending to cloud. AES is highly regarded as the finest encryption method due to its exceptional security, standardization, efficiency, and flexibility.

AES operation sequence

- **Key expansion:** The original key expands into 14 distinct keys for encryption and decryption.
- **SubBytes:** Each byte within the input block is replaced with a different byte, enhancing data mixing.
- **ShiftRows:** The input block's rows shift by varying amounts, further scrambling the data.
- **MixColumns:** Mathematical operations combine the input block's columns, enhancing data diffusion.

Upon completing these stages, the output block becomes the encrypted version of the input block. During decryption, the process reverses using the same 14 keys. AES's strength lies in its Key size and the number of rounds, with AES-256 providing the highest encryption level ((AES), n.d.).

5.2.2 Session Key Encryption.

RSA is the method chosen for session key encryption. RSA is an asymmetric cryptographic algorithm, is employed to encrypt and decrypt data using a public key and a private key. Its robustness lies in the difficulty of factoring large numbers, rendering it highly secure for various applications like secure email, file transfer, and digital signatures (RSA, n.d.).

The RSA process unfolds as follows:

Key Generation: Users generate two large prime numbers, p and q . From these primes, they calculate $n = p * q$, serving as the modulus for both public and private keys. A public exponent e , coprime to $(p-1)*(q-1)$, is chosen. The private exponent d is calculated such that $e * d \equiv 1 \pmod{(p-1)*(q-1)}$. The public key is represented as (n, e) , and the private key as (n, d) .

Encryption: To encrypt the plaintext message m for the recipient, the sender employs the recipient's public key (n, e) . The message m is converted into an integer ($m < n$). Encryption ensues using the public key with the formula: $c \equiv m^e \pmod{n}$. The resulting ciphertext c is then transmitted to the recipient.

Decryption: The recipient uses their private key (n, d) to decrypt the ciphertext c . Decryption proceeds with the formula: $m \equiv c^d \pmod{n}$, unveiling the original plaintext message m sent by the sender.

RSA's security thrives on the challenge of factoring the product $n = p * q$ for large prime numbers. So long as the private key and factors p and q remain undisclosed, RSA endures as a strong and widely adopted method.

5.2.3 Secure Channel.

The third phase of security is approached by the use of IPSEC VPN tunnel (*IPsec VPN Cisco Press*, n.d.). IPsec VPN's are one of the most widely used Tunneling techniques in the industry. IPsec comprises various encryption methods and component technologies, yet its operation can be broken down into five primary steps:

Initiating the IPsec Process: The IPsec process begins when specific traffic is considered interesting based on the IPsec security policy configured in the peers. This triggers the IKE process.

IKE Phase One: During this phase, IKE authenticates the IPsec peers and negotiates IKE Security Associations (SAs). These SAs establish a secure channel for further IPsec SA negotiation.

IKE Phase Two: In this phase, IKE negotiates IPsec SA parameters and creates matching IPsec SAs in the peers. These SAs define encryption methods, security parameters, and authentication algorithms for secure data transfer.

Data Transfer: Data is securely transferred between the IPsec peers using the established IPsec SAs. Encryption and decryption are performed based on the IPsec parameters and stored encryption keys.

IPsec Tunnel Termination: IPsec SAs are terminated either through explicit deletion or by timing out due to inactivity, closing the secure communication channel between the peers.

6. Design and Implementation Specification:

This section of the report will highlight the factors that led to choosing the techniques and methodology that was used to implement the project. While we look at conventional security using any of the most commonly heard latest encryption technique such as ECC or even one of the advanced methods like homomorphic encryption, which might cause issues such as Complexity, Performance, Usability, and Interoperability issues. The perspective of this research was to use a combination of techniques which are widely used in industry and combine them together to form a better technique. Another Important factor that supports the decision is that the chosen techniques AES and RSA are widely used, and the existing software and hardware infrastructure would support these techniques. Focus was to suggest a method that would easily blend into the existing system without performance or compatibility issues.

