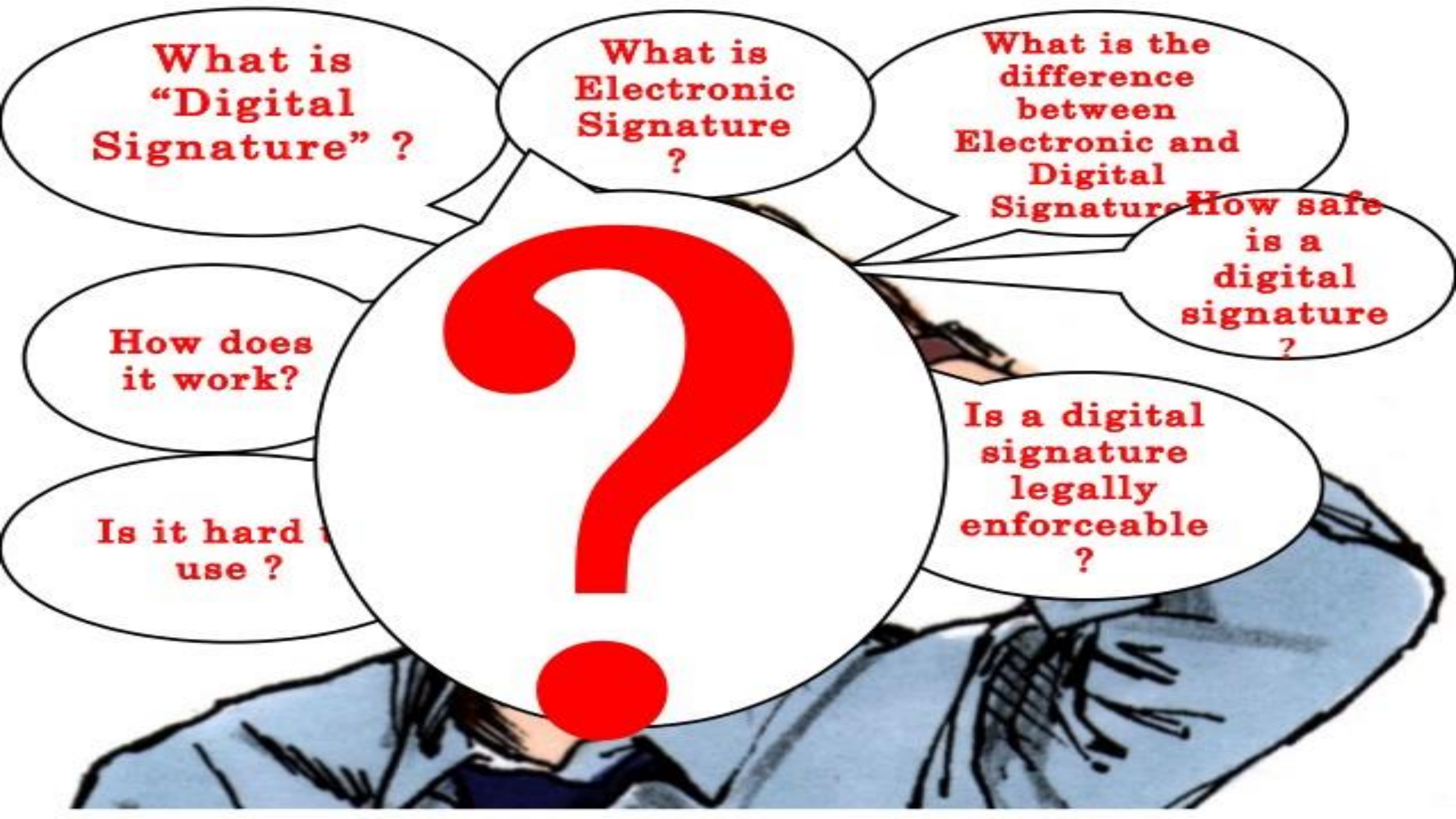
