CS 4732/57322 Homework #7

Due electronically by midnight August 3rd, 2020.

For submission, if done on paper please scan and submit as a pdf. If done in word, please submit the .docx or .doc format.

IMPORTANT: Clearly indicate outside resources utilized and sign below. Failure to cite use of outside resources will be reported for appropriate disciplinary actions. Note that discussions with other students are encouraged; copying – with or without modifications – is unacceptable and will also be reported.

I discussed one or more problems with the following people:

I hereby certify that any outside resources utilized, other than the textbook and class materials, are clearly cited. All other material I provide for this homework submission is my own original work.

Printed name

1. (6 points) Session keys and master keys are private and usually no more secure than each other. So why would it be fine to use a master key to distribute session keys? How do we gain anything?

Session keys- encryption or decryption keys that are generated to ensure the security of a communications sessions between a user and another computer.

Master key is a cryptographic key whose only purpose is to protect other keys like sessions keys during transmission. So when a transmission take place, entity first request the Key Distribution Centre to give master key of other entity , this forms the registration process and then the session keys are shared. It is better to use master keys to distribute session keys because master keys are stored in secure hardware in the cryptographic feature, . All other keys like session keys that are encrypted under these master keys are stored outside the protected area of the cryptographic feature.
Thus , it is an effective way to protect a large number of keys while needing to provide physical security for only a few master keys

2. [12 points] a) What is the difference between a digital signature and a MAC?

   Digital signature provides Integrity, Authentication as well as Non-repudiation. Whereas MAC provides only Integrity and Authentication but not Non-repudiation. Non-repudiation is the assurance that the third party cannot deny the validity or authenticity of the sender. Digital Signature uses asymmetric key pair for encryption and decryption whereas MAC uses symmetric key pair.

   b) Give an example of something you would use a MAC for but not a digital signature.

An example when MAC can be used but not a digital signature is to send a notification or signal to users that the network is unavailable. MAC is useful in this case because it is cheaper and more reliable as it has only one destination checking the authenticity.

   c) Give an example of something you would use a digital signature for but not a MAC.

An example when Digital Signature can be used but not MAC is checking online certificate status or list as it monitors the authenticity on both the sides with the use of asymmetric key pair. Digital signature confirms the authenticity of the organization the certificate belongs to.

3. (4 points) What is stored in a bitcoin wallet?

A Bitcoin wallet is analogous to a physical wallet. However, instead of storing physical currency, the **wallet** stores relevant information such as the secure private key used to access Bitcoin addresses and carry out transactions. Types of Bitcoin wallets are desktop, mobile, web, and hardware.

4. (8 points) If in bitcoin, addresses are derived from public keys, isn't that a weakness? Could someone use one of your public keys to generate a bitcoin address? If so, could they do anything nefarious with that? Explain your answer on what you think the limit of what they could do with it.

No, bitcoin addresses are derived from public keys, isn't weakness and yes you could send bitcoins directly to the public key: in fact, both Pay-to-Pub Key (P2PK) and Pay-to-Pub Key-Hash (P2PKH) were introduced in the first Bitcoin release. IIRC, P2PK is still used for Coinbase transactions sometimes, today.

P2PK transactions are slightly bigger for outputs but significantly smaller for inputs.

One advantage of P2PKH is that addresses are shorter than public keys. This allows addresses to be represented

with 34 characters in Base58check.
If there were a standard to present public keys in Base58check, they'd probably have 51 characters. Arguably, it is easier to type a character jumble that is only 34 characters than one that is 51 characters.

But really, addresses get used because there is a standard for them and there is none for public keys.

3 : No, because Bitcoin, as well as all other major cryptocurrencies that came after it, is built upon public-key cryptography, a cryptographic system that uses pairs of keys: public keys, which are publicly known and essential for identification. we can't create private key using someone's other public key.

4: No, because the security of this system comes from the one-way street that is getting from the private key to the public address. It is not possible to derive the public key from the address; likewise, it is impossible to derive the private key from the public key.

5. (6 points) Compare and contrast Public Key authorities with a Public Key directory.

Public Key Authority:	**Public Key Directory:**
It is the main concept for stronger security for public key distribution.	Its responsibility is to generate the public key for every system.
To distribute public key, the scheme Public Key Authority without Certification Authority (PKAw CA) and the scheme Public Key Authority with Certification Authority (CA) is used.	It is a publically available directory.
It maintains a greater degree of security.	It uses an algorithm.
The public authority maintains a directory with the name and public key for every participant. This directory is called Public Key Authority.	The public authority maintains a directory with the name and public key for every participant. This directory is called Public Key Authority.
Public Key Authority handles key distribution and management.	The scheme with an algorithm provides better security aspects for the distribution.
It builds on the public directory for providing stronger security providing tighter control over the distribution of the public keys.	The cons is it uses a lot of public key and nonce, because of which there are errors that are more likely to occur.
	It is vulnerable as a forgery in the creation of public keys can happen.
Users interact with the directory in real-time for obtaining the required public key.	To tackle forgery in the creation of public keys, it maintains a publicly available dynamic directory of public keys.
It has some drawbacks.	Some trusted entity is responsible for maintaining and distributing the public directory.

The public key authority could be a bottleneck or a single point of failure in the system as a user must contact and request the authority every time for public-key when it wants to communicate with anybody.	It is less secured than Public Key Authority.
	Public key is maintained by the authority directory.

6) (6 points) In the key exchange protocols that we saw, many of them used nonces. Yet these nonces are not shared and agreed upon, instead one of the parties picks one and then sends it to the other. What security in particular are these nonces used for?

When considering cryptography then nonce is considered to be an arbitrary number which could be used only once in the cryptographic communication which is similar in spirit with the nonce and is considered as the random or pseudo-random number which is being issued in that of the authentication protocol for ensuring the old communication to not be reused in the replay attacks.

These are mostly included in the data exchange by that of the protocol and is being used for the means to guarantee the transmission of the live data instead of replaying the data hence detecting as well as protecting against any replay attacks. This is mostly to be used for security and used in cryptographical communication.

7) (6 points) Using RSA for a digital signature, the message is hashed and then the hash is encrypted using RSA to create the signature. Why not just encrypt the entire message instead? It sure seems to be a more straightforward way to sign the message.

The main reason for this is one of the significant properties of a cryptographic hash function, it is impossible to find two different messages with the same hash. If the person encrypted the entire document with RSA and sends his private key along with it, then the main disadvantage is that if the adversary gets to know the private key, he would also know the message and can modify it.