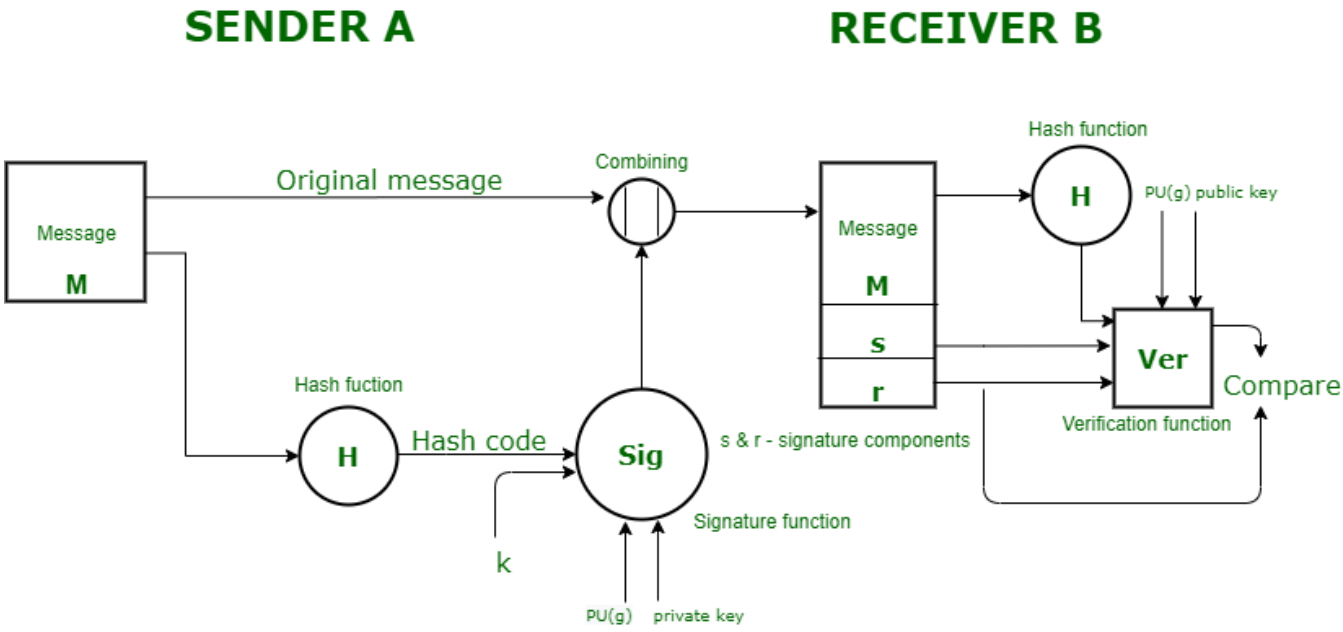
CYBER SECURITY & CRYPTOGRAPHY (part-2)

techworldthink • February 11, 2022

5. Explain important steps in DSS.

digital signature is a way of authenticating a digital data coming from a trusted source.

Digital Signature Standard (DSS) is a Federal Information Processing Standard(FIPS) which defines algorithms that are used to generate digital signatures with the help of Secure Hash Algorithm(SHA) for the authentication of electronic documents. DSS only provides us with the digital signature function and not with any encryption or key exchanging strategies.