AES is highly secure, efficient, and standardized for compatibility. Its versatility offers key sizes of 128, 192, or 256 bits, ensuring data confidentiality. AES's proven resilience to attacks makes it a widely adopted encryption standard. While RSA's asymmetric encryption enables secure key exchange and digital signatures without sharing private keys. Its security relies on the challenge of factoring large numbers. RSA's public/private key pair enhances secure communication and data protection. The architecture of the proposed system design is as shown below.

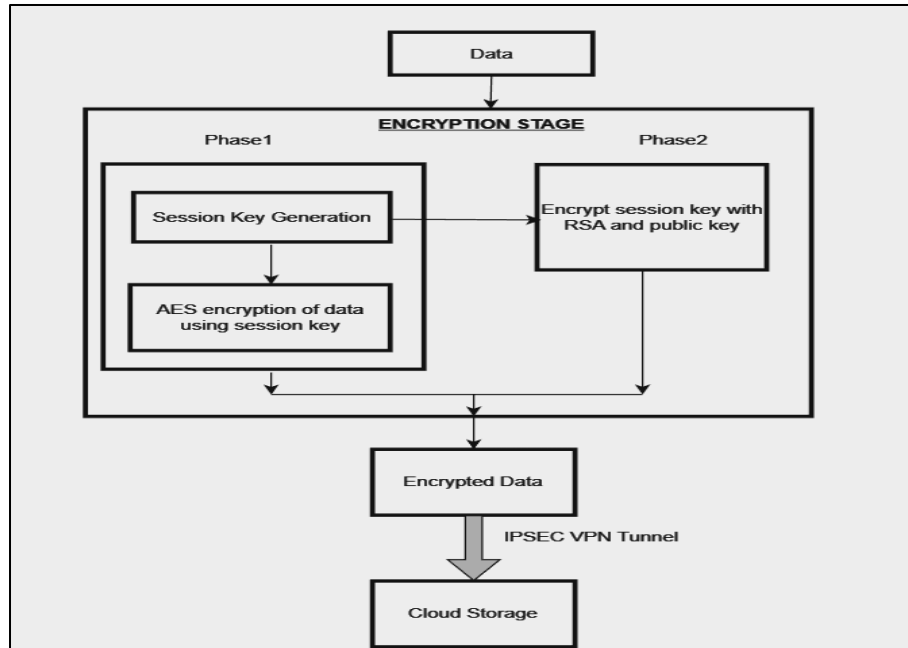


Figure 2 Architecture diagram

The architecture of the proposed system design as shown in the proposed system as shown in figure2 describes how the systems operate at each stage. The process begins when the data to be stored reaches the FOG node which performs encryption. Initially the key is generated, then phase one the symmetric key encryption is done. At the phase 2 the encryption key is encrypted using the RSA, at the end of encryption stage both the encrypted file and key is merged to form a hybrid file. The next phase is where the secure VPN tunnel is established and to the cloud storage device, to transport the file over the secure medium. We will not be performing any decryption at the storage node since the purpose of the work is to ensure security of data stored at cloud storage. The data will be decrypted only within the fog node inside the company infrastructure and not in a public domain. This reduces the risk of an attacker compromising the public key to decrypt the data when in transit or when stored in public domain.

7. Implementation

The table below shows the hardware configuration and software setting made on the host machine to host the experimental setup. The physical machine is an HP Pavilion machine with the below configurations.

System Configuration	
Operating system	Microsoft Windows 11 Home Single Language/64bit
Processor	AMD Ryzen 5 3550H- 2100 Mhz, 4 Core(s), 8 Logical Processor(s)
Ram	16 GB
Virtual Memory	9.45 GB
Graphics card	4GB
Virtualization	Enabled at Bios

Figure 3 System Configurations

The figure given below describes the infrastructure designed on the GNS3 platform.

