New Directions in Cryptography

Uygar Kaya, Tuğcan Hoşer, Onur Alaçam

Spring 2021

Project Report

Abstract: The article summarizes the increasing use of teleprocessing has necessitated the development of modern cryptographic mechanisms that minimize the need for secure key-distribution networks and providing the alternative of a written signature. These problems have been tried to be solved by using information theory and computation theory.

1. Introduction

With the development of computer controlled communication networks, the need for other communication systems has decreased, but with the development of computer controlled communication networks, some security problems arise, as examples of these problems are privacy and authentication, but modern cryptographic solutions cannot meet this problem. *"New Directions in Cryptography"* co-written by *Diffie* and *Hellman* article seeks solutions to these problems.

The most well-known cryptographic challenge is privacy, which involves preventing unauthorized data extraction from communications over an insecure network. Nevertheless, in order to use encryption to ensure secrecy, the exchanging parties must currently exchange a key that is only understood by them.

In the Privacy part there are two approach. One of the approach is *Public key cryptosystem* where instead of one secret key you have two distinct keys E and D where E is used for encrypting messages and D is used for decrypting messages. The keys are chosen such that computing the decrypting key D from E is computationally infeasible. The other approach is *Public Key Distribution System* the idea here is that once again two people who do not share a secret key that they have exchanged in advance want to arrive at a secret key.

The second problem this paper looks at is the problem of producing digital signatures we want these digital signatures to have the same properties as written signatures on paper documents which are to say that only the owner of the signature could have produced that signature and it is attached to a specific document, in other words, a digital signature proves the authenticity of a document and it can be checked by anyone to verify that authenticity.

In Section 3 (Main Section(s)), we will describe more detailed version of solutions.

The Diffie - Hellman algorithm is used in most applications. The following applications are examples of some of them;

1. *Transport Layer Security (TLS) / Secure Sockets Layer (SSL)*
2. *Internet Protocol Security (IPSec)*

Transport Layer Security (TLS) / Secure Sockets Layer (SSL)

TLS (Transport Layer Security) is a security protocol for the transport layer. TLS was derived from the Secure Service Layer (SSL) authentication protocol, and it guarantees that no third party can eavesdrops or tampers with any message. TLS/SSL can support secure data transmission by encrypting it.

Internet Protocol Security (IPSec)

IP security (IPSec) is an Internet Engineering Task Force (IETF) specification suite of protocols that provide data verification, integrity, and secrecy between two communication points over an IP network. It also specifies how packets are encrypted, decrypted, and authenticated. It defines the protocols for stable key sharing and key control.

2. Background

2.1 Definitions

Conventional Cryptography: Cryptographic system which uses same key used by sender to encrypt message and by receiver to decrypt message. It was only type of encryption in use prior to development of public-key encryption.

Public Key Cryptography: It is include the *Public Key Cryptosystem* and *Public Key Distribution Systems.*

Public Key Cryptosystem: Encryption and Decryption are controlled by two different keys, E and D such that computing D from E is computationally infeasible.

Public Key Distribution Systems: Provide an alternative method for removing the necessity for a secure key distribution route.

Plaintext: This is the original message. This message is given to the Encryption algorithm as an input. In this article, it is shown that P.

Ciphertext: The cipher text is generated by the Encryption algorithm. This message is beyond our comprehension. In this article, it is shown that C.

Message Authentication: Prevents illegal communications from being injected into a public channel, guaranteeing the message's recipient of the sender's identity.

User Authentication: Unauthorized users are prevented from accessing sensitive information via user authentication.

Computationally Secure: A system which is secure due to the computational cost of cryptanalysis, but which would succumb to an attack with unlimited computation.

Unconditionally Secure: A system which can resist any cryptanalytic attack, no matter how much computation is allowed.

Ciphertext Only Attack: A cryptanalytic attack in which the cryptanalyst possesses only ciphertext.

Known Plaintext Attack: A cryptanalytic attack in which the cryptanalyst possesses a substantial quantity of corresponding plaintext and ciphertext.

Chosen Plaintext Attack: A cryptanalytic attack in which the cryptanalyst can submit an unlimited number of plaintext messages of his own choosing and examine the resulting cryptograms.

One Time Pad: An encryption technique that cannot be cracked, but requires the use of a one-time pre-shared key the same size as, or longer than, the message being sent.

Computationally Infeasible: A computation which although computable would take far too many resources to actually compute.

