In the field of information security, there are a handful of essential topics that provide a foundation for understanding other technologies. One of these topics is **cryptography**, which is a body of knowledge that deals with the protection and preservation of information. In short, cryptography refers to a collection of techniques that scramble some messages so that only intended recipients can read them.

A **shift cipher** works by substituting each character with the character a certain number positions to the left or right of the current character. This is called “shifting” the character. The Caesar cipher uses a key of 3, meaning A encrypts to D, B encrypts to E, and so on.

**Authentication** is the process of positively identifying a party as a user, computer, or service. Authentication of software drivers plays a vital role in system stability because having a driver signed and verified as coming from the actual vendor and not from some other unknown (and untrusted) source ensures that the code in question meets certain standards.

Some of the systems that make use of encryption include:

* Internet Protocol version 6 (IPv6), which uses encryption to authenticate, validate, and protect sensitive traffic
* IP Security (IPSec), which is a component of IPv6, is optional in IPv4, and is used in VPNs
* Simple Network Management Protocol (SNMP) v2 and higher
* Secure Sockets Layer (SSL), which makes extensive use of cryptography
* Transport Layer Security (TLS), the successor to SSL
* Secure Shell (SSH), a replacement for some older protocols
* FTP, Telnet, SMTP, POP3, HTTP

**Integrity** is the ability to verify that information has not been altered and has remained in the form originally intended by the creator.

Yet another service that encryption can provide is **nonrepudiation**, or the ability to have definitive proof that a message originated from a specific party. Common examples of nonrepudiation measures are digital certificates and message authentication codes (MACs). One of the more common uses of nonrepudiation is in messaging or email systems.

There are 2 basic types of cryptographic mechanisms: symmetric and asymmetric. The differences between the two mechanisms are significant. **Symmetric cryptography** uses a single shared key for encrypting and decrypting data, whereas **asymmetric cryptography** requires 2 keys, one public and one private. Any operation performed with one key can be reversed only with the other. Regardless of which type of algorithm you use, data is encrypted by applying a key to an encryption algorithm. The algorithm uses the key to perform mathematical substitutions, transpositions, permutations, or other operations on plaintext to create ciphertext.

Cryptography covers the confidentiality, integrity, and nonrepudiation of information, but originally, cryptography was used only to protect confidentiality.

* **Egyptian hieroglyphics** – in many ways, the colorful and mysterious glyphs that cover walls and tombs of ancient Egypt can be considered substitution cipher
* **Scytale** – the Spartans used this technique to send encoded messages to the front line. It used a rod of fixed diameter with a leather strap that was wrapped around it.
* **Polyalphabetic cipher (Vigenere cipher)** – a substitution cipher that uses multiple substitution alphabets.
* **JN-25** – this was an encryption process the Japanese used during World War 2 to encrypt sensitive information. Allied cryptographers broke the JN-25 code, and American military leaders were able to use it to their advantage.
* **Concealment cipher** – in this method, the message is present but concealed in some way; as an example, the hidden message may be the first letter in each sentence or every sixth word in a sentence.
* **One-time pad** – this technique uses a large, nonrepeating key. Each cipher key character is used exactly once and then destroyed. Keys must be completely random, or nearly so, and must be as long as the message.

**Confidentiality** – ensures that only authorized subjects can access data

**authenticity** – ensures that data can be verified as valid and can be trusted

**integrity** – ensures that only authorized subjects can modify data

**nonrepudiation** – provides positive evidence that a message or action originated with a certain party

The terms **algorithm** and **cipher** are used interchangeably to describe the formula or process used to perform encryption. A Caesar cipher has 27 keys.

**Symmetric encryption** uses the same key to encrypt and to decrypt data. When encrypting a given piece of data, there are two different approaches an algorithm can use: **stream cipher** or **block cipher**. Stream ciphers operate one bit at a time by applying a pseudorandom key to the plaintext. In a block cipher, data is divided into fixed lengths, or blocks (often 64 bits); all the bits are then transformed by the cipher to produce an output. The output size of each of these ciphers is the same as the input size, which means they can be used for real-time applications, such as voice and video.

