Topic 6.3: Data Encryption

Encryption is the translation of data into a secret code that cannot be read without the use of a password or secret key. Unencrypted data is called *plain text,* whereas encrypted data is called *cipher text*. Here is a short sample of cipher text:

```
FdgvdIC7sDv7G1Z7pCNzFLp0lgB9ACm8r5RZOBi
```

Why is Encryption Needed?

Encryption is intended to prevent unauthorized individuals from accessing sensitive data. While it is technically possible to decrypt cipher text without the required key, good encryption methods with strong keys make any "brute force" attempt to break the encryption key impossible from a practical perspective because it would take many years of high-speed processing to find the right key by sequentially guessing possibilities.

Here are some of the reasons organizations encrypt data:

- **Legal Protection**: There are many federal, state and local laws that require protection of sensitive data. Failure to take reasonable precautions to protect the data from unauthorized disclosure (encryption being one of them) can expose an organization to criminal and civil penalties, including incarceration, staggering fines, and providing identity theft monitoring to potential victims. Add in the lost business due to a damaged reputation, and the requirement becomes even more compelling. Here are a few of the pertinent privacy laws:
    - **The Privacy Act of 1974 (and its Amendments)** prohibits government agencies from disclosing personally identifiable information without the subject individual's consent, with twelve specific exceptions.
    - **The Freedom of Information Act** requires disclosure of official government records given an authorized request, with nine specific exceptions (personal and medical data falls under one of the exceptions).
    - **The Health Insurance Portability and Accountability Act (HIPAA)** requires the protection of health information and related personal identifiable information, including three required types of security safeguards.
    - **The Gramm-Leach-Bliley Act** requires financial institutions to explain their information disclosure practices to customers and to safeguard sensitive data.
    - **The Family Educational Rights and Privacy Act (FERPA)** requires protection of academic records.

    There are many more. See SUMMARY OF SELECTED FEDERAL LAWS AND REGULATIONS ADDRESSING CONFIDENTIALITY, PRIVACY AND SECURITY

- **Industry Standards and Regulations:** Many industries have established standards for handling sensitive information. Here are a few:
    - **The PCI Standards Council** provides payment card industry compliance standards and guidelines for the protection of account information, including the PCI DSS (Data Security Standard).

    - **The American Bankers Association** provides compliance solutions, tools, training, and other resources for the banking industry.
    - **The International Organization for Standardization (ISO)** provides the ISO 27000 family of standards to help organizations keep information assets secure.
- **Maintenance of an Organization's Reputation:** Security breaches undermine the public's trust in an organization. This can result in loss of market share, or in extreme cases, the inability to continue operating.

Encryption Algorithms

An encryption algorithm is a program that converts plain text into cipher text. The first step is to convert all the characters in the plain text message into numbers. The algorithm then uses a sequence of mathematical operations to convert the plain text into cipher text, essentially scrambling the data so that it is unreadable until a reverse of the encryption steps are performed to change the cipher text back into plain text. In order to make the encryption unique, at least one key is required. The key is simply a very large number that is used as part of the calculations performed by the algorithm. Most modern encryption keys are computed using two very large prime numbers, which makes keys more difficult to break because a prime number can be evenly divided by only two numbers – itself and the number 1.

Some of the prominent encryption algorithms are:

DES (Data Encryption Standard): The original DES standard was developed in the 1970s using 56 bit keys (plus 8 parity bits for a total key size of 64 bits). It was published as Federal Information Processing (FIPS) standard in 1977. The key size turned out to be too small and the algorithm was easily breakable, as publicly proven in 1999. DES was followed by Double DES, which doubled the size of the key, and later by Triple DES, which tripled the size of the key. Although use has fallen over time, Triple DES is still in use today.

RSA, named using the first letters of the last names of the three people who developed it (Ron Rivest, Adi Shamir, and Leonard Adleman), was first publicly described in 1977. It is based a factoring technique using two large prime numbers along with an auxiliary value. RSA has become the de facto standard for encrypting data transmitted over the internet. A publicly available encryption application called Pretty Good Privacy (PGP) was written in 1991 by Phil Zimmermann. PGP makes use of the RSA algorithm. Modern RSA implementations use keys as large as 1024 bits.

Blowfish is an algorithm intended to replace DES. The cipher splits messages into blocks of 64 bits and encrypts each block individually. Blowfish is known for its speed and effectiveness. In fact, many claim it has never been defeated. Blowfish is common in e-commerce and in password management tools, where it is used to protect passwords.

Twofish is a successor to Blowfish, developed by the same security expert, Bruce Schneier. It uses a 256 bit key. You will find Twofish bundled in encryption software such as PhotoEncrypt, GPG, and TrueCrypt.