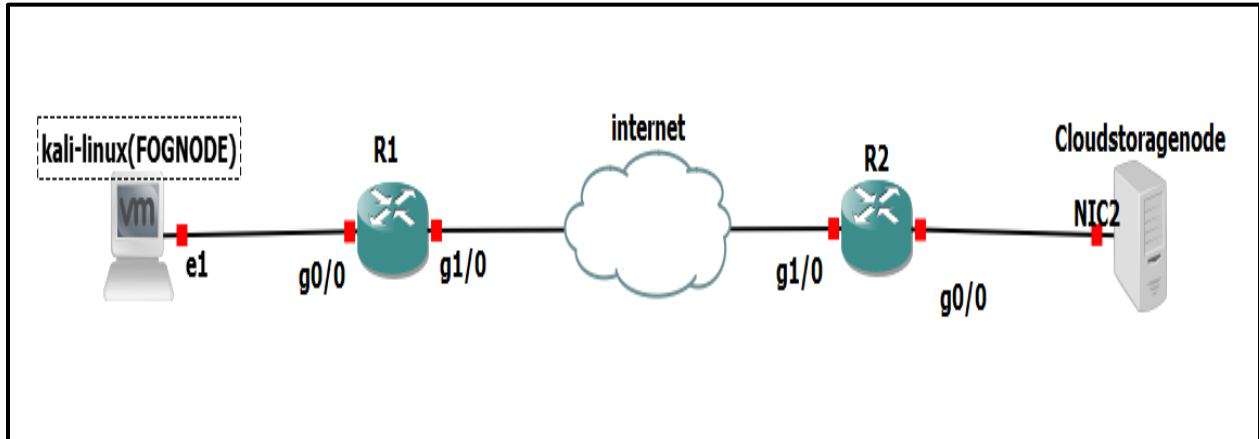


Figure 4 Infrastructure Setup.

The next table shown below describes the software components that was used to build the project setup on the host machine.

Infrastructure components	
GNS3	Version 2.2.41
VMware Workstation pro 16	Version 16.2.4 build-20089737
kali-Linux(FOG NODE)	kali linux machine serving role of fog compute device at the customer environment that performs the encryption.
R1	Cisco c7200 router, acting at edge router at the Customer environment
R2	Cisco c7200 router, acting at edge router at the Customer environment
Cloud Storage Node	Windows machine serving role of cloud storage device at the CSP environment.

GNS3: The implementation is carried out in an isolated GNS3 environment. GNS3 is one of the tools used in industry during the design stage of an infrastructure. The tools give real-time results helping professionals evaluate the design of any proposed system. The GNS3 is running as a VM on the VMware as this is recommended method when simulating multivendor devices and keep the devices isolated from the normal operation of host machine.

VMware Workstation pro: VMware workstation is a hosted hypervisor tool that can be run on both windows and Linux environments. The application allows individuals to host and operate multiple virtual machines on a physical machine.

Kali Linux: This is a Linux operating system that is hosted as a virtual machine in GNS3 performing the role of a FOG node that is present in the client network. The Operating system comes with built-in packages with a wide variety of functions such as encryption tools, penetration testing tools, scanning and many other utilities.

R1 and R2: These are Cisco C7200 routers, both these routers act as the devices at the network edge for Client hosted FOG network and Cloud Service provider (CSP) network respectively. The

IPsec VPN tunnel is built between these routers assuming them to be in different network connected over the internet.

Cloud Storage Node: This is a windows 10 Virtual machine hosted in GNS3 virtual machine. This device acts as the Storage device hosted in cloud environment. The encrypted file sent over VPN is received and stored here and fetched back by fog node when needed to be decrypted. This node will not perform any decryption, this is done to restrict need for the keys to leave the FOG environment, adding additional security to the data.

8. Evaluation

The model has been successfully implemented and will describe the steps evaluating if the system is operating as expected.

8.1 Verifying system operation.

The kali Linux machine has successfully performed an AES 256 encryption on the file and the private and public key has also been encrypted. Kali Linux made use of the GNUPG tool to perform the encryption of data. We have used a nonpublic data set for testing the environment. The first screen shot would show the content of the data in plain text.