Block Ciphers: An encryption method that encrypts a block of text using a deterministic algorithm and a symmetric key.

Injection: Injection is a security breaching approach that involves modifying a program unexpectedly.

Eavesdropping: The act of secretly or stealthily listening to the private conversation or communications of others without their consent in order to gather information.

Public Channel: A public channel is viewable to anybody on your team.

Trap Doors: Easy to complete in one direction, but requires a secret to execute the opposite computation effectively.

One-Way Authentication: Only one user verifies the identity of the other user.

2.2 Notations

P: *Plaintext*
C: *Ciphertext*
M: *Finite Message Space*
S_K : *EncryptionAlgorithm*
E_K : *Enciphering Key*
D_K : *Deciphering Key*

Plaintext: $\mathrm{P} = \mathrm{S}_k^{-1}(C) = S_k^{-1}(S_k(P))$

Ciphertext: $\mathrm{C} = \mathrm{S}_k\{P\}$

Enciphering Key: $\mathrm{E}_K : \{M\} \rightarrow \{M\}$

Deciphering Key: $\mathrm{D}_K : \{M\} \rightarrow \{M\}$

2.3 Detailed Problem Statement

Cryptography suffered from one fundamental problem which was that two parties that wanted to privately communicate needed to have shared a secret key in advance and this key exchange would have had to happen via some other mechanism like a courier or registered mail or meeting in person and once this key had been shared they could then send encrypted messages back and forth to each other obviously in situations where you want to do this with people you might not have yet met this is impractical you cannot wait until you can first securely exchange a key. In this *Conventional Cryptography*, Authenticity of message cannot be guaranteed, because both sender and receiver use the same key, messages cannot be verified to have come from a particular user.

In the *Conventional Cryptography* system, we use the *One-Time Pad* and unbreakable but this require extremely long keys; therefore, in most applications, the usage of *One-Time Pad* extremely expensive.

3. Main Section(s)

3.1 Conventional Cryptography

Before sending the plaintext, the sender uses the key to encrypt the message and convert it to ciphertext. Then the receiver receiving this ciphertext needs the key that the sender uses to decrypt the ciphertext.This key is transmitted by the sender over the secure channel to the receiver.Then the receiver uses this key to decrypt the ciphertext and access the plaintext. This system is called Conventional Cryptography.

Conventional Cryptography was the only encryption method used before *Public Key Cryptography*. Due to the problems mentioned in section 2.3, *Conventional Cryptography* cannot satisfy certain needs, so *Public Key Cryptography* has emerged that can satisfy these needs.

3.2 Public Key Cryptography

Public Key Cryptography is divided into two different subheadings which are *Public Key Cryptosystem* and *Public Key Distribution System*.

3.2.1 Public Key Cryptosystem

In Public Key Cryptosystem, each user has a key pair which are E_K (Enciphering Key) and D_K (Deciphering Key). E_K is the inverse of D_K. Moreover, it is computationally infeasible to find D_K at the end of specific operations from E_K. In this way, E_K can be kept publicly. In contrast, D_K must be kept secret. It is also important that public file of E_K must be protected against unauthorized modifications.

For example, if Alice wants to send a message to Bob, she encrypts the message using Bob's public key then sends the encrypted message to Bob. Bob, who then receives the message, decrypts the message using his private key and opens the message. Even if users other than Bob reaches the message, they cannot open the message because only Bob knows his private key.

3.2.2 Public Key Distribution System

X = Private Key

Y = Public Key

q = Prime Number

α = Primitive Root of q

$\alpha^X \bmod q$ = One-Way Function

$Y = \alpha^X \bmod q$

$X = \log_\alpha Y \bmod q$

$K_{ab} = {Y_a}^{X_b} \bmod q = \alpha^{X_a X_b} \bmod q$

Calculation of Y from X is easy because the mod of an exponential number can be easily computed. In contrast, calculation of X from Y is hard because of solving the discrete logarithm problem.

$3^{29} \bmod 17 = ? \rightarrow Easy$
$3^{?} \bmod 17 = 12 \rightarrow Hard$

The public key distribution system is an approach that eliminates the need for peer-to-peer authentication and secure channels. In this approach, people's private keys are randomly generated. These people create their own public keys using the one-way function of their private keys. Then, They receive each other's public key insecure. Afterwards, in order to understand that the message came from the person we are talking to, the primitive root of the function we use in the one-way function is changed with the public key of the person we are talking to, and the result is the same number on the opposite side, and authentication is provided.