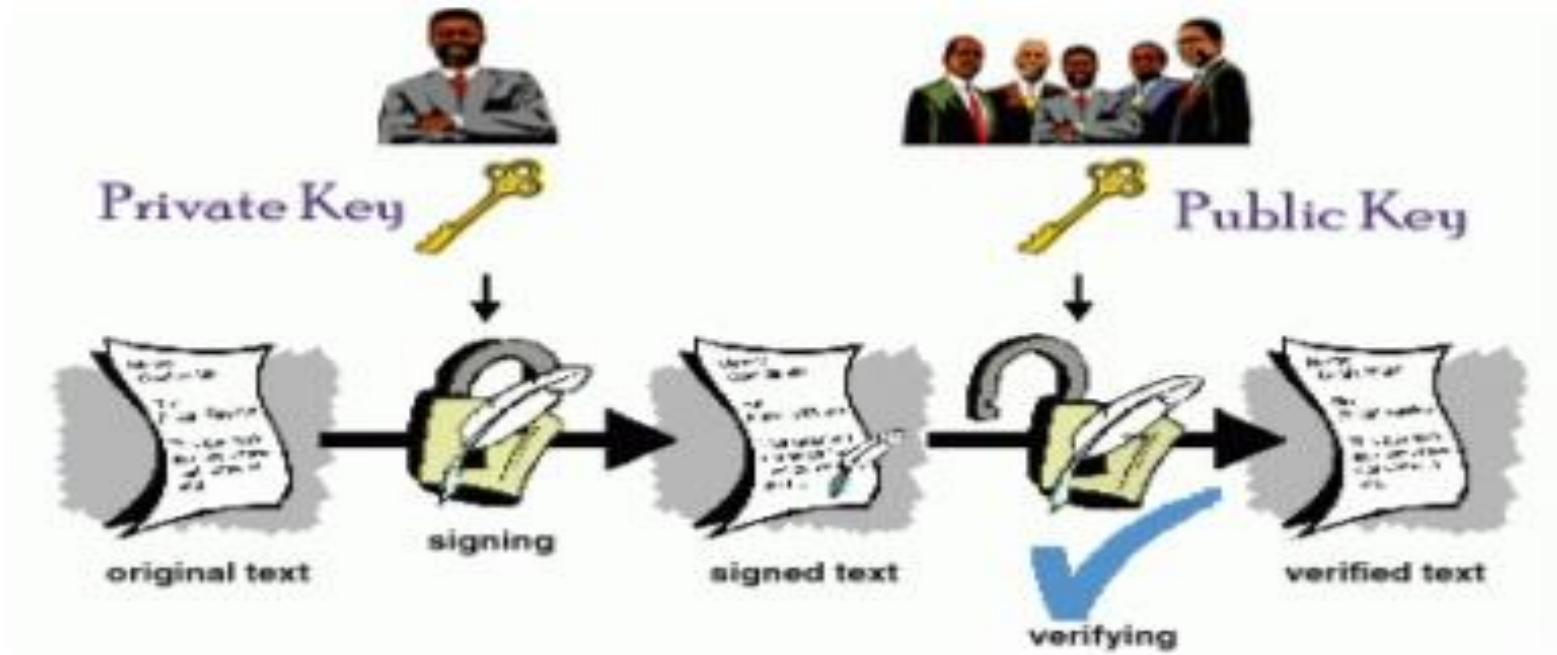