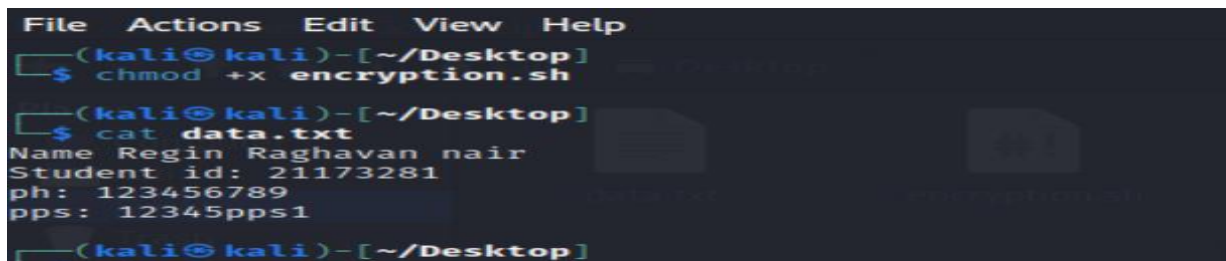
A terminal window screenshot from a Kali Linux machine. The prompt is '(kali@kali)-[~/Desktop]'. The user enters '\$ chmod +x encryption.sh'. The prompt changes to '\$ cat data.txt'. The output shows: 'Name Regin Raghavan nair', 'Student id: 21173281', 'ph: 123456789', and 'pps: 12345pps1'. The prompt returns to '(kali@kali)-[~/Desktop]'.

Figure 5 Plaintext data

GnuPG is a free opensource package installed in kali that can be used to implement OpenPGP encryption.it can be used to perform wide range to tasks such as encryption, digital signatures, and key management. The following steps were performed using GnuPG to encrypt the data.

```
# GPG key pair generation (public and private keys)
gpg --gen-key
# A 256-bit AES encryption key is generated"
openssl rand -out aes_key.bin 32
# Encrypt the data.txt file using AES256 .
gpg --symmetric --cipher-algo AES256 --output encrypted_data.gpg data.txt
# Encrypt the AES encryption key "aes_key.bin" with the public key.
gpg --encrypt --recipient reginnair@gmail.com --output aes_key.bin.gpg aes_key.bin
# Concatenate the encrypted data and the encrypted AES key into a single file for storage at cloud.
cat encrypted_data.gpg aes_key.bin.gpg > combined_file.gpg
```

The concatenated file is encrypted to random encrypted contents. The file is named a “combined_file.gpg” . upon trying to retrieve the content of the file using cat we would see that the file has been encrypted and all we could see is scrambled file with random values as its content

as shown in figure below.

```
(kali@kali)-[~/Desktop]
└─$ cat combined_file.gpg
Le'ee;+l+D+++Zx*3+mh!+z*x1*#zk+e+ h+*+i+/8+?+A+++++o+_+e+e+; :1F++ Ag+++>+<j*!Ki++ 变 +e+eRe]+++++H++
++8+rw+Re++++,++G+u+v++++E;d+ ++|++++y+e+++++l*2+;eC {}gCBK +*P+j_l`*+]++++s+j+++++A"+:;n_+++pm7+++S@+++8u++-+
+N+f+5
+|k+U+m+5+++E
+|w++++y+++Sc/) ++6mN5Cc+++t+I:++s+ N+~|++ 2u++++v#I+="++J;f+s+++++H+e+|++j6j+++++
+o
+<I+vr+Qy+vy+|^
+-+^5+e+e+.|++++\|^*B++2+r+散+Lkq++`bUL+Z2+p+++P+.]C++t^Q+i
+e+eI+++ (Seed+e+e+| +++5+++++U+0r{_a+N2+jA6+zF+j*1++v+++u+MHP+
+++E+q'0++Wl+++l+uk+Vp++
(kali@kali)-[~/Desktop]
└─$
```

Figure 6 encrypted data.

