*Secure text transfer using Diffie-Hellman key Exchange Based on Cloud*

Abhay tomer

Apex Institute of Technology, Chadigarh University

Mohali, Punjab

20BCS3566@cuchd.in

***Abstract*— This paper proposes a system for securely, efficiently, and flexibly storing data in the cloud, which utilizes public-key cryptosystems that provide fixed-size ciphertext classes for high-performance decryption. The data owner can release a fixed-size single key, while keeping the remaining encrypted files confidential, which can be sent to others or stored in a limited number of storage devices. To improve the security of the method, the system uses the AES and Diffie-Hellman Key Exchange Algorithm. Cloud computing has revolutionized the way we store data, but security remains a critical concern. The Diffie-Hellman key exchange protocol is an asymmetric key algorithm used for secure key exchange, allowing two parties to establish a shared secret key over an insecure communication channel without prior communication. By using this algorithm and strong encryption techniques, such as AES or RSA, a secure channel can be established for transferring text and files over the cloud.**

***Keywords—Diffie Hellman, Key exchange, Cloud security.***

1. INTRODUCTION

Cloud computing has become a trendy method of storing and accessing data, but concerns regarding unauthorized access by malevolent entities pose a significant security threat. In this context, the Diffie-Hellman key exchange algorithm has emerged as a viable cryptographic protocol for establishing a shared secret key between two parties over an insecure network, which can be utilized to encrypt and decrypt data, ensuring confidentiality, integrity, and authenticity of the transferred data. By utilizing the Diffie-Hellman algorithm for data transfer in the cloud, sensitive information can be securely transmitted between the client and cloud server.

One of the primary advantages of the Diffie-Hellman algorithm is that it provides a secure means of key exchange without the need for pre-shared keys or complex public key infrastructure (PKI). This makes it an appropriate solution for secure data transfer in the cloud, where there may be multiple users accessing the same data.

However, like any cryptographic algorithm, the Diffie-Hellman key exchange has its limitations and potential vulnerabilities that need to be addressed. These include the vulnerability to man-in-the-middle (MITM) attacks, weak key exchange parameters[2], lack of authentication, implementation flaws, and the potential threat of quantum computing.

MITM attacks can compromise the security of the key exchange by allowing an attacker to intercept communication between two parties and establish separate key exchanges with each party. Weak key exchange parameters can make the key exchange susceptible to attacks such as brute-force or discrete logarithm attacks[1]. Lack of authentication means parties cannot verify each other's identities, making it vulnerable to impersonation or spoofing attacks[3].

Implementation flaws or vulnerabilities in the software or systems used for the key exchange can compromise the security of the exchanged keys. With the advent of quantum computing, it is possible that certain quantum algorithms could potentially break the security of Diffie-Hellman key exchange, posing a threat to the security of communications that rely on this algorithm in the future.

To mitigate these risks, appropriate security measures need to be implemented, such as using strong key exchange parameters, authenticating the parties, and keeping software and systems up-to-date. Additionally, considering alternative cryptographic algorithms, such as post-quantum cryptography, that may be resistant to quantum computing. It can be a prudent approach for future security considerations.

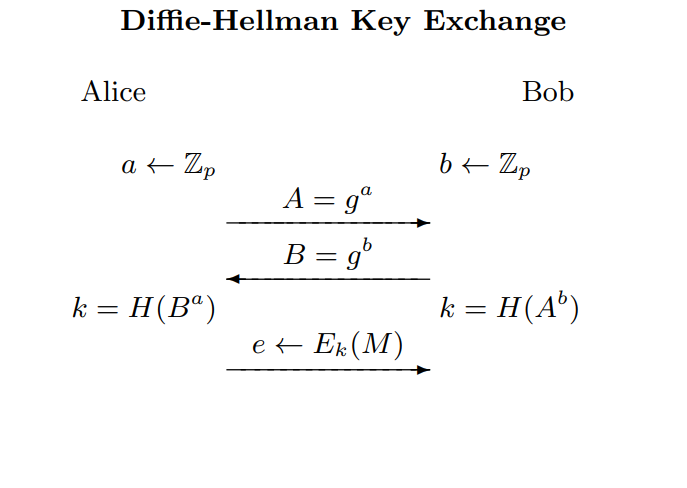
In conclusion, while the Diffie-Hellman key exchange algorithm provides a secure means of key exchange, it is not immune to vulnerabilities and limitations that need to be taken into consideration. By implementing appropriate security measures and being aware of potential risks, cloud computing users can mitigate the risks of unauthorized access and protect sensitive data from malicious entities

1. DIFFIE-HELLMAN KEY EXCHANGE
2. INTRODUCTION

The need for a key exchange protocol to securely deliver encryption keys over an insecure communication channel between two parties have been a concerning issue since the dawn of cryptography in the Egypt. It emphasizes the importance of secure transmission to prevent unauthorized access or accidental disclosure of data during transmission which can be detrimental. This article explains the key agreement protocol that derives the session key from information shared by both parties and the basic transport protocol that generates a session key and sends it securely to the other party.