Digital Signature


What is "Digital Signature" ?
What is Electronic Signature ?
What is the difference between Electronic and Digital Signature
How safe is a digital signature ?
How does it work?
Is it hard
use ?
Is a digital signature legally enforceable ?

What is Digital Signature?

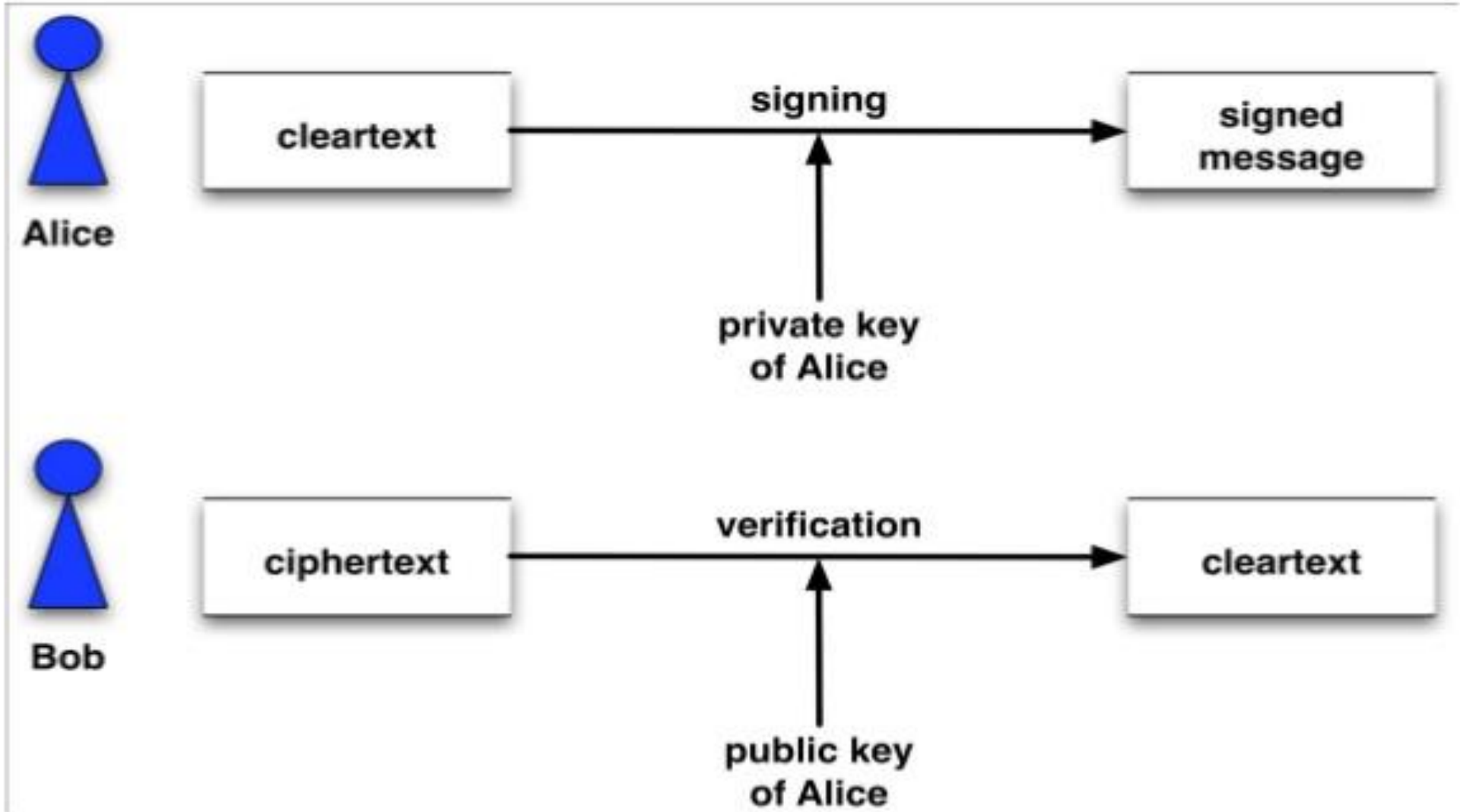
- Digital Signature is a type of electronic signature that encrypts documents with digital codes that are particularly difficult to duplicate.
- A digital signature (standard electronic signature) takes the concept of traditional paper-based signing and turns it into an electronic "fingerprint." This "fingerprint," or coded message, is unique to both the document and the signer and binds them together.
- It is used to validate the authenticity and integrity of a message, software or *digital* document. Digital signatures cryptographically bind an electronic identity to an electronic document and the signature cannot be copied to another document.


Private Key
Public Key
original text
signing
signed text
verifying
verified text

What is Digital Signature?

- Digital signature technique is based on public key cryptography with a difference.
- In public key cryptography a pair of keys are used, one public key and one private key. The public key is often used for message encryption , and the private key is often used for decrypting the message.
- However in case of digital signature message is encrypted with the private key and decrypted with the public key.
- Only a specific person with the corresponding private key can encrypt the message or in other words sign the message. However any party who has the signatory's public key can encrypt the message, in other words can verify the message.

