

Cryptography on the Internet

**Computer Communications and Networking
ENG SC546**

**Jody Chai
Melanie Leung
Michael Ducott
Wingsze Yuen**

May 4, 2001

CONTENTS

1. Introduction
2. Security Overview
3. Types of Cryptographic Algorithms
 - 3.1. Secret-Key Cryptography
 - 3.2. Public-Key Cryptography
 - 3.3. Hash Functions
4. Cryptographic Protocols
 - 4.1. PGP - Pretty Good Privacy
 - 4.1.1. Security of IDEA
 - 4.1.2. Security of RSA
 - 4.1. SSL - Secure Sockets Layer
5. Conclusion
6. Bibliography

1. Introduction

In today's world, the Internet provides essential communication between tens of millions of people and is constantly growing as a tool for commerce. However, now that the Internet is so essential and open to the public, security is an important issue to keep in mind. All communication over the Internet uses the Transmission Control Protocol/Internet Protocol (TCP/IP). TCP/IP allows information to be sent from one computer to another through a sequence of intermediate computers and separate networks. The fact that TCP/IP allows information to pass through intermediate computers makes it possible for a third party to interfere with communications by eavesdropping, tampering information, or even impersonating someone else while sending information.

In order to protect against such violations, there are several solutions. The main purpose of this paper is to define some of the terms and concepts behind today's cryptographic methods and algorithms, as well as to define some of the latest protocols in use today.

2. Security Overview

Since TCP/IP allows information to pass through intermediate computers, a third party could interfere with communications. Some of the most common solutions to this problem include setting up a firewall, tamper detection, authentication, privacy/confidentiality, integrity, non-repudiation, and cryptography. A firewall shields an internet from the Internet. A firewall is built from a software and hardware combination using routers to filter packets and a proxy server. A firewall protects networked computers from intentional hostile intrusion that could compromise confidentiality or result in data corruption or denial of service. It may be a hardware device or a software program running on a secure host computer. In either case, it must have at least two network interfaces, one for the network it is intended to protect, and one for the network it is exposed to. A firewall sits at the junction point or gateway between the two networks, usually a private network and a public network such as the Internet.

A firewall examines all traffic routed between the two networks to see if it meets certain criteria. If it does, it is routed between the networks, otherwise it is stopped. A firewall filters both inbound and outbound traffic. It can also manage public access to private networked resources such as host applications. It can be used to log all attempts to enter the private network and trigger alarms when hostile or unauthorized entry is attempted. Firewalls can filter packets based on their source and destination addresses and port numbers. This is known as address filtering. Firewalls can also filter specific types of network traffic. This is also known as protocol filtering because the decision to forward or reject traffic is dependant upon the protocol used. Firewalls can also filter traffic by packet attribute or state.

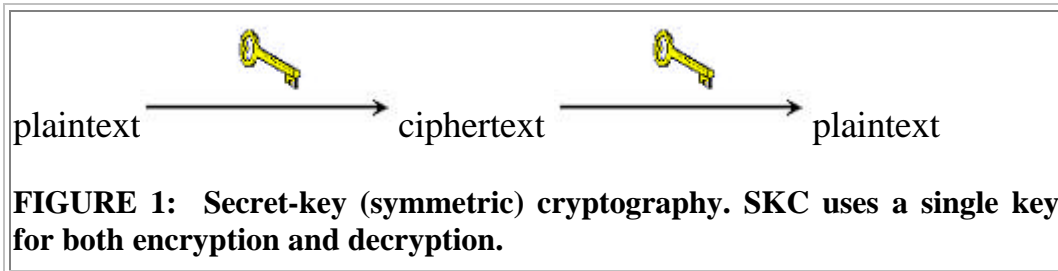
Tamper detection allows the information receiver to verify that it has not been modified during transmission. If there were any attempt to modify or substitute data, a false message would be detected. Authentication allows the information receiver to determine who sent the message. Privacy/confidentiality ensures that no one can read the message except the intended receiver. Integrity assures the receiver that the message that they received was not modified in any way since it was sent from the origin. Non-repudiation is a mechanism that proves that the sender really sent the message. Lastly, cryptography allows two communication parties to disguise information they send to each other. The sender encrypts the information before sending it. The receiver decrypts the information after receiving it.

In cryptographic terminology, the message is called plaintext or cleartext. Encryption is encoding the contents of the message in such a way that hides its contents from outsiders. The encrypted message is called the ciphertext. The process of retrieving the plaintext from the ciphertext is called decryption. Encryption and decryption usually make use of a key, and the coding method is such that decryption can be performed only by knowing the proper key. Therefore, cryptography not only protects data from theft or alteration, but can also be used for user authentication. In cryptography, the size of the key is directly proportional to the level of security. The larger the key size, the harder it is to crack a block of encrypted data because there are more possible combinations. For the most part, there are three types of cryptographic algorithms typically used to accomplish data protection: secret key (also known as symmetric) cryptography, public-key (also known as asymmetric) cryptography, and hash functions, each of these will be described in the next section.