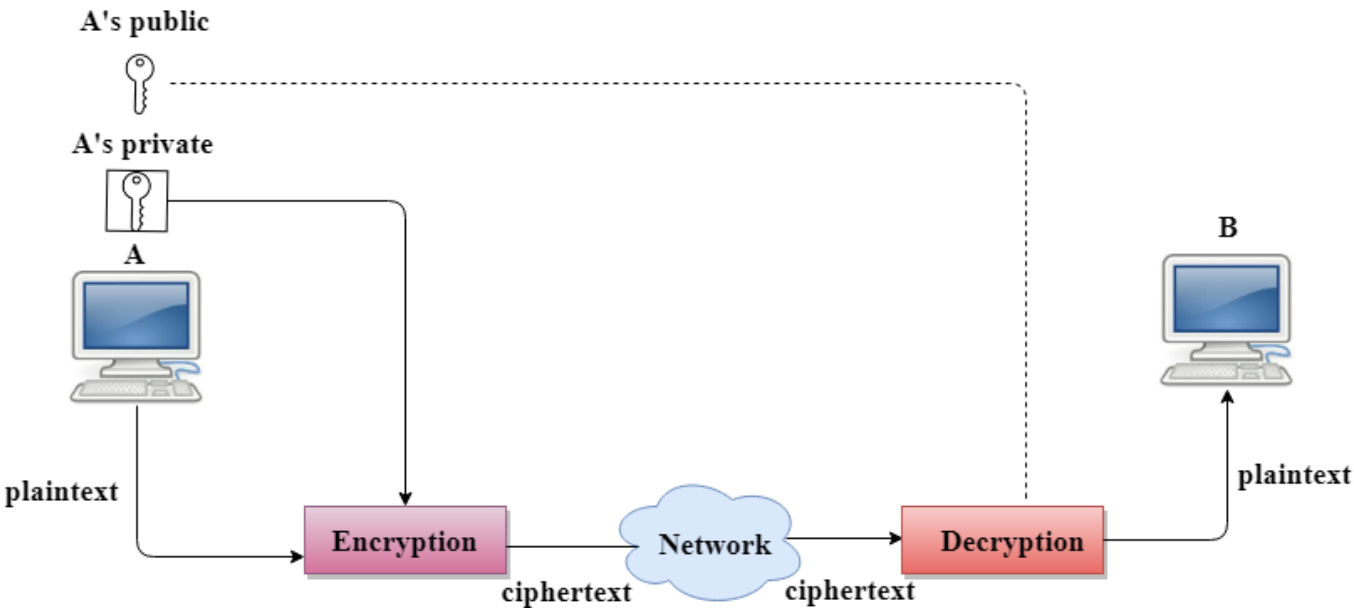
The Digital Signature is a technique which is used to validate the authenticity and integrity of the message.The basic idea behind the Digital Signature is to sign a document. When we send a document electronically, we can also sign it. We can sign a document in two ways: to sign a whole document and to sign a digest.

- In Digital Signature, a public key encryption technique is used to sign a document. However, the roles of a public key and private key are different here.

The sender uses a private key to encrypt the message while the receiver uses the public key of the sender to decrypt the message.

- In Digital Signature, the private key is used for encryption while the public key is used for decryption.
- Digital Signature cannot be achieved by using secret key encryption.

Signing the Whole Document



Digital Signature is used to achieve the following three aspects:

- Integrity: The Digital Signature preserves the integrity of a message because, if any malicious attack intercepts a message and partially or totally changes it, then the decrypted message would be impossible.
- Authentication: We can use the following reasoning to show how the message is authenticated. If an intruder (user X) sends a message pretending that it is coming from someone else (user A), user X uses her own private key to encrypt the message. The message is decrypted by using the public key of user A. Therefore this makes the message unreadable. Encryption with X's private key and decryption with A's public key results in garbage value.

- Non-Repudiation: Digital Signature also provides non-repudiation. If the sender denies sending the message, then her private key corresponding to her public key is tested on the plaintext. If the decrypted message is the same as the original message, then we know that the sender has sent the message.

Note: Digital Signature does not provide privacy. If there is a need for privacy, then another layer of encryption/decryption is applied.

