**15 chapter syllabus**

CIA Triad: \*\*\*(confidentiality, integrity, availability)

<http://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA>

**Authentication**: In security systems, authentication is distinct from authorization , which is the process of giving individuals access to system objects based on their identity.

**Cryptography**: is the practice and study of techniques for secure communication in the presence of third parties called adversaries.

**Q:** What is cryptography used for?

**cryptography** is **used** to provide secrecy and **integrity** to our data, and both authentication and anonymity to our communications.

ALSO: <http://searchsecurity.techtarget.com/definition/encryption>

Encryption and Decryption:

**Encryption** is the process of translating plain text data (plaintext) into something that appears to be random and meaningless (ciphertext). **Decryption** is the process of converting ciphertext back to plaintext.

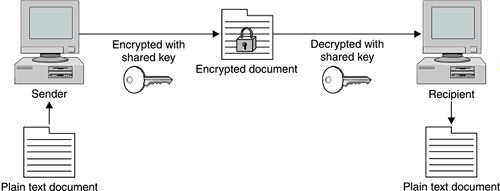
Types of encryption and decryption:

### Types of Encryption

There are two main ways to do encryption today. The first kind of encryption, called **symmetric cryptography** or **shared secret encryption**, has been used since ancient Egyptian times. This form of encryption uses a secret key, called the **shared secret**, to scramble the data into unintelligible gibberish. The person on the other end needs the shared secret (key) to unlock the data—the encryption algorithm. You can change the key and change the results of the encryption. It is called symmetric cryptography because the same key is used on both ends for both encryption and decryption (see [Figure 9.2](http://books.gigatux.nl/mirror/securitytools/ddu/ch09lev1sec1.html#ch09fig02)).

##### Figure 9.2. Symmetric Cryptography

[[View full size image]](http://books.gigatux.nl/mirror/securitytools/ddu/images/0321194438/graphics/09fig02_alt.jpg)



The problem with this method is that you have to communicate the secret key securely to your intended recipient. If your enemy intercepts the key, he can read the message. All kinds of systems were invented to try to get around this basic weakness, but the fact remained: you still had to communicate the secret key in some way to your intended recipient before you could commence secure communications.

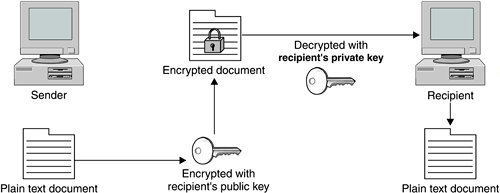
A revolution in encryption was started when Whitfield Diffie, Martin Hellman, and Ralph Merkle invented Public Key cryptography. (Actually, there is some debate whether the British civil servant James Ellis really invented it earlier and kept it secret, but Diffie, Hellman, and Merkle were the first to go public with it in 1976.) They were trying to solve the age-old problem of key exchange. Diffie wondered how two individuals wanting to make a financial transaction over an electronic network could do so securely. He was thinking far ahead here, because the Internet was in its infancy at the time and e-commerce didn't yet exist. If big governments had problems dealing with the key exchange problem, how could the average person manage this? He wanted to come up with a system by which two parties could easily hold protected conversations and secure transactions without having to exchange keys every time. He knew that if he could solve the key exchange problem, it would be a huge advance in cryptography.

Diffie partnered with Martin Hellman and Ralph Merkle. It took them a few years, but finally they came up with a system called **public key encryption** (PKE), also known as **asymmetric cryptography**.

Asymmetric cryptography uses encryption that splits the key into two smaller keys. One of the keys is made public and one is kept private. You encrypt a message with the recipient's public key. The recipient can then decrypt it with their private key. And they can do the same for you, encrypting a message with your public key so you can decrypt it with your private key (see [Figure 9.3](http://books.gigatux.nl/mirror/securitytools/ddu/ch09lev1sec1.html#ch09fig03)). The difference here is that you don't need someone's private key to send him or her a secure message. You use his or her public key, which doesn't have to be kept secure (in fact, it can be published like a phone number). By using your recipient's public key, you know that only that person can encrypt it using his or her private key. This system allows two entities to communicate securely without any prior exchange of keys.

##### Figure 9.3. Asymmetric Cryptography (Public Key)

[[View full size image]](http://books.gigatux.nl/mirror/securitytools/ddu/images/0321194438/graphics/09fig03_alt.jpg)



Asymmetric cryptography is usually implemented by the use of one-way functions. In mathematic terms, these are functions that are easy to compute in one direction but very difficult to compute in reverse. This is what allows you to publish your public key, which is derived from your private key. It is very difficult to work backwards and determine the private key. A common one-way function used today is factoring large prime numbers. It is easy to multiply two prime numbers together and get a product. However, to determine which of the many possibilities are the two factors of the product is one of the great mathematical problems. If anyone were to invent a method for easily deducing factors of large prime numbers, it could make obsolete much of the public key encryption used today. Fortunately, other one-way functions work for this application, such as calculations on elliptical curves or computation of inverse logarithms over a finite field.

