Cipher algorithms and keys:

A mathematical formula designed to obscure the value and content of data.
Most cipher algorithms uses a key or more in order to encrypt/decrpy the data.
in order to increase the protection of the data, increasing the size of the keys is required. However the larger the key, the more computing time is needed in order to encrypt/decrypt the data.

Encryptin:

Is a method of transforming readable data 'plain-text' into a form that is unreadable. (data remains confidential and private)

There are two main components of encryption:

Symmetric cryptography:

With symmetric cryptography the same key is used for both encryption and decryption of the data.
The algorithm and key combination determines how the plain-text will be jumbled up, which is a process of substitution and transposition of those characters (if the algorithm or key are weak, then the encryption will also be weak)

256-bit AES is the bit length (the strength of the algorithm), higher number in algorithm means the stronger it is, but the slower to encrypt and decrypt. (IE: a door with many locks is more secure but it will take longer to open and close it).

Advantages	Disadvantages
-secure: using the right,long size (256-bit) symmetric key is very secure. Takes a million of years to brute-force it.	-key exchange: a major problem of symmetric key, usually the key is encrypted with a different key, and the recipient must already have the key that will be needed to decrypt the encrypted secret-key. This can lead to a never-ending dependency on another key.
-fast: it is relatively ease to do, giving a very good reading and writing performance.	-more damage if compromised: when compromised, every thing was encrypted with the specified key can be decrypted. Both communication sides will be comprimised.
-authentication: It provides a degree of authentication because data encrypted can be decrypted only with the same key.	

symmetric encryption algorithm types:

-Advanced Encryption Standard (AES) – most common.

-Data Encryption Standard (DES)

-Triple-DES (3DES)

-Blowfish

-RC4

-RC5

-RC6

Asymmetric cryptography:

Asymmetric cryptography a pair of keys are involved (public & private key). The public key is published and private key is secret. The data that is encrypted with the public key can be decrypted only with the private key.

Both keys are mathematically related and the two keys are generated at the same time. For example, any site uses HTTPS method has a public and private key, they use to exchange asymmetric session key in order to encrypt the communication.

(if encrypt with public key, private key is required to decrypt. and if encrypted with private key, public key is required to decrypt – its not possible to encrypt and decrypt using the same key.)

Advantages	Disadvantages
-key exchange: no need for key exchange. (encrypting and decrypting using different keys)	-slow: This encryption method is usually slow, because of the length of its keys.
-increased security: Large sized keys (1024-2048 bits)	
-authentication: This method allow Digital Signatures.	

symmetric encryption algorithm types:

-Rivest-Shamir-Adleman(RSA) – most common.

-Elliptic curve cryptosystem(ECC)

-Diffie-Hellman(DHE)

-El Gamal

Data integrity and message digests:

In the case of data transfer, data flowing between applications in a public network enviroment can flows across any number of nodes or networks, encrypted data prevent these nodes/networks from understanding the data, but because loss of controll on those nodes/networks, data can be altered before it reaches the destination. Even if the data is encrypted, this may cause a harm for the applications and distrupt it.

In order to exchange data probably and to ensure that it is not altered, Message Digest or Hash functions is used for this purpose.

Those functions presents a fingerprint of the data, if the data changes the message digest/hash changes in ways that cannot predict.

The message digest is calculated and appended to the encrypted data, when message is received the message digest is recalculated from the encrypted data and compared to the message digest that was appended to the original message, if the values do not match, this means that the data is corrupted and will not be processed.

the hash function takes data in any size and convert it into a fixed size string of characters. The main feature of hash function is that there is no way to convert back to the original input (one way hash function – no keys required), any change happens to the data will changes the hash value.
hash functions:-
MD2, MD4, MD5
HAVAL
SHA, SHA1, SHA256, SHA384, SHA512
TIGER

Digital signature:

this is used to validate the data's integrity.

The digital signature is basically a one-way hash or message digest of the original data that is encrypted with the signer's private key.

The recipient first uses the signer's public key to decrypt the digital singnature, then the recipient can see information about the hash algorithm so this enables the recipient to generate a one-way hash of the same data, and then comparing the hashes (received hash and generated hash), if they match this means the data has not changed since it was signed and the recipient can assure that the public key used to decrypt the digital signature corresponds to the private key used to create the digital signature.

Authentication:

This is the process of establishing that the sender/receiver is who claims to be.

Authentication is established through public-key certificates. And must be digitally signed by a third party who vouches for the authenticity.

Digital certificate:

This is an electronic document that is used to identify an entity and associate it with a public-key. This address the problem of impersonation. Those cetificates are issued by (CAs – certificate authority), CAs are entities that validate identities and issues certifications.

Clients and servers use cetificates issued by CA to determine the other certificates that can be trusted.