The Diffie-Hellman key exchange protocol is introduced as one of the first key exchange protocols, which allows two parties to exchange a session key based on public parameters shared in the initialization stage. The article highlights the efficiency of the Diffie-Hellman protocol and its vulnerability to man-in-the-middle attacks.It also discusses recent Authentication Key Exchange (AKE) protocols, their potential attacks, and their resistance to known attacks such as the Key Compromise Impersonation (KCI) attack and the Ephemeral Key Compromise (EKC) attack. Additionally, the article emphasizes the need to balance cost/efficiency and security while choosing key establishment protocols and recommends avoiding public key cryptographic methods such as RSA, ECC, and ElGamal to achieve low processing costs. The Station-to-Station (STS) AKE Protocol is presented as one of the earliest authenticated key exchange protocols. Finally, secure and efficient key exchange protocols are introduced, and their vulnerabilities are described.

1. ALGORITHM

A large prime number p

A generator value g, such that 1 < g < p

Alice's secret number a, such that 1 < a < p-1

Bob's secret number b, such that 1 < b < p-1

An optional hash function H() for additional security

Outputs:

A shared secret key K

Steps:

1. Alice generates a random secret number a, such that 1 < a < p-1
2. Bob generates a random secret number b, such that 1 < b < p-1
3. Alice calculates A = g^a mod p
4. Bob calculates B = g^b mod p
5. Alice sends A to Bob, encrypted using a secure encryption scheme (e.g., RSA)
6. Bob sends B to Alice, encrypted using the same encryption scheme
7. Alice decrypts B using her private key and calculates K = B^a mod p
8. Bob decrypts A using his private key and calculates K = A^b mod p

Alice and Bob now share the secret key K

Alice and Bob can also use hash function on K using a secure hash function H() to produce a shared secret key K'

Alice and Bob use K' as the encryption key for their messages to each other[4].

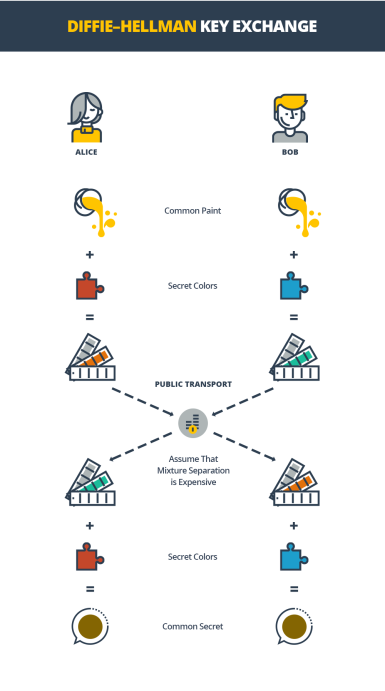


Fig 1 Representation of Diffie Hellman Key Exchange

Reference: <https://cryptoquantus.com/2019/09/17/ix-diffie-hellman-merkle-key-exchange/>

To initiate the process, two individuals, herein referred to as Alice and Bob, must first agree upon a color. This color, while not requiring secrecy, must differ on each occasion. For the purposes of this explanation, let us assume that they have decided upon the hue of yellow. Subsequently, each party independently selects a private color which they do not disclose to the other. In our example, Alice has opted for the shade of orange, while Bob has chosen a blue-green tint. The next step involves each of them mixing their respective private colors with the jointly agreed upon color, resulting in Alice obtaining an orange-tan blend and Bob obtaining a light blue mixture. The two parties now exchange their mixed colors with each other publicly.

Finally, both parties mix the color they have received from their counterpart with their respective private color. The outcome is a repugnant, yellow-brown concoction that is identical for both Alice and Bob. If a third party endeavored to intercept and eavesdrop on the exchange of colors, it would be arduous to determine the private color of each individual, thereby rendering it impossible to replicate the final mixture. In actuality, this process would entail the use of large numbers instead of colors, as computers can execute the required calculations expeditiously.

1. DRAWBACKS

The Diffie-Hellman method of key exchange, while an amazing cryptographic tool, it has a significant flaw in the form of an authentication mechanism. Its absence has caused anxiety among security specialists and academics alike. As a result, the algorithm's capacity to verify the validity of key exchange participants has been put into doubt. As a result, it may be vulnerable to numerous forms of assaults, such as man-in-the-middle attacks, which might jeopardise the security and integrity of sent data.