We have verified that the data is retrieved by decrypting the file by retrieving the key from the concatenated file and decrypting AES to access the original data

The second phase of security that was implemented as part if this was the IPsec Tunnel on the cisco router. It has been verified from the logs seen on router that the VPN session was established when trying to send data from FOG node to cloud storage device. Figure one shows the successful establishment of VPN between both the edge router.

```
R2#sh crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst          src          state         conn-id status
12.1.1.4     12.1.1.2     QM_IDLE       1001 ACTIVE
IPv6 Crypto ISAKMP SA
```

Figure 7 Active VPN session

8.2 Evaluation

There are a wide variety of methods that are used in the industry for encryption of data. The system has been tested against other Symmetric encryption methods such as Blowfish and 3des in conjunction with RSA to encrypt the encryption key to evaluate if the combination of AES and RSA is an efficient method for PGP. The test has been performed on a wide range of data sizes from 256kb to 3 mb. The table below shows the encryption performance of the system.

Size/time(ms)	Blowfish+RSA	3DES+RSA	AES+RSA
256kb	0.908	0.896	0.503
512kb	1.075	1.113	1.017
1mb	1.121	1.118	1.126
3mb	1.817	1.543	1.55

Figure 8 Encryption performance

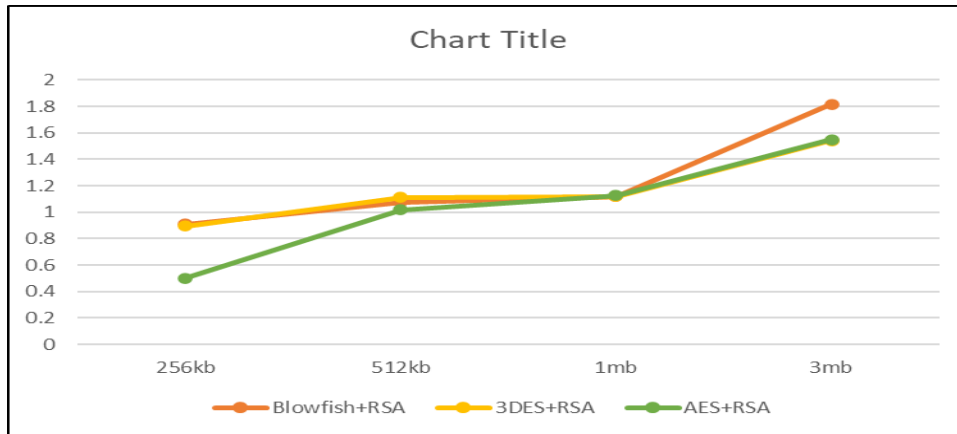


Figure 9 performance graph

The AES+RSA pairing consistently exhibits the quickest performance across all file sizes, with encryption durations of 0.503 ms, 1.017 ms, 1.126 ms, and 1.55 ms respectively. While Blowfish+RSA and 3DES+RSA demonstrate competitive speed, the AES+RSA pairing underscores its effectiveness and appropriateness for safeguarding data across different file sizes.

9. Conclusion and Discussion

The research aimed on developing a system which is capable of performing encryption that is not a conventional method that would strengthen the security of data stored in the cloud platform.

Literature gap1 solved: The system is a combination of well-known and strong encryption techniques like AES and RSA. The main reason for choosing these techniques are that it is proven to be nearly impossible to be cracked and secondly being one of the encryption techniques, where new an old systems (software and hardware) would be well compatible with these technology and can be integrated to an existing infrastructure without businesses having to invest time, resource and money into upgradation of infrastructure.

Literature gap2 solved: The second level of security is the Site-to-Site IPsec VPN tunnel which again equipped the system to have an AES-256 encrypted and hashed medium for data to travel than being sent on open internet.

The combination of the above two techniques leads to an outcome, a system that ensure the “security of data at rest and in transit” which was not focused on in most of the research and in real time at organization where sole dependency is on the encryption technique to protect critical data.