How to Use?


Alice
cleartext
signing
signed message
private key of Alice
Bob
ciphertext
verification
cleartext
public key of Alice

Confidentiality Issues

- It should be possible for the receiver of a message to ascertain its origin. An intruder should not be able to masquerade as someone else.
- It should be possible for the receiver of a message to verify that it has not been modified in transit. An intruder should not be able to substitute a false message for a legitimate one.
- A sender should not be able to falsely deny later that he sent a message.

Attributes of Digital Signature

- Digital signature ensures the confidentiality via the following three attributes:

1. Authentication
2. Integrity
3. Non-repudiation

Attributes of Digital Signature

- **Authentication:** Authentication means *the act of proving who you say you are.* Authentication means that you know who created and sent the message. Digital signature is used to authenticate the source of messages. It ensures the user of the sender.
- **Integrity:** Integrity ensures that when a message is sent over a network, the data that arrives is the same as the data that was originally sent. Integrity is the assurance that the information is trustworthy and accurate. Digital signature ensures the integrity of message.
- **Non-repudiation:** this is an important criteria of digital signature. As digital signature ensures the authentication of the message, so the sender can't repudiate it later. At the same time it also ensures the identity of the receiver, so the receiver can't repudiate it later.

What is Electronic Signature?

- An electronic signature is a typed name or a scanned image of a handwritten signature.
- As a result, e-signatures are very problematic when it comes to maintaining integrity and security, as nothing prevents one individual from typing another individual's name.
- Due to this reality, an electronic signature that does not incorporate additional measures of security (the way digital signatures do) is considered an insecure way of signing documentation.

Difference Between Digital and Electronic Signature

- A digital signature, often referred to as advanced or standard version of electronic signature, that provides the highest levels of security and universal acceptance.
- Digital signatures are based on Public Key Infrastructure (PKI) technology, and guarantee signer identity and intent, data integrity, and the non-repudiation of signed documents. The digital signature cannot be copied, tampered with or altered.

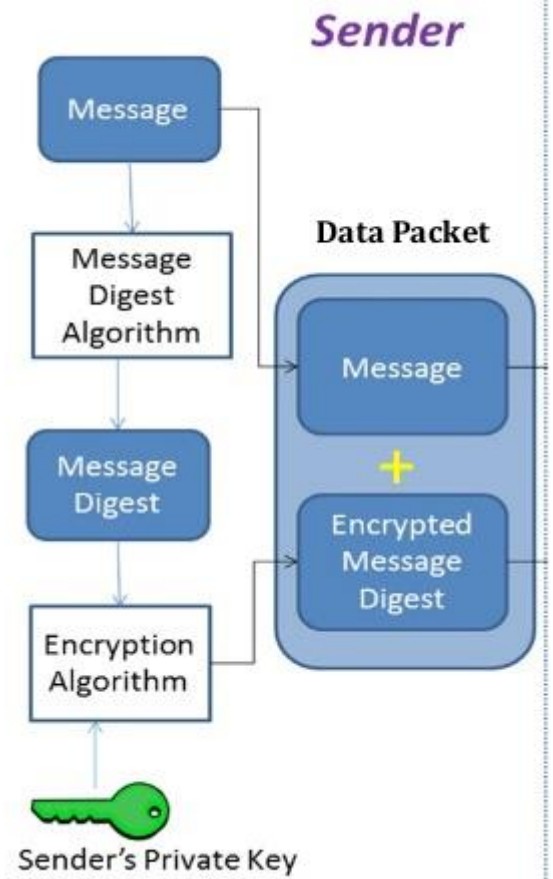
Difference Between Digital and Electronic Signature

- In addition, because digital signatures are based on standard PKI technology, they can be validated by anyone without the need for proprietary verification software.
- On the other hand, there is no standard format for electronic signatures that may be a digitized image of a handwritten signature, a symbol, voiceprint, etc., used to identify the author(s) of an electronic message.
- An electronic signature is vulnerable to copying and tampering, and invites forgery. In many cases, electronic signatures are not legally binding and will require proprietary software to validate the e-signature.

