**Lecture 3 Notes**

Slide 4

* Digital signatures are like electronic “fingerprints”.
* In the coded message, the digital signature securely associates a signer with a document in a recorded transaction.
* Digital signatures use a standard, accepted format, called Public Key Infrastructure (PKI)

Slide 5

* Use of signatures is recorded in the Talmud (fourth century), complete with security procedures to prevent the alteration of documents after they are signed.
* The practice of authenticating documents by affixing handwritten signatures began to be used within the Roman Empire in the year AD 439, during the rule of Valentinian III.
* It is from this Roman usage of signatures that the practice obtained its significance in Western legal tradition.

Slide 6

* To provide Authenticity, Integrity and Non-repudiation to electronic documents
* To use the Internet as the safe and secure medium for Banking, e-Commerce and e-Governance with Security of Servers
* Alice signs a message—"Hello Bob!"—by appending to the original message a version encrypted with her private key. Bob receives both the message and signature. He uses Alice's public key to verify the authenticity of the message, i.e. that the message, decrypted using the public key, exactly matches the original message.

Slide 7

* Familiarize yourself with the following terms to better understand how digital signatures work:
* **Hash function** – A hash function (also called a “hash”) is a fixed-length string of numbers and letters generated from a mathematical algorithm and an arbitrarily sized file such as an email, document, picture, or other type of data. This generated string is unique to the file being hashed and is a one-way function— a computed hash cannot be reversed to find other files that may generate the same hash value. Some of the more popular hashing algorithms in use today are Secure Hash Algorithm-1 (SHA-1), the Secure Hashing Algorithm-2 family (SHA-2 and SHA-256), and Message Digest 5 (MD5).
* **Public key cryptography**– Public key cryptography (also known as asymmetric encryption) is a cryptographic method that uses a key pair system. One key, called the private key, encrypts the data and is kept secret. The other key, called the public key, decrypts the data and is distributed openly to others. Public key cryptography can be used several ways to ensure confidentiality, integrity, and authenticity. Public key cryptography can
  + Ensure integrity by creating a digital signature of the message using the sender’s private key. This is done by hashing the message and encrypting the hash value with their private key. By doing this, any changes to the message will result in a different hash value.
  + Ensure confidentiality by encrypting the entire message with the recipient’s public key. This means that only the recipient, who is in possession of the corresponding private key, can read the message.
  + Verify the user’s identity using the public key and checking it against a certificate authority.
* **Public key infrastructure (PKI)** – PKI consists of the policies, standards, people, and systems that support the distribution of public keys and the identity validation of individuals or entities with digital certificates and a certificate authority.
* **Certificate authority (CA)**– A CA is a trusted third party that validates a person’s identity and either generates a public/private key pair on their behalf or associates an existing public key provided by the person to that person. Once a CA validates someone’s identity, they issue a digital certificate that is digitally signed by the CA. The digital certificate can then be used to verify a person associated with a public key when requested.
* **Digital certificates** – Digital certificates are analogous to driver licenses in that their purpose is to identify the holder of a certificate. Digital certificates contain the public key of the individual or organization and are digitally signed by a CA. Other information about the organization, individual, and CA can be included in the certificate as well.
* **Pretty Good Privacy (PGP)/OpenPGP** – PGP/OpenPGP is an alternative to PKI. With PGP/OpenPGP, users “trust” other users by signing certificates of people with verifiable identities. The more interconnected these signatures are, the higher the likelihood of verifying a particular user on the internet. This concept is called the “Web of Trust.”

Slide 8

### How do digital signatures work?

* Digital signatures, like handwritten signatures, are unique to each signer. Digital signature solution providers, such as DocuSign, follow a specific protocol, called [PKI](https://www.docusign.com/how-it-works/electronic-signature/digital-signature/digital-signature-faq#pki). PKI requires the provider to use a mathematical algorithm to generate two long numbers, called keys. One key is public, and one key is private.
* When a signer electronically signs a document, the signature is created using the signer’s private key, which is always securely kept by the signer. The mathematical algorithm acts like a cipher, creating data matching the signed document, called a hash, and encrypting that data. The resulting encrypted data is the digital signature. The signature is also marked with the time that the document was signed. If the document changes after signing, the digital signature is invalidated.
* As an example, Jane signs an agreement to sell a timeshare using her private key. The buyer receives the document. The buyer who receives the document also receives a copy of Jane’s public key. If the public key can’t decrypt the signature (via the cipher from which the keys were created), it means the signature isn’t Jane’s, or has been changed since it was signed. The signature is then considered invalid.
* To protect the integrity of the signature, PKI requires that the keys be created, conducted, and saved in a secure manner, and often requires the services of a reliable [Certificate Authority (CA)](https://www.docusign.com/how-it-works/electronic-signature/digital-signature/digital-signature-faq#cert). Digital signature providers, like DocuSign, meet PKI requirements for safe digital signing.