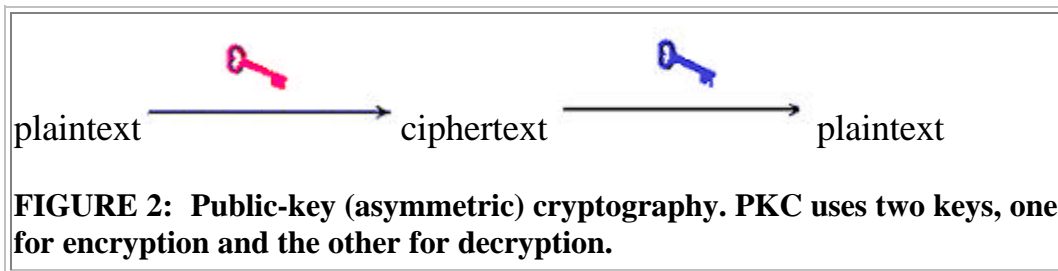
3. Types of Cryptographic Algorithms

The three types of algorithms that will be discussed are:

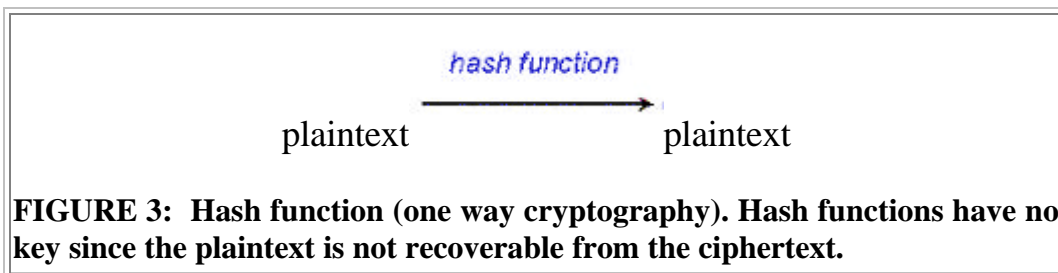
Secret-Key Cryptography: Uses a single key for both encryption and decryption



Public-Key Cryptography: Uses one key for encryption and another for decryption



Hash Functions: Uses a mathematical transformation to irreversibly "encrypt" information



3.1 Secret-Key Cryptography

In secret key cryptography, a single key is used for both encryption and decryption. As shown in Figure 1, the sender uses the key to encrypt the plaintext and then sends the ciphertext to the receiver. The receiver applies the same key to decrypt the message and recover the

plaintext. Because a single key is used for both functions, secret key cryptography is also called symmetric encryption.

There are several widely used secret key cryptography schemes and they are generally categorized as being either block ciphers or stream ciphers. A block cipher is so-called because it encrypts blocks of data at a time. The same plaintext block will always be encrypted into the same ciphertext when using the same key. Stream ciphers operate on a single bit, byte, or word at a time, and implements a feedback mechanism so that the same plaintext will yield different ciphertext everytime it is encrypted. Some of the major methods are listed in the table below:

Algorithm	Description
ROT13	Keyless text scrambler, very weak
DES	56-bit block cipher; patented, but freely usable
RC2 to RC5	Variable key length block cipher; proprietary
IDEA	128-bit block cipher; patented
Skipjack	80-bit stream cipher; classified (owned by the US Government-CIA)

The most common secret-key cryptography scheme is Data Encryption Standard (DES). DES (actually the Data Encryption Algorithm, or DEA) is a block-cipher employing a 6-bit key that operates on 64-bit blocks.

3.2 Public-Key Cryptography

Public-key cryptography (PKC) was first described publicly by Stanford University professor Martin Hellman and graduate student Whitfield Diffie in 1976. PKC employs two keys that are mathematically related. One key is used to encrypt the plaintext and the other key is used to decrypt the ciphertext. Both keys are required for the process to work (Figure 2). Because a pair of keys is required, this approach is also called asymmetric cryptography.

In PKC, one of the keys is designated the public key. The other key is designated the private key and is never revealed to another party. For example, the sender encrypts some information using the intended receiver's public key. The receiver decrypts the ciphertext using their own private key. This method could be also used in both directions at the same time. The sender, for example, could encrypt the plaintext first with their own private key and then encrypt

again with the receiver's public key. This scheme might be used where it is important that the sender cannot deny sending the message (non-repudiation).