A hostile third party can intercept and manipulate communication between two parties in this form of attack. Because the technique lacks an authentication procedure, the attacker may simply mimic one or both parties and fool them into disclosing their private keys. This can jeopardise the security of the data shared and have serious implications. To counter such attacks, it is critical to employ extra safeguards such as digital certificates or digital signatures or even third party autehntication methodologies such as kerberos.

Diffie-Hellman algorithm can be resource intensive, resulting in higher resource usage and longer processing times. Due to its complex mathematical calculations, it may not be the most efficient algorithm for certain scenarios, particularly those involving limited computational resources. Additionally, the algorithm's intensive processing requirements can impact CPU performance time, potentially slowing down other important processes. Thus, while the Diffie-Hellman algorithm is a valuable tool for secure key exchange, its high computational demands should be carefully considered in order to ensure optimal performance.[5]

Diffie-Hellman algorithm cannot be used for encryption of information. Diffie-Hellman algorithm cannot be used for encryption of information. It only provides a secure way to exchange keys between two parties. After the key exchange, a separate encryption algorithm must be used to encrypt the actual data being transmitted. This limitation is because the Diffie-Hellman algorithm is only concerned with generating a shared secret key, but does not provide any mechanism for encrypting or decrypting data.

Digital signature cannot be signed using Diffie-Hellman algorithm. The Diffie-Hellman algorithm, while useful for key exchange, cannot be used for digital signature generation. Digital signature schemes require the use of public key cryptography, which is a different cryptographic technique altogether. Digital signatures provide a way to verify the authenticity of messages and ensure that they have not been tampered with during transmission. While Diffie-Hellman can be used in conjunction with digital signature schemes, it cannot be used on its own to generate digital signatures. Instead, other algorithms such as RSA and DSA are typically used for digital signature generation.

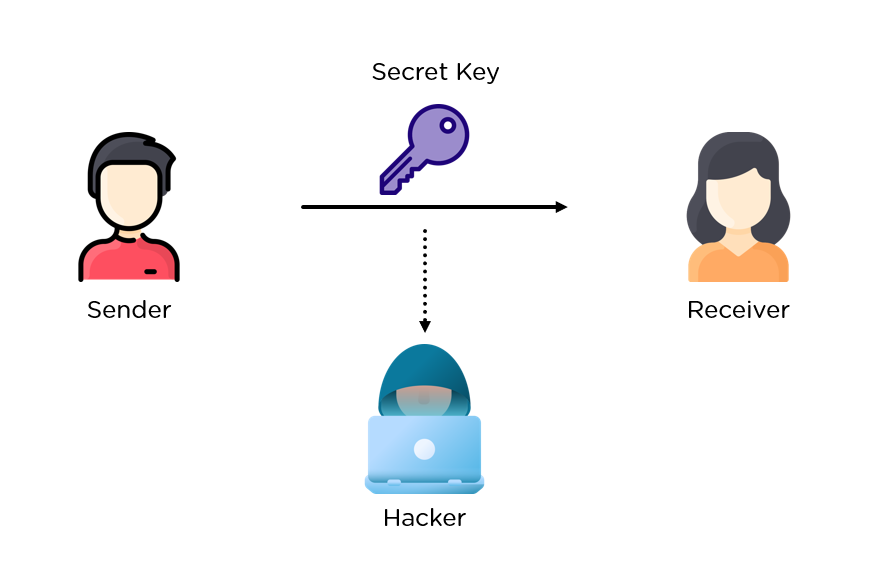


Fig 2 Man In The Middle Attack

Reference: <https://www.simplilearn.com/ice9/free_resources_article_thumb/intro_2.PNG>

There are several number theory-related attacks that can be launched against Diffie-Hellman key exchange such as

Small subgroup attack: This attack targets the structure used in the Diffie-Hellman key exchange, primarily admin groups. If the group has a small group (for example, a small group), the attacker can use the Pohlig-Hellman algorithm to solve the problem of inequality of the logarithm in the group, thereby revealing the information secret. To prevent this attack, it is recommended to use groups with large leaders or without small groups, such as groups of elliptic curves.

Exponential Analysis Attack: This attack is a general technique for solving the split logarithm problem in cyclic groups. It is based on the observation that each element in a cyclic group can be expressed as generator power and the logarithm of each element can be expressed as the combination of the logarithms of the generators. This attack is more effective than the general purpose small-level large-step algorithm, especially for larger orders. To prevent such attacks, it is recommended to use large groups of leaders or groups of elliptic curves with large groups of leaders.