While cloud has become one of the booming technologies and most organizations are migrating towards the cloud infrastructure, there is continuous research being done to strengthen the security if data and ensuring Confidentiality, integrity, and Authenticity of the data to the utmost degree. The proposed system has proven to be reliable and ensures the security of data at any state, be it at rest or in transit. While the proposed system has been proven to be significantly impactful in securing data, the success of an unbreakable system is looking for future scope of development. At the next stage, the future scope of work would be,

- 1) Extensive testing of the system in real-world environment where speed of data generation and transfer of data is at tremendous speed. They would enable us to study the performance, reliability, and scalability of the proposed system outside the sandbox environment.
- 2) Study should also be made to enhance the system by adding functionality like user friendly interface allowing an authorized non-technical person being able to encrypt the data and send it to cloud, while enabling the system securely to store the keys and certificates generated in a secured isolated storage, retrievable only by authenticated personals.
- 3) Comparing the proposed system with other advanced ML-based encryption system is Homomorphic Encryption or AI based neural crypt to evaluate the performance in comparison with the latest techniques in cyber world.

10. References

- 1) Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2018). A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys*, 51(4). <https://doi.org/10.1145/3214303>
- 2) (AES). (n.d.). Retrieved August 3, 2023, from <https://www.geeksforgeeks.org/advanced-encryption-standard-aes/?ref=lbp>
- 3) Alhumam, N. A., Alyemni, N. S., & Hafizur Rahman, M. M. (2023). Cyber Security in Fog Computing Using Blockchain: A Mini Literature Review. *5th International Conference on Artificial Intelligence in Information and Communication, ICAIIC 2023*, 553–557. <https://doi.org/10.1109/ICAIIIC57133.2023.10066994>
- 4) Aljumah, A., & Ahanger, T. A. (2018). Fog computing and security issues: A review. *2018 7th International Conference on Computers Communications and Control, ICCCC 2018 - Proceedings*, 237–239. <https://doi.org/10.1109/ICCCC.2018.8390464>
- 5) Guo, W., Dong, X., Cao, Z., & Shen, J. (2017). *Efficient Attribute-Based Searchable Encryption on the Cloud Storage*. <https://eprint.iacr.org/2017/782.pdf>
- 6) *How PGP works*. (n.d.). Retrieved August 3, 2023, from <https://users.ece.cmu.edu/~adrian/630-f04/PGP-intro.html>
- 7) *IPsec VPN Cisco Press*. (n.d.). Retrieved August 3, 2023, from <https://www.ciscopress.com/articles/article.asp?p=24833&seqNum=6>
- 8) Karra, M., Ramadas, M., & Mishra, V. P. (2019). Overview of Cloud Fog Computing Paradigms. *Proceedings of 2019 International Conference on Computational Intelligence and Knowledge Economy, ICCIKE 2019*, 392–396. <https://doi.org/10.1109/ICCIKE47802.2019.9004404>
- 9) Kiwelekar, A. W., Patil, P., Netak, L. D., & Waikar, S. U. (2021). Blockchain-based security services for fog computing. *Advances in Information Security*, 83, 271–290. https://doi.org/10.1007/978-3-030-57328-7_11/COVER
- 10) Kurniawan, Y., Albone, A., & Rahyuwibowo, H. (2011). The design of mini PGP security. *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics, ICEEI 2011*. <https://doi.org/10.1109/ICEEI.2011.6021726>
- 11) Mukherjee, M., Matam, R., Shu, L., Maglaras, L., Ferrag, M. A., Choudhury, N., & Kumar, V. (2017). Security and Privacy in Fog Computing: Challenges. *IEEE Access*, 5, 19293–19304. <https://doi.org/10.1109/ACCESS.2017.2749422>
- 12) PGP - Pretty Good Privacy. (n.d.). In *Security in E-Learning* (pp. 157–166). Springer-Verlag. https://doi.org/10.1007/0-387-26065-X_11