Signing the Digest

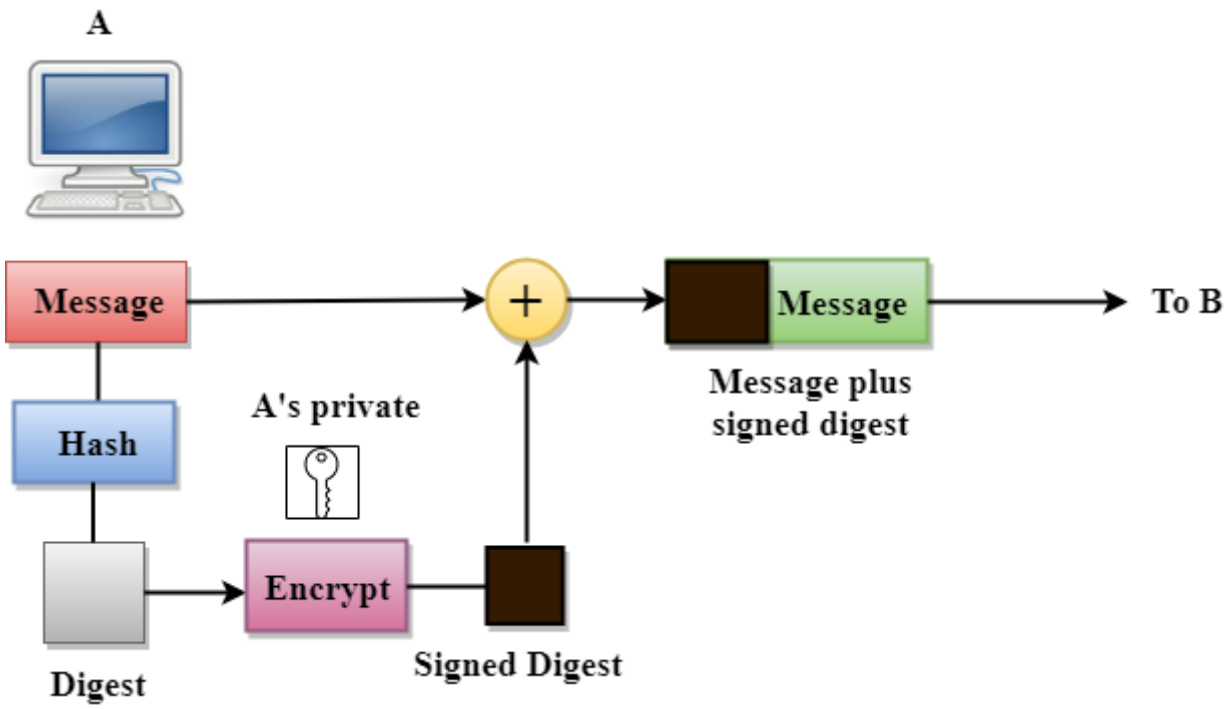
- Public key encryption is efficient if the message is short. If the message is long, a public key encryption is inefficient to use. The solution to this problem is to let the sender sign a digest of the document instead of the whole document.
- The sender creates a miniature version (digest) of the document and then signs it, the receiver checks the signature of the miniature version.
- The hash function is used to create a digest of the message. The hash function creates a fixed-size digest from the variable-length message.
- The two most common hash functions used: MD5 (Message Digest 5) and SHA-1 (Secure Hash Algorithm 1). The first one produces 120-bit digest while the second one produces a 160-bit digest.
- A hash function must have two properties to ensure the success:First, the digest must be one way, i.e., the digest can only be created from the message but not vice versa.
- Second, hashing is a one-to-one function, i.e., two messages should not create the same digest.

Following are the steps taken to ensure security:

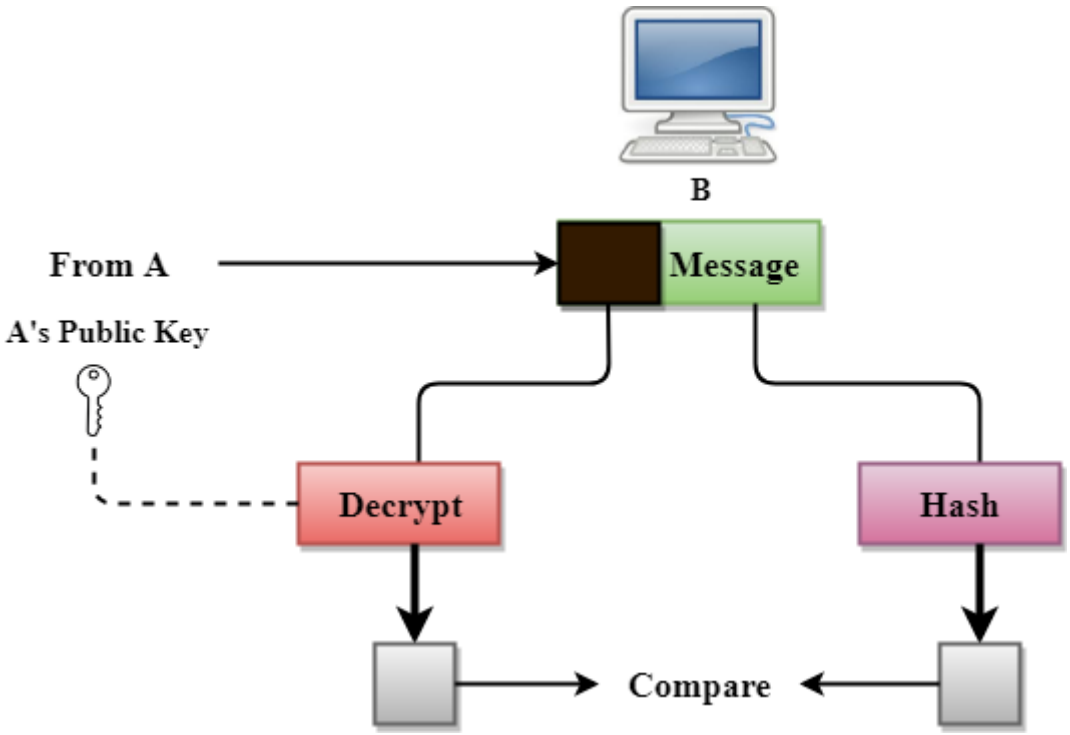
- The miniature version (digest) of the message is created by using a hash function.
- The digest is encrypted by using the sender's private key.
- After the digest is encrypted, then the encrypted digest is attached to the original message and sent to the receiver.