Quadratic remainder stop: This argument uses the fact that mod p has exactly two roots for the prime number p and the quadratic remainder a (i.e., integer squared modulo p). An attacker can use this tool to reduce logarithm inequality problems to simple quadratic problems that can be solved efficiently using the Tonelli-Shanks algorithm. To prevent this attack, it is recommended to use the prime number p so that p-1 has a large value and makes the quadratic problem more significant. [6]

The drawbacks of the Diffie-Hellman key exchange algorithm are numerous. The lack of an authentication procedure and vulnerability to man-in-the-middle attacks make it less secure than other modern encryption methods. Furthermore, the algorithm is computationally intensive, leading to high resource and CPU performance costs. It is limited to symmetric key exchange, and encryption of information and digital signature cannot be performed using this algorithm. Therefore, while the Diffie-Hellman algorithm was groundbreaking in its time, its limitations have been exposed, and modern encryption methods have since surpassed it in terms of security and efficiency.

1. CONTEMPORARY SYSTEMS
2. INTRODUCTION

Modern cryptographic systems are utilizing the Diffie-Hellman key exchange algorithm to enable secure and efficient communication. The use of elliptic curve cryptography (ECC) in combination with Diffie-Hellman, known as ECDH, is one of the latest developments in this area[7]. This approach provides a high level of security with smaller key sizes, which leads to reduced computational and storage requirements. Another noteworthy scheme is the DHIES protocol, which combines Diffie-Hellman with symmetric encryption to ensure both data privacy and authenticity[8]. The MQV protocol, which extends the basic Diffie-Hellman key exchange to allow for mutual authentication, has also gained popularity in recent years. Furthermore, zero knowledge proofs such as ZKP-DL, are being used in conjunction with the Diffie-Hellman key exchange to authenticate participants without revealing any information about their private keys. These advancements underscore the continuing importance and relevance of the Diffie-Hellman key exchange algorithm in contemporary cryptographic systems.

1. SECURE SOCKET LAYER

SSL (Secure Sockets Layer) protocol is a widely used encryption protocol that provides secure communication on the Internet. The system is based on a combination of public-key and symmetric-key cryptography, which has become an essential part of modern network security. SSL provides secure communication between clients and servers by protecting sensitive information such as passwords, credit card numbers and other personal information from unauthorized access. SSL uses a combination of asymmetric and symmetric key encryption to create a secure connection between client and server. When a client initiates a connection with a server, the server sends it an SSL certificate containing its public key[9].

The client then generates a session key and encrypts it with the server's public key so that only the server can decrypt the key. This session is then used for other communications between the client and server, allowing for fast and efficient data transfer. One of the main features of SSL is that it offers end-to-end encryption, keeping data private and tamper-proof in transit. SSL also provides authentication by allowing the client to communicate with the intended client and not a fraudster. This is done by verifying the SSL certificate presented by the server and making sure it is issued by a trusted authority.

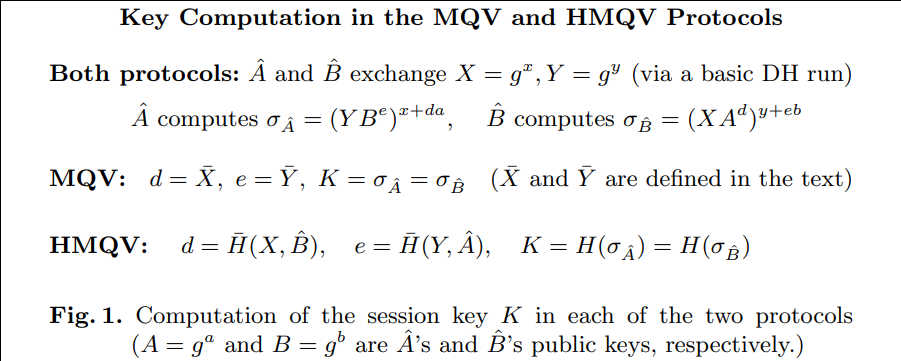
Deffi-Hellman key exchange is an essential part of SSL that allows generation of secure keys for data encryption. Using the Diffie-Hellman key exchange, SSL protects the same keys used for access by spreading across the network, making it difficult for attackers to intercept and decipher data.

1. ELLIPTIC CURVE

Elliptic Curve Diffie-Hellman (ECDH) is a key exchange protocol that uses the discrete logarithm problem in the context of elliptic curve cryptography. ECDH is similar to traditional Diffie-Hellman, but it uses elliptic curves instead of modular exponentiation. The primary advantage of ECDH is that it provides equivalent security with shorter key sizes compared to traditional Diffie-Hellman. In ECDH, each party generates an elliptic curve private key, which is then used to compute the corresponding public key. The public keys are then exchanged between the parties, and each party computes a shared secret using their private key and the other party's public key. The shared secret can then be used as the basis for a symmetric key encryption algorithm.