AES (Advanced Encryption Standard) is the algorithm currently used as the standard by the U.S. Government and numerous other organizations. It runs very efficiently using a 128-bit key, but can uses keys of 192 and 256 bits for higher security.

Looking into the future, two evolving techniques show promise:

Honey Encryption deters hackers by returning readable fake data for every incorrect guess of the key code, in the form of an incorrect plain text password or encryption key. This not only slows hackers down by burying them with seemingly useful data, but it also makes it difficult to recognize correct data should they guess the correct key.

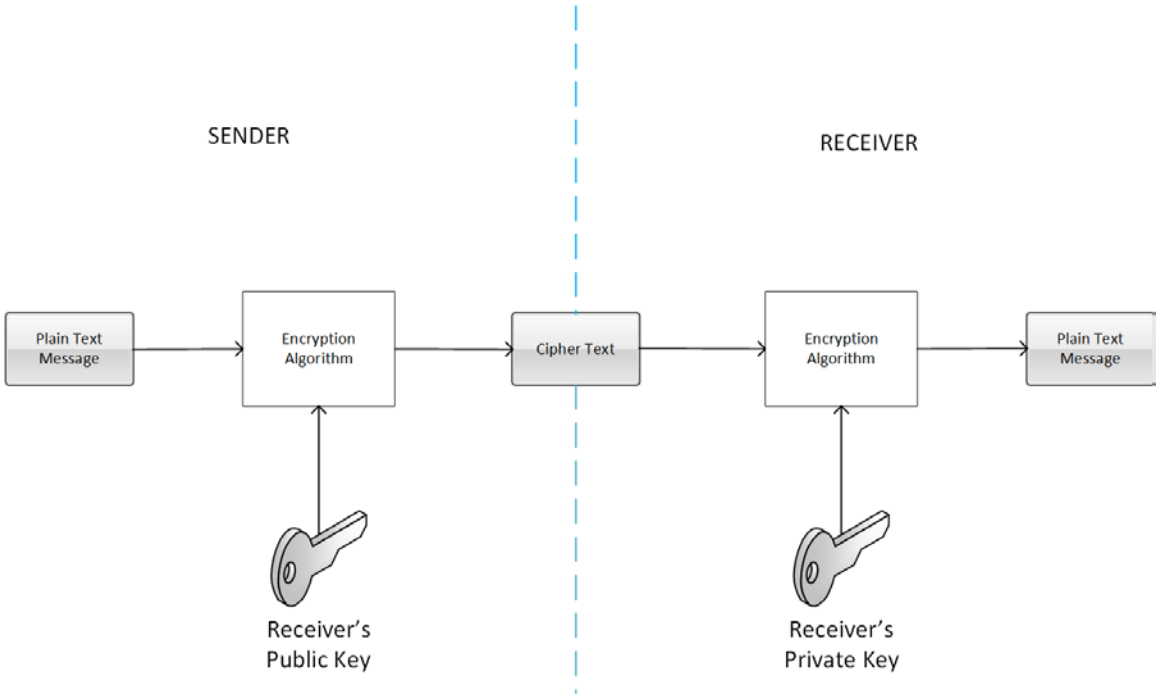
Quantum Key Distribution shares keys by embedding them in photons sent over a fiber optic cable network. See http://fortune.com/2013/10/14/unbreakable-encryption-comes-to-the-u-s/

Types of Encryption

There are two basic types of encryption.