Some of the major methods are listed in the table below:

Algorithm	Description
Diffie-Hellman	Key exchange protocol; patented
RSA	Public-key encryption and digital signature; patented
ElGamal	Public key encryption and digital signature; patented
DSA	Digital Signatures only; patented

The first, and still most common, PKC implementation is RSA, named for the three MIT mathematicians who developed it-Ronald Rivest, Adi Shamir, and Leonard Adleman. RSA is used for key exchange or encryption of small blocks of data. RSA uses a variable size encryption block and a variable size key. The key-pair is derived from a very large number, n , that is the product of yielding an n with roughly twice as many digits as the prime factors. The public key information includes n and a derivative of one of the factors of n .

An alternative to RSA was published by the National Institute for Standards and Technology (NIST) in 1991. The Digital Signature Algorithm (DSA) is part of NIST's proposed Digital Signature Standard (DSS), both part of the U.S. government's desire to define a next-generation cryptography system.

Diffie-Hellman is a system for exchanging cryptographic keys between active parties. Diffie-Hellman is not actually a method of encryption and decryption, but a method of developing and exchanging shared private keys over a public communication channel. ElGamal may be used for encryption of digital signatures similarly to the RSA algorithm.

3.3 Hash Functions

Hash functions, also called message digests and one-way encryption, are algorithms that, in some sense, use no key (Figure 3). Instead, they transform the plaintext mathematically so that the contents and length of the plaintext are not recoverable from the ciphertext.

Hash algorithms are typically used to provide a digital fingerprint of a file's contents, often used to ensure that the file has not been altered by an intruder or virus. Hash functions are

well suited for ensuring data integrity because any change made to the contents of a message will result in the receiver calculating a different hash value than the one placed in the transmission by the sender. Hash functions are also commonly employed by many operating systems to encrypt passwords.

Among the common hash functions in use today in commercial cryptographic applications are a family of Message Digest (MD) algorithms, all of which are byte-oriented schemes that produce a 128-bit hash value from an arbitrary-length message.

The Secure Hash Algorithm (SHA), proposed by NIST for their Secure Hash Standard (SHS), is seeing increased use in commercial products today. SHA produces a 160-bit hash value.

4. Cryptographic Protocols

4.1 Pretty Good Privacy

Pretty Good Privacy (PGP) is a public key system for encrypting electronic mail using the RSA public key cipher. It encrypts the message using the International Data Encryption Algorithm (IDEA) cipher with a randomly generated key. It then encrypts the key using the recipients' public key. When the recipient receives the message, PGP uses the private RSA, key to decrypt the IDEA key and then uses that IDEA key to decrypt the message.

PGP is a hybrid cryptosystem, it combines features of conventional and public key cryptography. PGP creates a session key, which is a one-time-only secret key. This key is a random number generated from the random movements of the mouse and the keystrokes you type. This session key works with a very secure, fast conventional encryption algorithm to encrypt the plaintext; the result is ciphertext. Once the data is encrypted, the session key is then encrypted to the recipient's public key. This public key-encrypted session key is transmitted along with the ciphertext to the recipient.

PGP can also be used to sign messages. It first computes a "hash" of the message using the hash function. It then encrypts this hash output (128 bits or 16 bytes) with the secret RSA key of the sender. Any recipient can calculate the same hash output of the received message, use the senders public key to decrypt the signature. If the output to this decryption agrees with the

recipients calculated hash output, then the recipient knows both that the sender actually sent that message, and that not a single bit of that message has been changed.

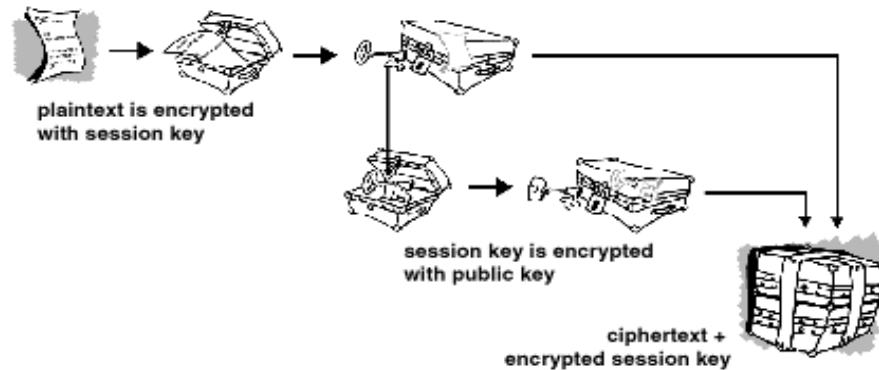


Fig. 4: Shows how a message is decrypted

4.1.1 Security of IDEA

IDEA, which was finalized in 1992, is a block cipher that operates on 64-bit blocks of data. The only method of attack is brute force. The keyspace of IDEA is 128-bits. That is about 340×10^{36} . To actually break this key, about half the keyspace (170×10^{36}) must be searched. This is nearly impossible to do.

4.1.2 Security of RSA

RSA is the first full fledged public key cryptosystem that gets its security from the fact that factoring very large composites.