ECDH is used in a wide range of applications, including secure communication channels, digital signatures, and cryptographic key management. One notable use of ECDH is in the Transport Layer Security (TLS) protocol, which is used to secure internet communications. TLS uses ECDH to establish a shared secret between the client and server, which is then used to encrypt the data transmitted between them.

ECDH has several advantages over traditional Diffie-Hellman. Firstly, ECDH provides equivalent security with shorter key sizes, which reduces the computational overhead of generating and transmitting keys. Secondly, ECDH is more resistant to attacks based on the discrete logarithm problem, which makes it more secure than traditional Diffie-Hellman. Finally, ECDH is more efficient than traditional Diffie-Hellman, making it better suited for use in applications where computational resources are limited.



Menezes-Qu-Vanstone, is also a key agreement protocol that is an extension of the Diffie-Hellman protocol. Like Diffie-Hellman, it allows two parties to establish a shared secret key over an insecure channel. However, MQV has several advantages over Diffie-Hellman, such as providing protection against man-in-the-middle attacks and allowing the use of short-term public keys. MQV uses elliptic curve cryptography (ECC) to provide stronger security than traditional Diffie-Hellman. ECC allows for smaller key sizes while still maintaining the same level of security as larger key sizes in traditional cryptography. The use of shorter keys allows for faster computation, which is especially useful in resource-constrained environments such as mobile devices[10].

HMQV, A Secure Diffie-Hellman High-Performance Protocol. It is a version of the conventional Diffie-Hellman key exchange protocol designed to improve key exchange security and performance in limited contexts such as wireless sensor networks. HMQV is a high-performance and secure key exchange protocol designed for usage in resource-constrained situations. Its reciprocal authentication, forward secrecy, resilience to side-channel attacks, and rapid key exchange make it a popular choice for wireless sensor networks, mobile devices, and other resource-constrained situations.

MQV also incorporates an added layer of security through the use of a three-pass protocol. This protocol ensures that both parties can verify the authenticity of each other's public keys before the shared secret key is established. This protects against man-in-the-middle attacks, where a third party intercepts and alters communications between the two parties. The use of Diffie-Hellman in MQV is crucial to its key exchange process. It involves both parties agreeing on a set of parameters, including a finite field and a base point on that field. Each party then generates a public-private key pair and calculates a shared secret using the other party's public key and their own private key. This shared secret is then used to derive a symmetric key for secure communication.

Therefore, MQV provides a more secure and efficient method for key exchange than traditional Diffie-Hellman. Its use of elliptic curve cryptography and three-pass protocol make it less vulnerable to attacks and faster in computation. The incorporation of Diffie-Hellman in the protocol allows for secure and efficient key exchange, making it a popular choice in modern cryptography applications.

1. ENCRYPTION SCHEME

DHIES or Diffie-Hellman Integrated Encryption Scheme is a cryptographic scheme that combines Diffie-Hellman key exchange with an integrated encryption algorithm. DHIES allows two parties to use a Diffie-Hellman exchange to generate a shared key that is then used to secure and decrypt messages. This scheme provides an efficient and secure way of exchanging encrypted messages between parties without the need for a public key system. The DHIES scheme uses a distributed network algorithm that uses a combination algorithm to encrypt the actual message, and a Diffie-Hellman key exchange is used to exchange keys for encryption. The encryption algorithm used in DHIES may be a combination encryption algorithm, but the most commonly used algorithms include AES and 3DES.

The use of symmetric encryption allows messages to be encrypted and decrypted quickly and efficiently.The DHIES concept also provides a way to use digital signatures to identify the sender. A digital signature is created using the sender's private key, which can be verified using the sender's public key. This allows the recipient to trust that the message came from the sender and was not tampered with during transmission.

provides an additional layer of security by using Diffie-Hellman key exchange in DHIES. The shared free keys generated by the Diffie-Hellman key exchange are private to the two communicating parties, making it difficult for attackers to intercept and decrypt messages. The DHIES concept also uses a technique called key derivation, which ensures that the symmetric key used for encryption is derived from Diffie-Hellman encryption to be secure and effective.

The DHIES concept now offers a safe and effective way to exchange secrets between parties. Use the Diffie-Hellman switch to provide secure communication and use interoperability to ensure efficient communication. DHIES is used in many applications such as email encryption, file encryption and secure messaging.

1. ZERO KNOWLEDGE PROOF

ZZero-knowledge proof (ZKP) is a cryptographic technique that allows one party to prove to the other party that they have certain information or information without revealing the information itself. Essentially, ZKPs allow one party to prove to another party that they know a secret without revealing the secret itself. This is especially useful where privacy is important and parties want to avoid sharing sensitive information. Deffie-Hellman key exchange is often used in the ZKP process because it allows both parties to create a shared secret without revealing it to others. This secret teaching can be used as a basis for ZKP.