- The receiver receives the original message and encrypted digest and separates the two. The receiver implements the hash function on the original message to create the second digest, and it also decrypts the received digest by using the public key of the sender. If both the digests are same, then all the aspects of security are preserved.

At the Sender site



At the Receiver site



6. Describe the terms (a) birthday attack (b) hashcash (c) blind signature.

(a) birthday attack

Birthday attack is a type of cryptographic attack that belongs to a class of brute force attacks. It exploits the mathematics behind the birthday problem in probability theory. The success of this attack largely depends upon the higher likelihood of collisions found between random attack attempts and a fixed degree of permutations, as described in the birthday paradox problem.

Digital Signature Susceptibility

A digital signature is one area that is highly susceptible to a birthday attack.

- f – cryptographic function.
- *M*-message signed as *f(m)* using a specific secret key.

- So suppose Mark wants to cheat Jack by getting a fraudulent document signed by him.
- Mark makes a legitimate document called (**m**) and a fraudulent one named as (**m'**)
- Here if Mark changes (m) to (**m')** at several positions he will be able to create multiple variations of the legitimate document (m).
- Similarly, Mark also created several variations of fraudulent documents.
- Here Mark can use the hash function to match hash values of f(m)= f(m').
- Now even if Jack signs the legitimate document, Mark can easily replace it with the matching fraudulent document and prove that Jack originally signed the fraudulent document.

To conclude, security experts recommend that we should also use a very strong combination and a long sequence of bit length to prevent brute-force attacks.

(b) hashcash

Hashcash is a proof-of-work system used to limit email spam and denial-of-service attacks, and more recently has become known for its use in bitcoin (and other cryptocurrencies) as part of the mining algorithm. Hashcash was proposed in 1997 by Adam Back and described more formally in Back's 2002 paper "Hashcash - A Denial of Service Counter-Measure"

Hashcash is a cryptographic hash-based proof-of-work algorithm that requires a selectable amount of work to compute, but the proof can be verified efficiently. For email uses, a textual encoding of a hashcash stamp is added to the header of an email to prove the sender has expended a modest amount of CPU time calculating the stamp prior to sending the email. In other words, as the sender has taken a certain amount of time to generate the stamp and send the email, it is unlikely that they are a spammer. The receiver can, at negligible computational cost, verify that the stamp is valid. However, the only known way to find a header with the necessary properties is brute force, trying random values until the answer is found; though testing an individual string is easy, satisfactory answers are rare enough that it will require a substantial number of tries to find the answer.

The hypothesis is that spammers, whose business model relies on their ability to send large numbers of emails with very little cost per message, will cease to be profitable if there is even a small cost for each spam they send. Receivers can verify whether a sender made such an investment and use the results to help filter email.

(c) blind signature.

A blind signature scheme is a type of digital signature that conceals the identity of the message contents and the sender.

In these schemes the sender's message is concealed — or blinded — prior to the recipient signing it. Blind signature schemes are useful in applications where information on the sender is an important feature of the communication.

Electronic voting is a real-world example of a blind signature scheme is a properly functioning electronic voting system. In this example, the sender (voter) and the signer (recipient, voting authority) are unrelated and the sender's personal privacy and voting preference are paramount. The signature is considered valid enough to warrant the vote being recorded with confidence and the voter remains anonymous.

Going further, should the voting authority be called upon to validate the information they received they are able to validate the message's authenticity but unable to connect it with the sender (called unlinkability).