Slide 9

* Although messages may often include information about the entity sending a message, that information may not be accurate. Digital signatures can be used to authenticate the source of messages. When ownership of a digital signature secret key is bound to a specific user, a valid signature shows that the message was sent by that user. The importance of high confidence in sender authenticity is especially obvious in a financial context. For example, suppose a bank's branch office sends instructions to the central office requesting a change in the balance of an account. If the central office is not convinced that such a message is truly sent from an authorized source, acting on such a request could be a grave mistake.
* In many scenarios, the sender and receiver of a message may have a need for confidence that the message has not been altered during transmission. Although encryption hides the contents of a message, it may be possible to *change* an encrypted message without understanding it. (Some encryption algorithms, known as nonmalleable ones, prevent this, but others do not.) However, if a message is digitally signed, any change in the message will invalidate the signature. Furthermore, there is no efficient way to modify a message and its signature to produce a new message with a valid signature, because this is still considered to be computationally infeasible by most cryptographic hash functions.

Slide 10

* A digital signature is a widely used security technique for assuring the integrity of data and authentication of its signer.
* The first step in creating a digital signature is to use asymmetric cryptography (also known as public-key cryptography) to generate a pair of keys:
* **Public key**: available to anyone. It is used during the authentication process.
* **Private key**: only known by the signer. It is used in the signing process.
* These keys are complementary: if data is encrypted with the private key, then the public key is required to decrypt the data. In a similar way, if the public key is used to encrypt the data, the private key should be used in the decryption.
* To generate the signature of a document, encrypt it with the private key, which only the signer has. Note that this does not provide any confidentiality: the public key is known to anyone, so anyone can decrypt the document. However, decrypting the signature with the public signer key and obtaining the same document guarantees that:
* The signature was generated by the private key. (Otherwise, the decryption procedure would fail.) And since only the signer has access to the private key, the signature was generated by the signer.
* The document has not been modified. (Otherwise, the document would not match the document obtained with the signature.)
* Since the signature is the same size as the document, the signed document ends up to be twice the document size. To solve this problem, a cryptographic hash is introduced.
* A cryptographic hash (also know as digest) is a function that maps data of any size to data of a fixed size and which is designed to be a one-way function. A cryptographic hash is impossible to invert. Several properties of the hash allows the function to be used in this context without introducing weakness in the scheme:
* Changes in the data produce changes in the hash of the data in an untraceable way.
* Given the hash of some data, it is infeasible to generate other data that has the same hash.
* Instead of encrypting the entire document to generate the signature, the signature is generated from just the hash of the document. This keeps the size of the signature relatively small. The image below shows the complete process:

Slide 11

* **Authenticating a signed document**
* To authenticate a signed document, follow this procedure:
* Decrypt the signature to obtain the hash of the original document.
* Compute the hash of the document.
* Compare the hashes. For them to match, the following conditions must be met:
* The document has not been modified. This provides the integrity guarantee.
* The key that was used to generate the signature is the private key associated to the public key that you used. This provides the authentication guarantee.
* If the hashes match, the signature is correct and the document is valid. The following diagram illustrates the authentication process:

Slide 12

* Bitcoin uses Elliptic Curve Digital Signature Algorithm (ECDSA) Based on elliptic curve cryptography Supports good randomness in key generation

Slide 13

* Alice generates 10 coins
* Sign the transaction A:10 using Alice’s private key and put that in the blockchain

Slide 14

* Alice transfers 5 coins to Bob
* Sign the transaction A-B:5 using Alice’s private key and put that in the blockchain

Slide 15

* Digital Signature using Public Key Cryptography
* Alice sign the message with her private key 𝑴′=𝑬(𝑴,)
* Alice send it to bob -M,M’
* Bob Verify the signature using Alice’s public key 𝑴=𝑬(𝑴’,)

Slide 16

Use the message digest to sign, instead of the original message to reduce the signature size.

* Sign the message with her private key S=𝑬(H(𝑴),)
* Alice send M,S to Bob
* Verify the signature using Alice’s public key H(𝑴)=𝑬(S,)

Slide 17

* Confidentiality – The ability to keep data hidden from unauthorized parties.
* Authentication – This property involves being able to verify that the other party is really who they say they are, and not some impostor or spy.
* Integrity – If data retains its integrity, it means that it hasn’t been altered or tampered with by anyone else.
* Non-repudiation – This property essentially means that the individual or entity who was responsible for an action cannot claim that they weren’t involved.

Slide 18

* Electronic Mail
* Data storage
* Electronic funds transfer
* Software Distribution
* eGovernance Applications

Slide 15

* **Digital Signature using Public Key Cryptography**
* Sign the message using the Private key
* Only Alice can know her private key
* Verify the signature using the Public key
* Everyone has Alice’s public key and they can verify the signature