- 13) *Pretty Good Privacy*. (n.d.). Retrieved August 1, 2023, from <https://users.ece.cmu.edu/~adrian/630-f04/PGP-intro.html#p10>
- 14) Rehman, S., Talat Bajwa, N., Shah, M. A., Aseeri, A. O., & Anjum, A. (2021). Hybrid AES-ECC Model for the Security of Data over Cloud Storage. *Electronics 2021, Vol. 10, Page 2673, 10(21)*, 2673. <https://doi.org/10.3390/ELECTRONICS10212673>
- 15) Rezapour, R., Asghari, P., Javadi, H. H. S., & Ghanbari, S. (2021). Security in fog computing: A systematic review on issues, challenges and solutions. *Computer Science Review, 41*, 100421. <https://doi.org/10.1016/J.COSREV.2021.100421>
- 16) *RSA*. (n.d.). Retrieved August 3, 2023, from <https://www.tutorialspoint.com/what-are-the-steps-in-rsa-in-information-security>
- 17) Sadkhan, S. B. (2021). Elliptic Curve Cryptography-Status, Challenges and Future trends. *Proceedings of the 7th International Engineering Conference "Research and Innovation Amid Global Pandemic", IEC 2021*, 167–171. <https://doi.org/10.1109/IEC52205.2021.9476090>
- 18) Samyururai, A., Revathi, K., Prema, P., Arulmozhiarasi, D. S., Jency, J., & Hemapriya, S. (2015). Secured Health Care Information exchange on cloud using attribute based encryption. *2015 3rd International Conference on Signal Processing, Communication and Networking, ICSCN 2015*. <https://doi.org/10.1109/ICSCN.2015.7219826>
- 19) Shruti, & Rani, S. (2022). Mitigating Security Problems in Fog Computing System. *Lecture Notes in Networks and Systems, 419 LNNS*, 612–622. https://doi.org/10.1007/978-3-030-96299-9_58/FIGURES/1
- 20) Soman, V. K., & Natarajan, V. (2017). An enhanced hybrid data security algorithm for cloud. *2017 International Conference on Networks and Advances in Computational Technologies, NetACT 2017*, 416–419. <https://doi.org/10.1109/NETACT.2017.8076807>
- 21) Sood, S. K. (2012). A combined approach to ensure data security in cloud computing. *Journal of Network and Computer Applications, 35(6)*, 1831–1838. <https://doi.org/10.1016/J.JNCA.2012.07.007>
- 22) Wang, C., Wu, J., Yuan, Y., & Liu, J. (2018). Insecurity of Cheng et al.'s efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage. *Proceedings - 15th IEEE International Symposium on Parallel and Distributed Processing with Applications and 16th IEEE International Conference on Ubiquitous Computing and Communications, ISPA/IUCC 2017*, 1387–1393. <https://doi.org/10.1109/ISPA/IUCC.2017.00210>
- 23) Wang, T., Zhou, J., Chen, X., Wang, G., Liu, A., & Liu, Y. (2018). A Three-Layer Privacy Preserving Cloud Storage Scheme Based on Computational Intelligence in Fog Computing. *IEEE Transactions on Emerging Topics in Computational Intelligence, 2(1)*, 3–12. <https://doi.org/10.1109/TETCI.2017.2764109>
- 24) Zaw, T. M., Thant, M., & Bezzateev, S. V. (2019). Database Security with AES Encryption, Elliptic Curve Encryption and Signature. *Wave Electronics and Its Application in Information and Telecommunication Systems, WECONF - Conference Proceedings*. <https://doi.org/10.1109/WECONF.2019.8840125>
- 25) Zhang, L., Zheng, Y., & Kantoa, R. (2016). *A Review of Homomorphic Encryption and its Applications*.
- 26) Zhang, P. Y., Zhou, M. C., & Fortino, G. (2018). Security and trust issues in Fog computing: A survey. *Future Generation Computer Systems, 88*, 16–27. <https://doi.org/10.1016/J.FUTURE.2018.05.008>