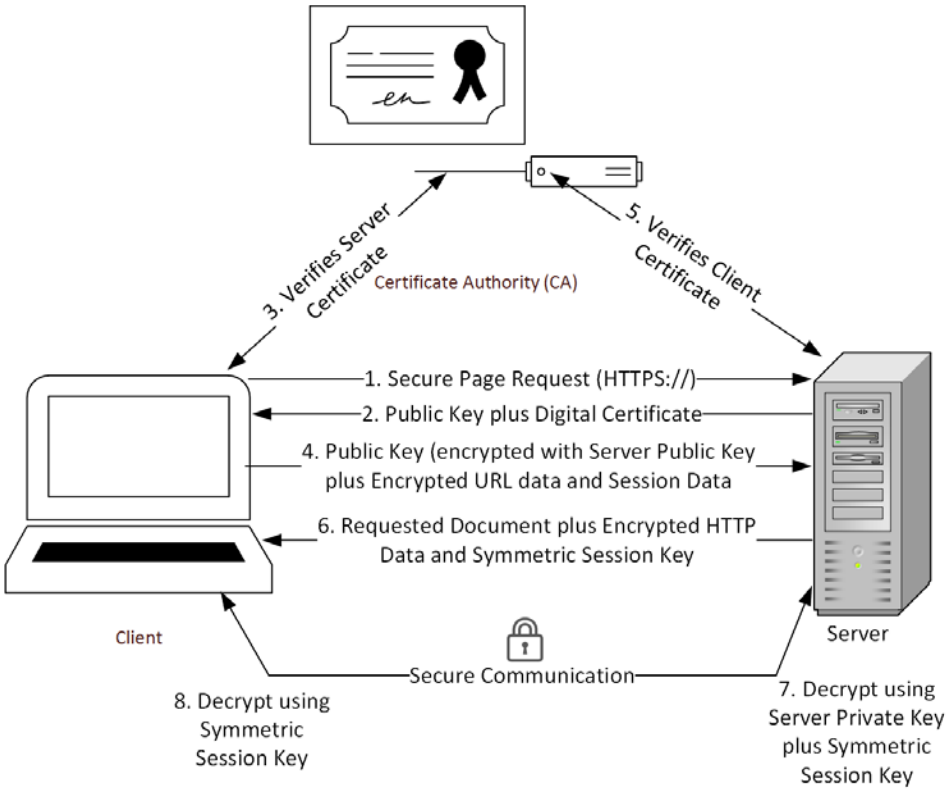
Symmetric key encryption uses a single key to encrypt plain text and to decrypt cipher text. While this type of encryption is often adequate for internal use within an organization, it is generally considered inadequate for the secure transfer of information across a public telecommunications network because both parties possess the same key. Such arrangements are particularly vulnerable to message spoofing where an unauthorized party encrypts data using a stolen key. When the receiving party receives the encrypted message and is able to decrypt it using the key they legally possess, they may assume they know the source of the message simply because they were able to successfully decrypt it. However, the ability to decrypt proves only one thing – whoever encrypted the message had the key required to do so. Triple DES, Blowfish, Twofish and AES use symmetric key encryption.

Asymmetric key encryption (often called ***public-key encryption***) uses a pair of keys—a ***public*** key and a ***private*** key. What the public key encrypts, the private key can decrypt, and vice versa. The names come from the expected use of the keys: the public key is given to anyone with which an enterprise does business, and the private key remains confidential and internal to the enterprise. To send a secure message that only the intended receiver can correctly interpret, you simply encrypt it with the receiver's public key. Only the intended recipient has the required key (their private key) for decryption. Here is an illustration:



However, even when public-key encryption is used, a single pair of keys is inadequate to fully protect data. If a manufacturer wishes to send secure messages to all of its customers, and it merely encrypts the data with its private key, anyone with the manufacturer's public key can decrypt the message, and since the manufacturer would have given its public key to all of its customers, the public key cannot be considered secure. If instead, the manufacturer uses a customer's public key to encrypt the message, then only the intended customer can decrypt the message. However, this does not prove that the message came from the manufacturer because, in theory, anyone could have found the manufacturer's public key and used it to encrypt the message. In order to establish communications that prove the authenticity of a message, organizations often use a ***certificate authority*** (CA), a third party that vouches for the authenticity of the sender by electronically signing the certificate and providing a public-private key pair for use with the certificate. (Certificate authorities usually charge for this service).

In HTTPS, or HTTP (HyperText Transfer Protocol) with SSL (Secure Sockets Layer), the client and server use certificates and a certificate authority to authenticate each other, and once authenticated, they exchange a symmetric session key (within a message encrypted using public-key encryption) to be used for the remainder of the session. Asymmetric key encryption is known to be slow, particularly the RSA algorithm. Therefore, switching to symmetric keys improves performance, while maintaining a secure exchange of information. Here is an illustration:



Encryption Guidelines

Here are some guidelines to follow regarding encryption:

- Encryption keys should be as long as possible. The longer the key, the more secure it is considered to be (within reason). However, longer keys lengthen the decryption process, so there is a tradeoff. Keys up to 128 bits have been routinely broken using various techniques. Most experts recommend at least 256 bit keys, longer if your algorithm and time requirements allow for longer keys.
- The loss of a symmetric key or a private key should be treated with the same seriousness as the loss of the data that it was used to encrypt.
- Sensitive data should be encrypted whenever it is permanently stored. Which data is considered sensitive is a judgment call that should be made by the business people who own the data, with the advice of legal counsel, not by the DBA. In general, however, any personal data (such as Social Security numbers and birthdates) that can be used for identity theft should be considered sensitive. Don't forget to encrypt backup media (tapes, CDs, DVDs, and so forth) – they are often more vulnerable to loss or theft than the primary storage devices.

- All data not considered public knowledge should be encrypted whenever transported electronically across network connections that are not otherwise encrypted. For example, if a company sends a purchase order file to a trading partner via FTP, the file should be encrypted, or a more secure method such as SFTP (Secure FTP) should be used. It is wise to assume that the bad guys are routinely monitoring public networks.
- E-mail is not considered secure, so any sensitive information to be sent via e-mail should be in an encrypted attachment instead of the main body of the e-mail message. Alternatively, some e-mail systems support encrypted and signed messages.