Soon after the paper by Diffie, Hellman, and Merkle was released, another group of three men developed a practical application of the theory. Their system for public key encryption was called RSA after their names: Ronald Rivest, Adi Shamir, and Leonard Adleman. They formed a company and began licensing their system. The adoption rate was slow and their company almost went out of business, until they cut a deal to take advantage of the growing Internet commerce field with a then little-known company called Netscape. The rest is history, and RSA is now the most widely used public key encryption algorithm. Diffie and Hellman eventually released a practical application of their own, but it is usable only for key exchanges, whereas RSA can do authentication and nonrepudiation.

Public key encryption is now behind every Web server that offers you a secure purchase. Your transaction is encrypted without giving or taking a secret key, and it all happens in the background. All we know as users is that the little SSL lock symbol displays in our browser and we feel safer. Imagine the effects on Internet commerce if every time you wanted to buy something online you had to think of a secret key, encrypt the message, and then somehow communicate that key to the other party. Obviously, e-commerce could not exist as it does today without public key cryptography.

There are many different encryption algorithms, protocols, and applications based on these two main types of encryption. The following sections introduce some of these.

#### Encryption Algorithms

Today, strength of encryption is usually measured by key size. No matter how strong the algorithm, the encrypted data can be subject to brute force attacks in which all possible combinations of keys are tried. Eventually the encryption can be cracked. For most modern ciphers with decent key lengths, the time to crack them with brute force them is measured in millennia. However, an undisclosed flaw in an algorithm or an advance in computer technology or mathematical methods could sharply decrease these times.

Generally, the thinking is that the key length should be suitable for keeping the data secure for a reasonable amount of time. If the item is very topical, such as battlefield communications or daily stock information, then a cipher that protects it for a matter of weeks or months is just fine. However, something like your credit card number or national security secrets need to be kept secure for a longer period, effectively forever. So using weaker encryption algorithms or shorter key lengths for some things is okay, as long as the information usefulness to an outsider expires in a short amount of time.

##### Data Encryption Standard (DES)

DES is the original standard that the U.S. government began promoting for both government and business use. Originally thought to be practically unbreakable in the 1970s, the increase in power and decrease in cost of computing has made its 56-bit key functionally obsolete for highly sensitive information. However, it is still used in many commercial products and is considered acceptable for lower security applications. It also is used in products that have slower processors, such as smart cards and appliance devices that can't process a larger key size.

##### TripleDES

TripleDES, or 3DES as it is sometimes written, is the newer, improved version of DES, and its name implies what it does. It runs DES three times on the data in three phases: encrypt, decrypt, and then encrypt again. It actually doesn't give a threefold increase in the strength of the cipher (because the first encryption key is used twice to encrypt the data and then a second key is used to encrypt the results of that process), but it still gives an effective key length of 168 bits, which is plenty strong for almost all uses.

##### RC4, RC5, and RC6

This is an encryption algorithm developed by Ronald Rivest, one of the developers of RSA, the first commercial application of public key cryptography. Improvements have been made over time to make it stronger and fix minor issues. The current version, RC6, allows up to a 2,040-bit key size and variable block size up to 128 bits.

##### AES

When the U.S. government realized that DES would eventually reach the end of its useful life, it began a search for a replacement. The National Institute of Standards and Technology (NIST), a government standards body, announced an open competition for a new algorithm that would become the new government standard. There were many competitors including RC6, Blowfish by renowned cryptographer Bruce Schneier, and other worthy algorithms. They settled on AES, which is based on an algorithm called Rijndael, designed by two Belgian cryptographers. This is significant because they used an open competition to decide on the standard. Also, selecting an algorithm by two non-American developers with no significant commercial interests helped to legitimize this selection worldwide. AES is rapidly becoming the new standard for encryption. It offers up to a 256-bit cipher key, which is more than enough power for the foreseeable future. Typically, AES is implemented in either 128- or 192-bit mode for performance considerations.

#### Encryption Applications

##### Hashes

Hashes are a special use of one-way functions to provide authentication and verification using encryption. A hash function takes a file and puts it through a function so that it produces a much smaller file of a set size. By hashing a file, you produce a unique fingerprint of it. This gives you a way to make sure that the file has not been altered in any way. By hashing a suspect file and comparing the hash to the known good hash, you can tell if any changes have been made. It is unlikely that a file with a different structure would produce an identical hash. Even changing one character changes the hash significantly. The chances of two different files producing the same hash are infinitesimal.

