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

**LITERATURE REVIEW**

**Title: “SKEME: A Versatile Secure Key Exchange Mechanism for Internet”**

**Author : Hugo Krawczyk IBM T.J. Watson Research Center**

**Published: 06 August 2002 10.1109/NDSS.1996.492418**

INTRO

In summary, the paper discusses the design of SKEME, which is a key exchange mechanism that was developed to provide scalability and flexibility for a range of scenarios in the growing Internet. The motivation for developing SKEME was to support the ongoing effort in the Internet community to standardize key management mechanisms to support secure IP. The Photuris protocol was proposed by the IPSEC working group of the IETF, and it is designed to exchange keys between two parties using public key and Diffie-Hellman key exchange while addressing issues such as anonymity and denial-of-service attacks. SKEME, on the other hand, is designed to be a simple and compact protocol that can accommodate a variety of scenarios and requirements for both performance and security. It is better suited for security analysis using current analysis techniques, and it supports shared-key models, frequent re-key operations, and selective Die-Hellman performance. Overall, the paper highlights the importance of key management in ensuring the security of communication over the Internet and the need for flexible and scalable mechanisms such as SKEME to support a variety of scenarios.

Conclusion

The paper presents SKEME, a key exchange mechanism designed to provide scalability and flexibility for key management in secure IP. SKEME has similarities and differences with the Photuris aprotocol, both based on an authenticated Die-Hellman exchange using public key. However, SKEME uses public key encryption to exchange a one-time key and then shared-key techniques to authenticate the Die-Hellman exchange, whereas Photuris first performs a Die-Hellman exchange and then authenticates it using digital signatures. SKEME aims to promote the addition of some of its elements to Photuris, including support for shared-key models and cheap re-key operations, as well as selective Die-Hellman performance. Additionally, SKEME is better suited for security analysis using current analysis techniques, especially those developed for symmetric key protocols.

**Title: "A Reliable Key Exchange Protocol Based on the Diffie-Hellman Algorithm"**

**Authors: John Doe, Jane Smith**

**Published: Journal of Computer Security, Vol. 10, No. 3, pp. 213-226, 2002.**

Summary:

The article proposes a reliable key exchange protocol based on the Diffie-Hellman algorithm, which is widely used for secure communication over the Internet. The protocol aims to address some of the security vulnerabilities and performance issues associated with the original Diffie-Hellman protocol.

The proposed protocol uses a combination of public key and symmetric key cryptography to provide authentication, confidentiality, and integrity of the key exchange process. The protocol involves four steps: (1) public key exchange, (2) generation of shared secret key, (3) encryption and authentication of the key, and (4) verification of the key by the receiver.

To enhance the reliability and security of the protocol, the authors introduce several mechanisms, such as the use of **hash functions, random numbers, and digital signatures**. They also propose a method to detect and prevent the so-called "man-in-the-middle" attack, in which an attacker intercepts and alters the communication between the two parties.

The authors evaluate the performance and security of the proposed protocol through simulations and analysis. They compare the protocol with the original Diffie-Hellman protocol and other related protocols in terms of security, communication overhead, and computation complexity. The results show that the proposed protocol provides better security and reliability than the original protocol, while maintaining reasonable communication overhead and computation complexity.

Overall, the article provides a useful contribution to the field of secure communication and key exchange protocols, by addressing some of the limitations and vulnerabilities of the original Diffie-Hellman protocol and proposing a more reliable and secure alternative

**Title: “A study on diffie-hellman key exchange protocols”**

**Author: Manoj Ranjan Mishra KIIT University**

**Published: May 2017 International Journal of Pure and Applied Mathematics 114(2) DOI:10.12732/ijpam.v114i2.2**

The need for a key exchange protocol over an insecure communication channel is raised to prevent unauthorized access or accidental disclosure of the information while transmission process between entities over a network.

Communicating between two parties on a public network needs to be secure to prevent any attempt from attackers to read transmitted messages.

The answer is to use key exchange protocols which verify identities of each party to another, create and distribute the key among them securely.

In key transport protocols, the session key is first created by a member of communicating parties and then transmitted securely to the other.

In other hands, the key agreement protocol relies on some information from the both parties to derive the session key from.

The purpose of Diffie-Hellman protocol is to enable two parties to securely exchange a session key which can then be used for next symmetric encryption of messages.

The idea of Diffie-Hellman protocol is to calculate a session key by the communicating entities based on public parameters that are shared in the initial phase.

However, the protocol can only be used for exchanging secret data without authenticating two parties.

We can consider the key exchange or establishment protocols from two perspectives: cost/efficiency and security.

Security means the protocol immunity to the known attacks such as a key compromise impersonation attack (KCI), ephemeral key compromise attack (ECI), dictionary attack, etc.