The most widely recognized symmetric-key algorithm is **DES**. DES is so recognized because it was thought to be the gold standard of data encryption for years, and then it was shown that advances in hardware technology allowed DES to be cracked in just a matter of minutes (or even less). Other popular symmetric algorithms include:

* **3DES (aka Triple DES)** – a more secure version of DES that performs the equivalent of 3 rounds of DES encryption. (Yes, there was a Double DES algorithm, which was quickly found to be just as easy to crack as the original DES when using a clever “meet-in-the-middle” attack)
* **Advanced Encryption Standard (AES)** – the successor for DES that is far more resistant to brute-force attacks. AES is mathematically constructed to be virtually impossible to break using current technology
* **Blowfish** – a highly efficient block cipher that can have a key length up to 448 bits
* **International Data Encryption Algorithm (IDEA)** – uses 64-bit input and output data blocks and features a 128-bit key
* **RC4** – a stream cipher designed by Ron Rivest that is used by WEP
* **RC5** – a fast block cipher designed by Ron Rivest that can use a large key size
* **RC6** – a cipher derived from RC5
* **Skipjack** – a symmetric algorithm of 80-bit lengths developed by the National Security Agency (NSA)

**Key management** is the process of carefully considering everything that possibly could happen to a key, from securing it on the local device to securing it on a remote device and providing protection against corruption and loss. The following responsibilities fall under key management:

* keys should be stored and transmitted by secure means to avoid interception by any unauthorized entity
* keys should be generated by a pseudorandom process (rather than letting users pick their own keys) to prevent guessing the key
* the key’s lifetime should correspond with the sensitivity of the data it is protecting, and the authorization to use it needs to expire in a timely fashion
* keys should be properly destroyed when the process for which they were used has lapsed. The destruction of keys will be defined in the key-management policies of the organization and should be done so with respect to those policies

**Asymmetric key cryptography** is also called public key cryptography, the name by which it is commonly known. Asymmetric encryption was derived from group theory, which allows for pairs of keys to be generated such that an operation performed with one key can be reversed only with the other key in the pair. The key pairs generated by asymmetric encryption systems are commonly known as public and private keys. Anyone who has access to the public key can encrypt data, but only the holder of the corresponding private key can decrypt it.

**Diffie-Hellman** – a process used to establish and exchange asymmetric keys over an insecure medium. The “hard” problem it uses is modular logarithms.

**El Gamal** – a hybrid algorithm that uses asymmetric keys to encrypt the symmetric key, which is used to encrypt the rest of a message. Based on Diffie-Hellman, it also relies on discrete logarithms.

**RSA (Rivest, Shamir, and Adleman)** – patented in 1977. RSA symbolically released its patent to the public about 48 hours before it expired in 2002. RSA is still used in various applications and processes, such as e-commerce and comparable applications.

**Elliptic curve cryptography (ECC)** – this is based on the difficulty of solving the elliptic curve discrete logarithm problem. Because the algorithm is so computationally intensive, shorter key lengths offer better security relative to other algorithms using the same key length.

Creating a digital signature of existing data requires two main steps. First, the message or information to be sent is passed through a hashing algorithm that creates a **hash** to verify the integrity of the message. Second, the hash is passed through the encryption process using the sender’s private key as the key in the encryption process.

**Certificate authority (CA)** – the entity responsible for enrollment, creation, management, validation, and revocation of digital certificates.

**Registration authority (RA)** – an entity responsible for accepting information about a party wishing to obtain a certificate; Ras generally do not issue certificates or manage certificates in any way. In some situations, entries known as local registration authorities (LRAs) are delegated the ability to issue certificates by a CA.

**Certificate revocation list (CRL)** – a list of certificates that have been revoked prior to their assigned expiration, which is published by the CA.

**Digital certificates** – pieces of information, much like a driver’s license in the real world, that are used to positively prove the identity of a person, party, computer, or service.

**Certificate distribution system** – a combination of software, hardware, services, and procedures used to distribute certificates.

**Root CA** – the CA that initiates all trust paths. The root CA is also the principal CA for its domain. The root CA can be thought of as the top of the pyramid.

**Peer CA** – has a self-signed certificate that is distributed to its certificate holders and used by them to initiate certification paths.

**Subordinate CA** – a certification authority in a hierarchical domain that does not begin trust paths. Trust initiates from some root CA. In some deployments, it is referred to as a child CA.

An **X.509** certificate includes these elements:

* version
* serial number
* signature algorithm ID
* issuer name
* validity period (not before/not after)
* subject name
* subject public key info (public key algorithm/subject public key)
* issuer unique identifier (optional)
* subject unique identifier (optional)
* extensions (optional)
* certificate signature algorithm
* certificate signature

7 key management issues that organizations need to address:

* generation
* distribution
* installation
* storage
* key change
* key control
* key disposal

Several ways a hacker can target a PKI for attack:

* **sabotage** – the PKI components or hardware may be subjected to a number of attacks, including vandalism, theft, hardware modification, and insertion of malicious code. Most attacks are designed to cause denial of service (DoS).
* **Communication disruption/modification** – these attacks target communications between the subscribers and the PKI components. The disruption could cause DoS but may also be used by the attacker to mount additional attacks, such as impersonation of a subscriber or the insertion of fake information.
* **Design and implementation flaws** – these attacks target flaws in the software or hardware on which the subscriber depends to generate or store key material and certificates.
* **Operator error** – these attacks target improper use of the PKI software or hardware by the operators and may result in DoS or the disclosure or modification of subscriber keys and certificates.
* **Operator impersonation** – these attacks target the user by impersonating a legitimate PKI operator. As an operator, the attacker could do almost anything a legitimate operator could do, including generate keys, issue certificates, revoke certificates, and modify data.
* **coercion/social engineering** – these attacks occur when the administrator or operator of a CA is induced into giving up some control over the CA or creating keys and certificates under duress or trickery.