Hashes are often provided on downloaded versions of software to make sure you are getting the real thing. This is important, especially with open source software, where it may have been passed around quite a bit or downloaded from another site. The official Web site will usually post the correct hash of the latest version. If the two don't match, then you know some changes have been made, possibly without the permission or knowledge of the software developers. The most popular hashing algorithm is called MD5.

##### Digital Certificates/Digital signature

Digital certificates are the "signature" of the Internet commerce world. These use a combination of encryption types to provide authentication. They prove that who you are connecting to is really who they say they are. Simply put, a certificate is a "certification" of where the information is coming from. A certificate contains the public key of the organization encrypted with either its private key or the private key of a signing authority. Using a signing or certificate authority is considered the more secure method of the two. If you can decrypt the certificate with their public key, then you can reasonably assume the Web site belongs to that organization.

Certificates are usually tied to a particular domain. They can be issued by a central entity, called a Certificate Authority (CA), or created and signed locally as described above. There are several of these organizations, the biggest of which is VeriSign, the company that also runs the domain names system. They have sanctioned many other companies to offer certificates under their authority. Getting a certificate from VeriSign or one of the companies it authorizes is like having someone vouch for you. Generally, they will not issue you a certificate until they verify the information you are putting in the certificate, either by phone or via some kind of paper documentation, such as a corporate charter. Once they "certify" you, they will take this information, including the URLs you are going to use the certificate for, and digitally "sign" it by encrypting it with their private key. Then a Web server or other program can use this certificate. When outside users receive some data, such as a Web page from the server, and it has a certificate attached, they can use public key cryptography to decrypt the certificate and verify your identity. Certificates are used most often at e-commerce Web sites, but they can also be used for any form of communications. SSH and Nessus both can use certificates for authentication. VPNs also can use certificates for authentication instead of passwords.

**What is a block cipher?**

A block cipher is an encryption algorithm that encrypts a fixed size of n-bits of data - known as a block - at one time. The usual sizes of each block are 64 bits, 128 bits, and 256 bits. So for example, a 64-bit block cipher will take in 64 bits of plaintext and encrypt it into 64 bits of ciphertext. In cases where bits of plaintext is shorter than the block size, padding schemes are called into play. Majority of the symmetric ciphers used today are actually block ciphers. DES, Triple DES, AES, IDEA, and Blowfish are some of the commonly used encryption algorithms that fall under this group.

## **What is a stream cipher?**

A stream cipher is an encryption algorithm that encrypts 1 bit or byte of plaintext at a time. It uses an infinite stream of pseudorandom bits as the key.

Example: RC4 algorithm

From book:

To protect a system, we must take security measures at four levels:

**1. Physical**. The site or sites containing the computer systems must be

physically secured against armed or surreptitious entry by intruders.

Both the machine rooms and the terminals or workstations that have

access to the machines must be secured.

**2. Human**. Authorization must be done carefully to assure that only

appropriate users have access to the system. Even authorized users,

however,may be “encouraged” to let others use their access (in exchange

for a bribe, for example). They may also be tricked into allowing

access via **social engineering**. One type of social-engineering attack

is **phishing**. Here, a legitimate-looking e-mail or web page misleads

a user into entering confidential information. Another technique is

**dumpster diving**, a general term for attempting to gather information in

order to gain unauthorized access to the computer (by looking through

trash, finding phone books, or finding notes containing passwords, for

example). These security problems are management and personnel issues,

not problems pertaining to operating systems.

**3. Operating system**. The system must protect itself from accidental or

purposeful security breaches. A runaway process could constitute an

accidental denial-of-service attack.Aquery to a service could reveal passwords.

A stack overflow could allow the launching of an unauthorized

process. The list of possible breaches is almost endless.

**4. Network**. Much computer data in modern systems travels over private

leased lines, shared lines like the Internet, wireless connections, or dial-up

lines. Intercepting these data could be just as harmful as breaking into a

computer, and interruption of communications could constitute a remote

denial-of-service attack, diminishing users’ use of and trust in the system.

Security at the first two levels must be maintained if operating-system

security is to be ensured. A weakness at a high level of security (physical or

human) allows circumvention of strict low-level (operating-system) security

measures. Thus, the old adage that a chain is only as strong as its weakest link

is especially true of system security. All of these aspects must be addressed for

security to be maintained.

**15.2.4 Stack and Buffer Overflow \*\*\* page 663**