There are 3 people on a network which are Alice, Bob and Eve. Alice and Bob want to communicate securely between them, so Alice and Bob create a public key using randomly assigned private keys $X_a = 15$ and $X_b = 13$ together with the common values of $\alpha = 3$ and $q = 17$, and this is the $Y_a = 3^{15} \bmod 17 = 6$ and $Y_b = 3^{13} \bmod 17 = 12$ share public keys. Using each other's public keys, $12^{15} \bmod 17 = 10$ and $6^{13} \bmod 17 = 10$, Alice and Bob can communicate securely with each other because of the same result. Although Eve knows Alice and Bob's public keys, Alice and Bob's private keys are computationally infeasible, so Eve will not be able to reach Alice and Bob's secret message.

3.3 One-Way Authentication

One of the main reasons we need cryptography is authentication. At that time, it was a crucial problem that the existing authentication systems were not purely digital, could be imitated, and did not meet the digital signature. Therefore, One-Way Authentication was created.

In daily life, written signatures provide authentication. This concept is used in digital life to create digital signatures. The digital signature must be easy for anyone to recognize as authentic, but it should be impossible for anyone except the authorized signer to produce the signature.

For instance, if Bob wants to know that Alice is the sender of the message (M), Alice first decrypts her own message with the decrypting key (D_A) and then sends $D_A(M)$ to Bob. Bob reaches the message by using Alice's public key, so it proves that the message came from Alice.

4. Discussion

Cryptography of Conventional Cryptography is straightforward. To provide security, a strong cryptography algorithm is required. The sender and receiver must keep a secure version of the private key. Public Key Cryptography is generally slower than conventional cryptography.

Since the sender and receiver use the same key, the source and authenticity of the message cannot be guaranteed, messages cannot be validate as coming from a specific user. It is not much secure compared to Public Key Cryptography. If the recipient loses the key, they cannot decrypt the message, so sending the message will fail.

The cryptanalytic difficulty of a system that can perform encryption and decryption operations in P time cannot exceed NP. Furthermore, cryptanalysis requires NP time and we also assume that the general cryptanalytic problem is NP complete.

5. Conclusion and Future Work

Conventional Cryptography was created because of the need for security. In Conventional Cryptography, users were communicating over a single secure key, but this created authentication and privacy issues. Together with the Public Key Cryptography that *Diffie-Hellman* revealed in the *"New Directions in Cryptography"* article, it solved the authentication and privacy problems by producing unique public keys and private keys.

One year after *"New Directions in Cryptography"* was published, the RSA algorithm was created by *Ron Rivest*, *Adi Shamir*, and *Leonard Adleman*. In the future, large universal quantum computers could break many popular Public Key Cryptographic systems such as RSA and Diffie-Hellman, but this cannot end encryption and privacy as we know it, and large-scale quantum computers are unlikely to be built in the next few years. Also, Symmetric encryption, or more specifically AES-256, is believed to be quantum resistant. This means that quantum computers are not expected to reduce attack time effectively if key sizes are large enough.

6. References:

• Diffie, Whitfield and Martin E. Hellman. New Directions in Cryptography. IEEE Transactions on Information Theory, 1976.

• "Applications and Limitations of Diffie-Hellman Algorithm." GeeksforGeeks, 5 Mar. 2020, www.geeksforgeeks.org/applications-and-limitations-of-diffie-hellman-algorithm.

• What is Diffie-Hellman? (2016, March 15). Unorde.Red. https://unorde.red/what-is-diffie-hellman/

• Public-Key Cryptography. (1995). https://ee.stanford.edu/ hellman/pkc.pdf

• Analysis of Diffie Hellman Key Exchange Algorithm with Proposed Key Exchange Algorithm. (2015). https://www.ijettcs.org/Volume4Issue1/IJETTCS-2015-01-10-24.pdf

• A Comparative Analysis of Encryption Algorithms for Better Utilization. (2013). https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.402.7978rep=rep1type=pdf

• "Transport Layer Security (TLS)." GeeksforGeeks, 26 Feb. 2021, www.geeksforgeeks.org/transport-layer-security-tls.

• "IP Security (IPSec)." GeeksforGeeks, 4 Feb. 2020, www.geeksforgeeks.org/ip-security-ipsec.