What is Standard?

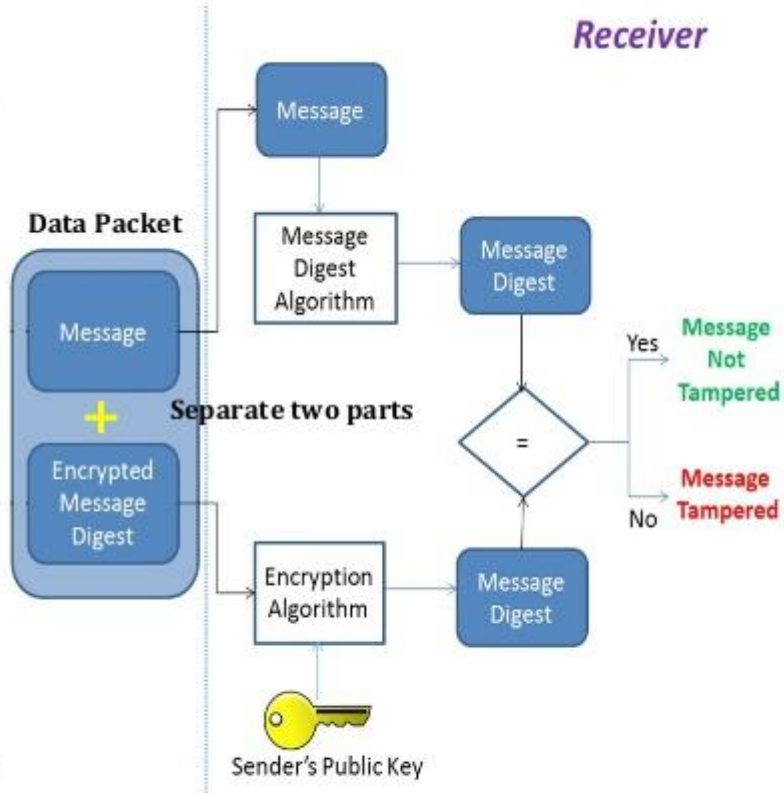
- There are three algorithms that are suitable for digital signature generation under the DSS standard.
- They are the Digital Signature Algorithm (DSA, which I will talk about more in depth later), the RSA algorithm, and the Elliptic Curve Digital Signature Algorithm (ECDSA).

Digital Signature Algorithm



1. Sender takes the input message.
2. Digest the input message using message digest algorithm
3. Apply DSA (using private key) to encrypt the digested message. Here the digital signature is padded into the digested message.
4. Create a data packet using the original message plus the encrypted message.
5. Send the packet to the intended receiver.

1. Receiver receives the packet.
2. Separate the original message and encrypted message.
3. Apply message digest algorithm on the original message.
4. Now Apply DSA (using public key) on the encrypted message to decrypt it. At this stage the digital signature is removed from the message and we get digested message.
5. Then compare them



DSA: Parameters

- **Signature Computation:** A DSA digital signature is computed using a set of domain parameters:
    1. **A private key x,**
    2. **A per-message or data secret number k,**
    3. **Data to be signed,**
    4. **And a hash function.**
- **Signature Verification:** A digital signature is verified using the same domain parameters:
    1. **A public key y** that is mathematically associated with the private key x used to generate the digital signature,
    2. **Data to be verified**,
    3. And **the same hash function** that was used during signature generation.

DSA: Parameters

- These parameters are defined as follows:

  p a prime modulus, where $2^{L-1} < p < 2^L$, and L is the bit length of p. Values for L are provided in Section 4.2.

  q a prime divisor of $(p - 1)$, where $2^{N-1} < q < 2^N$, and N is the bit length of q. Values for N are provided in Section 4.2.

  g a generator of the subgroup of order q mod p, such that $1 < g < p$.

  x the private key that must remain secret; x is a randomly or pseudorandomly generated integer, such that $0 < x < q$, i.e., x is in the range $[1, q-1]$.

  y the public key, where $y = g^x \bmod p$.

  k a secret number that is unique to each message; k is a randomly or pseudorandomly generated integer, such that $0 < k < q$, i.e., k is in the range $[1, q-1]$.