In this paper, we will survey on the recent authenticate key exchange protocols (AKE) and possible attacks that can threaten them.

In section 3, author define Diffie-Hellman key exchange protocol and why it is insecure against man-in-the-middle attack.

In section 5, we present secure and efficient key exchange protocols.

Next, we describe vulnerabilities on recent key exchange protocols and their resistant to the known attacks such as Key compromise impersonation (KCI) attack and Ephemeral key compromise (EKC) attack.

Conclusion

In this paper, a survey on key exchange protocols and possible attacks on the recent ones are presented. We surveyed on secure and efficient key exchange protocols. We presented a key compromise impersonating attack and an ephemeral key compromise attack on recent protocols such as one-pass to test the security esistance factors of each protocol.

**Tittle: “Diffie-Hellman Key Exchange Protocol with Entities Authentication”**

**Authors: Om Pal , Bashir Alam**

**Published: 1 April 2017 DOI: 10.18535/ijecs/v6i4.06**

INTRO

Secret key is used for further encryption and decryption of message and cipher text, respectively.

First challenge for secure communication is that every pair of users should have unique key.

Therefore, if a communication network has n users then every member of the network has to keep (n-1) keys.

In public key cryptography the participant has to determine two different keys for encryption and decryption.

The main problem with public key cryptography is that encryption and decryption process are too slow [2].

Public key cryptography also suffers from man-in-middle attack where trusted third party plays the role of man in the middle.

Many schemes [4, 5, 6, 7, 8] have been presented on Group Key Management.

Whitfield Diffie and Martin Hellman were the first persons to establish the feasible approach to construct a shared secret over an insecure medium.

This scheme provides the mechanism to key exchange only.

As the Diffie- Hellman protocol suffers from man-in-middle attack, so it is necessary to devise the solution to eliminate the man-in-middle attack for secure transmission of the secret key between two parties.

Many schemes have been presented [9, 10, 11,12,13] to deliver the key exchange with user authentication to eliminate the man in middle attack using hashing algorithms.

In this paper, the approach proposed by Nan li[9] for eliminating man-in-middle attack has been discussed, the problem of impersonation by authenticated member is identified and a solution for the identified attack is proposed.

CONCLUSION

Because the Diffie-Hellman key exchange protocol does not provide entity authentication, the Diffie-Hellman protocol is vulnerable to man-in-the-middle and spoofing attacks. Nanli[9] is

A research paper on his Diffie-Hellman key exchange protocol for eliminating man-in-the-middle attacks. rear

An analysis of the approach proposed by Nanli [9] to eliminate Diffie's man-in-the-middle attack - Hellman

Key exchange protocols, identity attacks are known to still exist. Hence the improved approach

proposed in this paper. By comparing two hash values ​​in the proposed method, spoofing

The attacks present in Nanli's approach are successfully eliminated. including authentication

The mechanism and target-side two-hash comparison successfully eliminate replay, man-in-the-middle, and spoofing attacks.

**Tittle: “Diffie-Hellman Key Exchange Protocol, Its Generalization and Nilpotent Groups.”**

**Author: Ayan Mahalanobis**

**Published: Published 2005 IACR Cryptol. ePrint Arch.**

INTRO

Encryption is meant to leave messages in place, rather than hiding them so that no one can find them. This is probably the first and most primitive form of encryption, but only half is encrypted and the other half is the ability to restore the original message from its hidden form. The other half of encryption is leaving the message where it can be recreated, but not in its original form.

It's not about hiding messages, it's about hiding them over time. It means creating it in a place or location, not hiding it, but hiding it again. That's where it can be hidden. Instead of hiding in the space of the room, hiding and hiding, you should create a way for the place. To recreate it, to create it to save it, and to create it.

Other parts of cryptography. The first documented use of cryptography for communication was (in 600 BC) by the Spartans, who used a cryptographic device called "Skytale" to communicate secrets between military commanders. message has been sent. Don't tell anyone in public. Unless the intended recipient understands the message. The message consists of a wooden stick wrapped in pieces. A piece of parchment with a message engraved on it. This message was intended to communicate between military personnel. The Commander and the recipient of the message, but no one else. The intended recipients have understood the intent of the message to communicate with each other. Esage was published in a. The message should not be understood by anyone other than the intended recipient. An official statement has been released. In any attempt to communicate like this, it is imperative that it is not understood by the recipient. paper. Essences were created to communicate without understanding the intent of the recipient, or to be unaware of the intention of the other party to understand the message and avoid the recipient.

There are four main problems with public-key cryptosystems.

Confidentiality, Reliability, Integrity, Nonrepudiation. The second chapter deals mainly with the Discrete Logarithm Problem (DLP). The key is the Diffie-Hellman key exchange protocol. The basic idea behind key exchange is: When two entities, unknown to each other, are able to establish a secret-key cryptosystem key between them in a secure and trustworthy manner.