Hashes are not used for encryption but rather for authentication and for ensuring integrity and providing nonrepudiation. A one-way hash function is also known as a fingerprint. Some of the most common current and historical hashing algorithms include:

* **message digest 2 (MD2)** – an older one-way hash function used in the privacy-enhanced mail (PEM) protocols along with MD5.
* **Message digest 4 (MD4)** – a one-way hash function that provides a 128-bit hash of the input message. Although faster and more secure than MD2, it also has been shown to contain vulnerabilities.
* **Message digest 5 (MD5)** – an improved and redesigned version of MD4, producing a 128-bit hash. MD5 is the most common cryptographic hashing algorithm in current use.
* **HAVAL** – a variable-length, one-way hash function and modification of MD5.
* **Secure hash algorithm-0/1 (SHA-0/1)** – provides a 160-bit fingerprint. SHA-0 and SHA-1 are no longer considered secure and are vulnerable to attacks.
* **Secure hash algorithm-2 (SHA-2)** – a group of SHA algorithms that each processes messages up to 512-bit blocks and adds padding if needed to get the data to added up to the right number of bits.
* **Secure hash algorithm-3 (SHA-3)** – formally known as Keccak, this algorithm was selected in 2012 as the NIST SHA-3 standard. It supports the same key lengths as SHA-2 but is far more secure.

Some common cryptographic systems include the following:

* **Secure Shell (SSH)** – an application that provides secure remote access capabilities. SSH is viewed as a replacement for the insecure protocols FTP, Telnet, and the Berkeley r-utilities. SSH defaults to port 22. SSHv1 has been found to contain vulnerabilities, so it is advisable to use SSHv2.
* **Secure Sockets Layer (SSL)** – a means for transmitting information securely over the internet, introduced by Netscape. SSL is application independent. SSL is cryptographic algorithm independent. The protocol is merely a framework to communicate certificates, encrypted keys, and data. One of the most widespread uses of SSL (or its successor, TLS) is to transport HTTP traffic securely. (HTTPS)
* **Transport Layer Security (TLS)** – a successor to SSL, TLS encrypts the communication between a host and client. TLS is composed of two layers, including the TLS Record Protocol and the TLS Handshake Protocol.
* **IP Security (IPSec)** – an end-to-end security technology that allows two devices to communicate securely. IPSec was developed to address the shortcomings of IPv4. Although it is an add-on for IPv4, it is built into IPv6. IPSec can be used to encrypt just the data or the data and the header.
* **Password Authentication Protocol (PAP)** – a protocol that is used for authentication but is not secure because the username and password are transmitted in cleartext.
* **Challenge Handshake Authentication Protocol (CHAP)** – a protocol that is more secure than PAP because of the method used to transfer the username and password. Its strength is that it uses a hashed value that is valid for only a single logon transaction.
* **Point-to-Point Tunneling Protocol (PPTP)** – a protocol developed by a group of vendors, PPTP is composed of two components: the transport that maintains the virtual connection and the encryption that ensures confidentiality.
* **Layer 2 Tunneling Protocol (L2TP)** – a protocol used to transfer data over VPNs. Implements encryption with IPSec.
* **Secure Socket Tunneling Protocol (SSTP)** – a protocol that uses SSL technology to set up a secure VPN communication channel.

**Brute-force password attacks** try every possible sequence of keys until the correct one is found. One problem with the brute-force attack, however, is that as key lengths grow, so do the power and time required to break them.

* **Ciphertext-only attack** – an attacker has some sample of ciphertext but lacks the corresponding plaintext or the key. The goal is to find the corresponding plaintext to determine how the mechanism works.
* **Known plaintext attack** – the attacker possesses the plaintext and ciphertext of one or more messages. The attacker will then use this acquired information to determine the key in use.
* **Chosen ciphertext attack** – the attacker can decrypt a deliberately chosen ciphertext into the corresponding plaintext. Essentially, the attacker can “feed” information into the decryption system and observe the output.

There are several methods that can be employed to **attack and obtain passwords:**

* dictionary password attacks
* hybrid attacks
* brute-force password attacks
* rainbow tables

To crack a password, all an attacker must do is obtain a piece of software with a dictionary list, which is easily obtainable. In most cases, the dictionary list or word files contain long lists of various words that have been predefined and can be quickly downloaded for use.