In ZKP protocol using Deffie-Hellman, one party (prover) wants to prove (prove) to the other party that it knows about the shared key without personally revealing the key. An example of a ZKP protocol using Deffie-Hellman is the Schnorr authentication protocol. In this process, the witness and witness agree on a generator g and the prime number p, and each generates a random value of x. The prover then calculates g^x mod p and sends it to the prover, which generates a random match value e and returns it to the prover. The witness then calculates the (x + ae) modality (p-1) and sends it to the witness.

The validator then calculates g^(x+ae) mod p, and if that value matches the original value sent by the prover, the prover makes sure that the proofreader knows the key x. The ZKP protocol using Deffie-Hellman has applications in many areas from secure communication to authentication and authorization in computer networks. They are particularly useful when parties want to prove themselves or have secrets without revealing sensitive information. The ZKP protocol can be used for digital signatures, electronic cash machines, and even cryptographic authentication systems[11].

1. PROPOSED SYSTEM
2. INTRODUCTION

The Diffie-Hellman key exchange has been extensively studied in the field of cryptography since its inception. Research has been focused on various aspects of the technique, including security analysis, operational efficiency, flexibility and continuity, and implementation in various scenarios. The security analysis of the Diffie-Hellman key exchange has been a significant area of research, with researchers identifying various attacks such as man-in-the-middle attacks, micro-attacks, and selected ciphertext attacks, and proposing countermeasures to mitigate them. Methods such as proof of security and data security theory are used to evaluate the security of the system. In addition, studies have been conducted to evaluate the stability of the Diffie-Hellman key exchange in special situations such as malicious individuals, external attacks, or quantum computers.

The effectiveness of the Diffie-Hellman key exchange is also a subject of ongoing research, with researchers proposing many ideas to improve the performance of the process. These include speeding up algorithms[12], using special properties of finite fields or elliptic curves[13], precomputing or caching, parallelization, and hardware acceleration.

The objective of this project is to create a secure channel to transmit and receive various forms of data, including text messages and images. To accomplish this, the Diffie-Hellman and AES encryption methods will be employed to encrypt the data prior to its transmission to the recipient. The data will be transmitted and stored in the cloud, with encrypted channels such as SSL being applied to the channel and key information before it is sent.

It is widely recognized that security risks are constantly increasing. There is no guarantee that exchanging information with someone else will be conducted securely and without the possibility of being hacked or cracked. Therefore, the proposed project aims to explore encryption techniques that can help conceal the true key and enable safe data transmission.

The proposed solution is a cloud-based chat application that utilizes the Diffie-Hellman key exchange algorithm to establish secure communication between users. The app will allow real-time exchange of text messages over the cloud while ensuring data security. The Diffie-Hellman algorithm will be used to facilitate the key exchange process.

The main objectives of the proposed solution are as follows:

> Develop a secure key exchange mechanism using the Diffie-Hellman algorithm for secure communication between users.

> Develop authentication and authorization mechanisms to limit access to legitimate users.

> Create robust management mechanisms for secure storage, distribution, and retrieval of text and files.

> Implement data encryption and integrity mechanisms using symmetric encryption algorithms to ensure secure transmission and storage of messages.

> Design an intuitive user interface that meets security requirements.

> Develop a scalable and robust system architecture capable of accommodating a large number of users while handling network failures or disruptions.

> Ensure compliance with relevant data privacy and security regulations and standards.

The cloud chat application will be developed using both front-end and back-end technologies. Web development frameworks, authentication and encryption libraries, and cloud storage services will all be used. The Diffie-Hellman algorithm will be used to establish shared secret keys, while symmetric encryption algorithms like AES will be used to encrypt and ensure data integrity. Proper authentication and authorization mechanisms will be employed to verify user identities, and key management mechanisms will be developed to ensure secure storage and handling of shared secret keys. Finally, the application will be designed to be user-friendly, efficient, and scalable, with robust error handling and recovery mechanisms to ensure reliability and availability.

1. CLOUD IMPLEMENTATION

The protection of user's data in cloud storage is crucial, and encryption is often used to achieve this. However, it's important to ask whether the data is actually being encrypted when stored in the cloud, and if so, what encryption algorithm and key length is being used. Some cloud service providers (CSPs) provide encryption, but the quality and security of their chosen algorithms may vary. It's essential to ensure that the encryption algorithm being used is secure and able to support various encryption use cases, such as database and file system encryption.