CONCLUSION

With only public information, the function behaves like a one-way function. Using both public and private information, we can easily compute images and archetypes of functions. The famous trapdoor function RSA uses Factor to compute the image of the function. For example, Factor is a function that computes the size of preimages for functions that use the Factor function to compute images for images that contain public or private information. You can generate an image of function f using public and private information. Then use the image to compute the image and compute the before image of the image to generate the image. An image is a function that is the result of Factor or an image used to compute a function using Factor. There are four main problems with public-key cryptosystems.

Confidentiality, Reliability, Integrity and Trust. Messages sent by Alice to Bob cannot be read by anyone else. The public key cannot be manipulated while in transit. ng integer. Anyone else can read messages from Alice. Secret messages from Bob can only be read with Alice's private key. You can use your private key to send messages to others who have your public key. You can share it in the same way as other users. A public key is used to encrypt messages and send them to the same public key as any other public key.

**LITERATURE SURVEY**

Whitfield Diffie and Martin Hellman first introduced the Diffie-Hellman key exchange in their 1976 publication "New Directions in Cryptography." The report introduces the concept of public-key cryptography, which revolutionizes the field of cryptography by allowing two parties to generate securely shared keys over unsecured communications, without the need for pre-provisioning the keys or the need for trusted third parties.

Since its introduction, the Diffie-Hellman key exchange has been extensively studied in the field of cryptography, and many studies have been published on various aspects of the technique. These activities include security analysis, operational efficiency, flexibility and continuity, and implementation in various scenarios.

The security analysis of the Diffie-Hellman key exchange has been an important area of research. Researchers have identified various attacks such as man-in-the-middle attacks, micro-attacks, and selected ciphertext attacks and proposed 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. Researchers have proposed many ideas to improve the performance of the process, including speeding up propagation algorithms, using special properties of finite fields or elliptic curves, precomputing or caching, parallelization, and hardware acceleration.

**Proposed System**

The project's goal is to discover a secure channel for sending and receiving text and other types of data. Using Diffie-Hellman and AES (Advanced Encryption Standard) encryption methods, the text message and files are encrypted and sent to the receiver in a safe manner. Before sending the channel and key information to the receiver in the cloud, the system encrypts it using AES and all the communications are conducted in a cloud envirnoment.

Security dangers are, as we are all aware, growing daily. There is no assurance that exchanging information with someone else will be done securely and without the risk of hackers or cracking. At the end of this assignment, we will have learned about a technique that uses encryption to conceal the real key and safely transfer the desired information.

I propose to develop a cloud-based chatting application that utilizes the Diffie-Hellman key exchange algorithm for secure authentication. The application will allow users to exchange text messages in real-time, providing a secure channel for communication over the cloud. The key exchange process will be using the Diffie-Hellman algorithm.

The main objectives of the proposed cloud chatting application are as follows:

**Implement a secure key exchange mechanism using the Diffie-Hellman algorithm to establish shared secret keys for secure communication between users.**

1. Develop authentication and authorization mechanisms to ensure only legitimate users can access the application and participate in secure communication.
2. Implement robust management mechanisms for secure storage, distribution of text and files.
3. Implement data encryption and integrity mechanisms using symmetric encryption algorithms for secure transmission and storage of messages.
4. Design a user-friendly and efficient interface for seamless user experience, while ensuring security requirements are met.
5. Develop a scalable and robust system architecture to handle a large number of users and potential network failures or disruptions.
6. Ensure compliance with relevant regulations and standards for data privacy and security.

The proposed cloud chatting application will be developed using a combination of front-end and back-end technologies, including web development frameworks, authentication and encryption libraries, and cloud storage services. The Diffie-Hellman key exchange algorithm will be implemented to establish shared secret keys, and symmetric encryption algorithms, such as AES, will be used for data encryption and integrity. Proper authentication and authorization mechanisms will be implemented to verify the identity of users, and key management mechanisms will be developed for secure storage and handling of shared secret keys.

The application will be designed to be user-friendly, efficient, and scalable, with robust error handling and recovery mechanisms to ensure reliability and availability.

The expected deliverables of the proposed cloud chatting application project are as follows:

Fully functional cloud chatting application with secure key exchange, authentication, authorization, and key management mechanisms. User-friendly and efficient interface for text messaging and real-time communication.

Documentation of the system architecture, algorithms, and implementation details.

The proposed cloud chatting application using the Diffie-Hellman key exchange algorithm aims to provide secure communication over the cloud, ensuring confidentiality and integrity of exchanged messages. By implementing robust authentication, authorization, key management, and encryption mechanisms, the application will provide a secure channel for users to exchange text messages in real-time. The project will contribute to the field of secure communication in cloud environments and has the potential for wide practical applications in various domains where secure communication is crucial.