The following is how RSA works:

- Find 2 very large primes, p and q .
- Find $n=pq$ (the public modulus).
- Choose e , such that $e < n$ and relatively prime to $(p-1)(q-1)$.
- Compute $d=e^{-1} \bmod [(p-1)(q-1)]$ OR $ed=1 \bmod [(p-1)(q-1)]$.
- e is the public exponent and d is the private one.
- The public-key is (n,e) , and the private key is (n,d) .

- p and q should never be revealed, preferably destroyed (PGP keeps p and q to speed operations by use of the Chinese Remainder Theorem, but they are kept encrypted)

Encryption is done by dividing the target message into blocks smaller than n and doing modular exponentiation:

$$c=m^e \text{ mod } n$$

Decryption is simply the inverse operation:

$$m=c^d \text{ mod } n$$

When attempting to attack the RSA, the attacker has access to the public key (e and n). What the attacker wants is the private key (d). To get d, n needs to be factored, which will yield p and q, which can then be used to calculate d. Factoring n is the best known attack against RSA to date.

Algorithm	Description
Trail Division	The oldest and least efficient, it has an exponential running time.
Quadratic Sieve (QS)	The fastest algorithm for numbers smaller than 110 digits.
Multiple Polynomial Quadratic Sieve (MPQS)	Faster version of QS
Double Large Prime Variation of the MPQS	Faster version of MPQS
Number Field Sieve (NFS)	Fastest algorithm known for numbers larger than 110 digits.

The following table estimates the effort required to factor the RSA public-key modulus lengths using the General Number Field Sieve.

Key Size	MIPS-years required to factor
512	30,000
768	200,000,000
1024	300,000,000,000
2048	300,000,000,000,000,000,000,000,000

As it can be seen by this table, attempting to find the correct key is nearly impossible given the amount of time that it would actually take to do so.

4.2 SSL Protocol

SSL provides one of the most commonly available security mechanisms on the Internet. SSL stands for Secure Sockets Layer, though IETF is renaming it TLS (Transport Layer Security). Developed by Netscape, SSL is used extensively by web browsers to provide secure connections for transferring credit cards numbers and other sensitive data. An SSL-protected HTTP transfer uses port 443.

SSL is based on cryptography. The data is encoded so that the only person that can decode it is the intended recipient, and not a third party who might be able to intercept the information in transit. The simplest way to do this is for the sender and receiver to use a secret key, which can be used along with an agreed-upon algorithm to scramble the data in such a way that only someone with the key can unscramble it. The secret key acts like a password. One of the most well-known secret key systems is the Data Encryption Standard (DES), developed by the U.S. National Security Agency. SSL uses a secret key system called RC4, developed by RSA, to encrypt its transfers. The use of a secret key implies that the participants in the conversation must have selected a key and communicated it among themselves in a secure manner. However, when establishing an Internet connection, there is usually no pre-arranged key, so a means must be provided of securely generating one.

A public key system, usually based on mathematical principles of modulo arithmetic, uses two keys. Information encrypted with one of the keys can only be decrypted with the other key, and vice versa. You cannot encrypt and then decrypt a message with only one key. Usually, the public key is published and then the private key. Now anyone can encrypt a message using the public key and transmit it across an insecure network, knowing that only the holder of the private key can decrypt it. Not only can you encrypt with the public key, but you can also encrypt with the private key. Anyone can decrypt such a message, but only the private key holder could have generated it in the first place. This gives us a means of digitally signing messages in a way that no one else could duplicate.

5. Conclusion

From e-mail to cellular communications, from secure web access to digital cash, cryptography is an essential part of today's information systems. Cryptography helps provide accountability, fairness, accurate, and confidentiality. It can help to prevent fraud in electronic commerce and assure the validity of financial transactions. It can prove ones identity or protect anonymity. However, no one can guarantee 100% security, only added protection.

Bibliography

- "AS/400 Annoucement at a Glance." <http://as400service.ibm.com/as400/as400v4r1/97nc/97nc46.htm>
- "Chapter 12: Encryption on the Internet." <http://ac.ceu.edu/cois/bb/Bcis1300/Chap12&13/sld001.htm>
- "Common cryptographic systems." <http://antares.cpe.fr/~tm/crypto/Crypto3.html>
- "Fundamentals of Cryptography." <http://www.infosyssec.org/infosyssec/cryptintro.htm>
- Garfinkel, Simson. PGP: Pretty Good Privacy. O'Reilly & Associates. New York, NY. 1994
- "Internet Security." <http://www.cbtsys.com/catalog/curricula/secure.htm>
- "Introduction to Public-key Cryptography." <http://developer.netscape.com/docs/manuals/security/pkin/index>
- "Security & Privacy: Safeguarding the Internet." <http://www.ils.unc.edu/yangk/inls181/lectures/security/sld001.htm>