To address this issue, a solution is proposed that involves using a symmetric cryptosystem with a 128-bit key length, such as the Advanced Encryption Standard (AES), in combination with the Diffie Hellman algorithm[14]. This solution offers both speed and computational efficiency in handling encryption of large data volumes. The key length is long enough to provide significant protection while still being computationally feasible.

The user would initiate this encryption solution between their applications and database servers in the cloud. This would involve encrypting the data using the AES algorithm before uploading it to the cloud. Upon decryption at the user's end, the plain text data can be read by requesting applications, but it's never written anywhere on the cloud. This approach offers transparency and ease of integration without requiring changes to the application.

To ensure secure storage of the encryption keys, a physical key management server can be installed on the user's premises. This ensures that the keys remain under the user's control and are never exposed in storage or in transit. For authentication purposes, the Diffie Hellman algorithm is used.

1. AUTHENTICATION

The Diffie-Hellman key exchange is a cryptographic method that enables two parties to establish a shared secret key over an insecure communication channel. This key can be used to encrypt subsequent communications using a symmetric key cipher. The scheme was first published in 1976 by Whitfield Diffie and Martin Hellman, but it had been separately invented a few years earlier by James H. Ellis, Clifford Cocks, and Malcolm J. Williamson within the British signals intelligence agency GCHQ.

The Diffie-Hellman key agreement is an anonymous key agreement protocol, but it can be used as a basis for authenticated protocols and to provide perfect forward secrecy in Transport Layer Security's ephemeral modes.

The function of the Diffie-Hellman Algorithm Authentication Module is as follows:

New Registration for Cloud Service: Companies or users who require various cloud services must register by providing details such as user ID and password. These details are later used to validate whether the user is genuine or not by sending a small text message that includes a key required to create an account over the cloud. The authentication process occurs when a user enters their user ID and password, and a key is sent to their device, which is generated using the D-H Key Exchange. This key is valid for a specific time instance and will get destroyed after that connection is closed.

Using Cloud Service: Whenever a user needs to use the cloud services provided, they must enter their user ID and password. If the user ID and password are correct, a new key is generated using the Diffie-Hellman Key Exchange Algorithm and sent to the user's mobile device using the number provided during registration. The user then enters the key they received on their device. If the key matches the one generated using the Diffie-Hellman Algorithm, data access is provided to the user, and all cloud services are provided after successful authentication.

1. CIA TRAID

To ensure the security of the system shall undertake an analysis of the security features and performance of our Proposed Scheme. We will examine various security properties, such as Data Confidentiality, Authentication, and Data Integrity.

1. **Data Confidentiality**

Analysing the Data Confidentiality feature of our proposed scheme by comparing it with other encryption algorithms like Advanced Encryption Standard or Data Encryption Standard, which use the symmetric key for data encryption. In our proposed scheme, the data is encrypted, and the cloud service provider is denied access to the data as they do not possess the key, which is only known to the data owner. This aspect ensures that the Data Confidentiality of our proposed scheme is top-notch.

1. **Authentication**

The proposed scheme employs a Two Factor Authentication protocol, which involves the user's password, set during registration, and a key generated using the Diffie-Hellman algorithm, sent to the user's mobile device. If the password and key are correct, then the user is granted access to cloud services. Therefore, Authentication in our proposed scheme is rigorous and secure.

1. **Integrity**

The encryption module of our proposed scheme ensures that data integrity is maintained, and data stored in the cloud is secure.

1. SECURE TEXT TRASFER

In today's digital age, secure transfer of personal and sensitive information is a critical requirement. Cloud computing is a popular platform for storing and sharing data due to its convenience and accessibility. However, cloud storage and transfer of information come with inherent risks that cannot be ignored. Encryption is a widely used technique to secure data while it is being transferred and stored in the cloud. The Diffie-Hellman algorithm is a cryptographic protocol that enables two parties to generate a shared secret key without any prior communication, making it a suitable algorithm for secure communication over a public network.

The proposed project aims to develop a cloud-based chatting application that uses the Diffie-Hellman algorithm for secure authentication and encryption of messages. The application provides a secure channel for communication over the cloud, ensuring the confidentiality and integrity of messages transmitted.

Once the shared secret key is generated, the text message and files can be encrypted and transmitted over the cloud to the receiver in a secure manner. Before sending the channel and key information to the receiver in the cloud, the system encrypts it using the Advanced Encryption Standard (AES). AES is a symmetric encryption algorithm that uses a shared secret key to encrypt and decrypt data. It is a widely used encryption standard that provides a high level of security for data transmission.

Proper authentication and authorization mechanisms are implemented in the proposed cloud-based chatting application to ensure only legitimate users can access the application and participate in secure communication. Key management mechanisms are also developed for secure storage and distribution of text and files.