- The first three parameters, p, q, and g, are public and can be common across a network of users. x is private and y is public key.

DSA: Parameters

- Selection of Parameter Sizes and Hash Functions for DSA

This Standard specifies the following choices for the pair L and N (the bit lengths of p and q, respectively):

$L = 1024, N = 160$

$L = 2048, N = 224$

$L = 2048, N = 256$

$L = 3072, N = 256$

DSA: How Signature is Generated by the Sender ?

- Let N be the bit length of q. And let min(N, outlen) denote the minimum of the positive integers N and outlen. Outlen is the bit length of the hash function output block.
- The signature of a message **m** consists of the pair of numbers r and s that is computed according to the following equations:
  - ✓ $r = (g^k \bmod p) \bmod q$
  - ✓ **z= H(m)** [z = the leftmost min(N, outlen) bits of Hash(m)]
  - ✓ $s = (k^{-1} (H(m) + xr)) \bmod q$.
- When computing s, the string z obtained from Hash(m) is converted to an integer.
- (r , s) is considered to be the signature of the sender.

DSA: How Signature is Generated by the Sender ?

- The values of r and s shall be checked to determine if r = 0 or s = 0.
- If either r = 0 or s = 0, a new value of **k** shall be generated, and the signature shall be recalculated.
- It is extremely rare that r = 0 or s = 0 if signatures are generated properly.
- The signature (r, s) is transmitted along with the message to the verifier.

DSA: How Signature is Verified by the Receiver ?

- Signature verification may be performed by any party the signatory (sender), the intended receiver or any other party using the signatory's public key.
- A signatory may wish to verify that the computed signature is correct or not, before sending the signed message to the intended receiver.
- The intended receiver (or any other party) verifies the signature to determine its authenticity upon on receiving the message.

DSA: How Signature is Verified by the Receiver ?

- Let m', r', and s' is the received versions of m, r, and s, respectively. If every thing goes right then m‘=m, r‘=r, and s‘=s.
- Let y be the public key of the claimed signatory and let N be the bit length of q.
- Also, let **min(N, outlen) denote the** minimum of the positive integers N and outlen, where outlen is the bit length of the hash function output block.
- The signature verification process is as follows:
  - The verifier will check that $0 < r' < q$ *and* $0 < s' < q$. If either condition is violated, the signature shall be rejected as invalid.

DSA: How Signature is Verified by the Receiver ?

2. If the two conditions in step 1 are satisfied, the verifier computes the following:

   $w = (s')^{-1} \bmod q.$

   $z = H(m')$

   $u1 = (z * w) \bmod q = (H(m') * w) \bmod q$

   $u2 = (r' * w) \bmod q.$

   $v = [\ ((g^{u1} * y^{u2}) \bmod p) \bmod q\ ]$

The string z obtained from Hash(m') is converted to an integer

2. If v = r', then the signature considered to be verified.
   If m' = m, r' = r, and s'= s then v = r‘.
3. If v ≠ r', then the message or the signature may have been modified. The signature is **considered invalid.**

Example

- Alice wants to send message **m** to Bob.
- **Sender side:**

1. Alice generates a random number, k, less than
2. Alice generates

$$r = (g^k \bmod p) \bmod q$$

$$s = (k^{-1} (H(m) + xr)) \bmod q$$

1. The parameters (r,s) are her signature; she sends these to bob along with the signed message m'.

Example

- **Receiver Side:**

1. Let bob received the message and signatures as m', r', and s'. Then Bob verifies the signature by computing:

   $w = (s')^{-1} \bmod q.$

   $z = H(m')$

   $u1 = (z * w) \bmod q = (H(m') * w) \bmod q$

   $u2 = (r' * w) \bmod q.$

   $v = [\ ((g^{u1} * y^{u2}) \bmod p) \bmod q\ ]$

2. If v = r', then the signature and message considered to be authenticated.
3. If v ≠ r', The signature and message is considered invalid.