The certificate issued by the CA binds a praticular public key to the name of entity that the certificare identifies. And in addition to a public-key, the certificate includes the name of the entity, expiration date, CA's name that issued the certification, serial number and other information. But the important part is the certification includes the digital signature of the issuing CA.

SSL(Secure Socket Layer) & TLS(Transport Layer Security):

SSL and TLS are cryptographic technologys inclue the symmetric and asymmetrical algorithms, hashes, digital signatures, message authentucation codes..etc to make a working security protocol. They are designed to provide communication security over the network.

SSL is positioned as a protocol layer between the (TCP) layer and the Application Layer to form a secure connection between clients and servers so that they can communicate in a secure manner over a network by providing:
Privacy: where data messages are encrypted so that only the two application endpoints understand the data.
Integrity: where message digests detect if any data was altered in flight.
Authentication: which verifies the identity of the remote node, application, or user by using digital certificates.
SSL is the older encryption protocol and TLS is the new one (more secured).

the connection is private because a symmetric algorithm protected such as AES with asymmetric algorithm such as RSA is used to encrypt/decrypt the data transmission, the keys for the symmetric encryption are generated uniquely for each connection and based on secret negotiation at the start of the session, the server and client negotiate the details of which encryption algorithm and cryptographic keys to use before the first bite of data transmitted.

TLS-handshake:

When two systems uses TLS attempt to connect, each system need to verify that the other supports TLS, this process is called TLS-handshake.

In TLS-handshake both parties decide upon the TLS version, encryption algorithm, cipher suite.. etc that will be used in the procedure.

The handshake process:

1- the client request the server to open a secure line, server response with TLS versions and cipher suits it works with. Once they agree upon common ones the handshake starts.

2- the server sends copy of its public key, attached to its digital certificate to the client, the client checks if the certificate legitimate and moves into the next step.

3-the client uses the sever's public key and it's private key to encrypt a 'session key' which will be used in order to encrypt/decrypt data in the particular session (this key will become invalid when connection is terminated).
Using asymmetric encryption to encrypt symmetric encryption.

4- both parties test the connection by sending encrypted messages, if they can decrypt them using the session key, then the handshake was successfully and connection is secured.

SSL & TLS versions:

TLS 1.0: vulnerable to various attacks, supports weak cryptography.

TLS 1.1: no protocol vulnerabilities known, supports bad cryptography not commonly used.

TLS 1.2: the most used TLS, latest vesion of TLS, uses strong cryptography.

Diffie-Hellman (DHE):

This asymmetric cryptography method allows two parties who met in the first time to securely establish a shared secret key in unsecured chanel in order to secure their communications.

This method uses the (PFS-Perfect Forward Secrecy), which means that the encryption system changes the keys to encrypt and decrypt data frequently and automatically. This ensures minimal damage in case of compromise.

Cryptographic explanation:

a client-server example: the client and the server first agree and share two public numbers, ('g'-small prime number & 'n'-very big number[2000/4000bits])

1- the client picks random number (A) and it's a very big number and keeps it secret. same as the sever with number (B)

2- the client calculates g^A mod n . and server g^B mod n. both of them generate the result and share it with each other (result becomes public)

3- the client recalculate $(g^B)^A$ mod n. and server $(g^A)^B$ mod n
this is the same as g^{A*B} mod n.

4- client and server now have a secret shared key, and it is the result of g^{A*B} mod n.

HTTP (Hypertext transfer protocol):

HTTP is an application-layer protocol used for transmitting hypermedia documents.

It is designed to communicate between the web browser and web servers (request-response). This protocol works on port 80 and it is not encrypted. (data can be seen in plain-text).

HTTP/0.9:

- extremely simple request-response telnet friendly protocol called 'one-line protocol'.

- request nature is single line (method+ path for the requested document)

- GET is the only supported method.

- response type is hypertext only.

- connection is terminated immediately after the response.

HTTP/1.0:

- browser friendly protocol.

- provide header fields includes rich metadata about request and response(HTTP version, Status code, Content type).

- response is not limited to typertext(it supports scripts, stylesheets, media..etc).

- GET, HEAD, POST methods are supported.

- connection is terminated immediately after the response.

HTTP/1.1:

- the version that is commenly used.

- supports pipelined connections, compression/decompression transfer, content negotiation, virtual hosting.

- faster response and great bandwidth savings (cache support)

- long-lived connection

-GET, HEAD, POST, PUT, DELETE, TRACE, OPTIONS methods are supported.

Soruces:

https://www.ibm.com/docs/en/ztpf/1.1.0.14?topic=security-concepts

https://www.appviewx.com/education-center/tls-ssl-certificates/what-is-tls-ssl-protocol

https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP

DEPLOYING DEFFIE-HELLMAN FOR TLS