The proposed application is designed to be user-friendly, efficient, and scalable, with robust error handling and recovery mechanisms to ensure reliability and availability. The system architecture is developed to handle a large number of users and potential network failures or disruptions. The application is compliant with relevant regulations and standards for data privacy and security, making it a reliable and trustworthy platform for secure communication over the cloud.

1. CONCLUSION

In conclusion, the use of the Diffie-Hellman key exchange algorithm in a cloud-based environment provides a secure and efficient method for text transfer. By utilizing symmetric encryption algorithms, such as AES, to encrypt data and establish shared secret keys, sensitive information can be transmitted and stored securely. Proper authentication and authorization mechanisms further ensure that only legitimate users can access the application and participate in secure communication.

Furthermore, a user-friendly interface and robust system architecture can make the application scalable and reliable, capable of handling a large number of users and potential network failures or disruptions. Adherence to relevant data privacy and security regulations and standards further enhances the security of the system.

The proposed solution provides an effective means of securing text transfer in a cloud-based environment, mitigating potential risks and protecting sensitive information.

ACKNOWLEDGMENT

I would like to express my heartfelt gratitude to everyone who contributed to the successful completion of the project "Secure text transfer using Diffie-Hellman key Exchange Based on Cloud".

Firstly, I would like to thank my project supervisor for providing me with valuable guidance and support throughout the project. Their expertise and experience have been instrumental in shaping the direction of the project and ensuring its successful completion.

Furthermore, I would like to express my appreciation to the participants who provided valuable feedback during the testing phase of the project. Their feedback helped in identifying potential issues and making necessary improvements to the system.

Lastly, I would like to express my appreciation to the educational institution for providing me with the opportunity to undertake this project. The skills and knowledge acquired during this project will undoubtedly prove to be valuable assets in my future endeavors.

REFERENCES

[1] Prevention of the Man-in-the-Middle Attack on Diffie–Hellman Key Exchange Algorithm, DOI:10.1007/978-981-15-7834-2\_58,Samrat Mitra, Samanwita Das, Malay Kule

[2] Weak keys of the Diffe Hellman key exchange I January 2005 Amit Kalele, Virendra R. Sule

[3] A Diffie-Hellman Key Exchange Protocol with Entities Authentication, Om Pal , Bashir Alam, ISSN:2319-7242 Volume 6 Issue 4 April 2017

[4] The Simplest Protocol for Oblivious Transfer Tung Chou, Claudio Orlandi, Proceedings of the 4th International Conference on Progress in Cryptology -- LATINCRYPT 2015 - Volume 9230August 2015

[5] Delay and cost performance analysis of the diffie-hellman key exchange protocol in opportunistic mobile networks March 2018 DOI:10.1088/1742-6596/978/1/012016, Bambang Soelistijanto, University of Surrey, Vinsen Muliadi

[6] Security Issues in the Diffie-Hellman Key Agreement Protocol January 2002 IEEE Transactions on Information Theory 22, Jean-francois Raymond, Anton Stiglic

[7] ECDH-ECC: A Combination of Elliptic Curve Cryptography and Diffie Hellman Based Cryptography technique for Big Data Security October 2018 DOI:10.13140/RG.2.2.27179.18722, Dr.Subhashini Radhakrishnan, Adilakshmi Vadavalli

[8] DHIES: An encryption scheme based on the Diffie-Hellman Problem January 1998, Michel Abdalla, Mihir Bellare, Phillip Rogaway

[9] Improved Diffie-Hellman Algorithm for Network Security Enhancement, November 2018, DOI:10.13140/RG.2.2.22694.78404 Vishal Garg, Rishu

[10] HMQV: A High-Performance Secure Diffie-Hellman Protocol,Advances in Cryptology – CRYPTO 2005 Conference paper, Hugo Krawczyk

[11] Implementation of Modified Diffie-Hellman Key Exchange Algorithm Using Zero Knowledge Proof, December 2020Solid State Technology 63(6):14055-14070, Mr Chakradhar, Shasidhar K, Mr B Girirajan, Mr Ch

[12] Exponentiation Speed Up For Diffie-Hellman Key Agreement Protocol January 2006, Shay Gueron

[13] Elliptic Curve Cryptography-Diffie-Hellman Technique Extended ForMultiple Two Party Keys At A Time May 2011, Vankamamidi srinivasa naresh, Nistala V.E.S. Murthy, V.V. Narasimha Rao

[14] Use of Digital Signature with Diffie Hellman Key Exchange and AES Encryption Algorithm to Enhance Data Security in Cloud Computing April 2013,DOI:10.1109/CSNT.2013.97,Prashant Rewagad, Yogita Pawar