# Homework 2

Please complete this assignment individually, each submission should be unique.

Total: 100 points

## Symmetric vs Asymmetric Encryption

1. What does it mean for an encryption to be ‘asymmetric’? (3 Points)

When encryption is asymmetric, it means that it has a public key for encryption and a private key for decryption. As the name suggests, the public key is shared with everyone, while the private key is kept secret, allowing only the intended recipient to decrypt the message.

1. If I want to send a message to Hanqiu and would like only us to be able to read it, what would be the most cost-effective method to encrypt it? (7 Points)

The most cost-effective way to encrypt the message would be symmetric encryption where both me and Hanqiu would have a shared public key for encryption and decryption. Symmetric encryption algorithms are generally faster and require less computational power than asymmetric encryption algorithms since there is only one key to encrypt and decrypt messages.

1. If I want to send all students in the class a message and cryptographically prove that the message came from me, how might I do so? (10 Points)

If you want to send students a message and cryptographically prove the message came from you, you can use asymmetric encryption with a digital signature. The encryption needs to be asymmetric because only the person with the private key can create a naïve digital signature verifiable by anyone who knows the public key. How this works is that, as the person with the private key, you sign the message with it, and the receivers can verify the message's authenticity with their public keys to ensure the message was sent by you and not tampered with.

## Chain of Trust

1. What is a certificate? (2 Points)

A certificate is a digital document used to verify a user, organization, or website and contains a public key. Certificates are issued by trusted entities called certificate authorities (CA) and include information like subject, public key, and CA signature.

Certificates are the most common format for public keys. For certificates on the web the owner of a certificate is called a subject and the common name is the identity of the owner.

1. What is a hashing algorithm? (2 Points)

Hashing algorithms are also known as message digests, one-way transformations, one-way functions, or hashing functions. Hashing algorithms usually have fixed lengths of 128 or 160 bits (16 or 20 bytes). Hashing algorithms are generally easy to compute and it is a one-way property where you are given H(m), but not m, but it’s challenging to find m. In this case, the weak collision-free hash algorithm is that H(m’) is the same as H(m), which makes it harder to find H(m). And in the case of strong collision-free, it is not feasible to find m1, m2 where H(m1) = H(m2). Hash functions are generally used to verify and/or create digital signatures, password hashing where the password isn’t needed to verify it, and message integrity using keyed hashes or an agreed secret key that doesn’t require an encryption algorithm so technology is exportable.

A hashing algorithm is a function that converts data of any size into a fixed-size string of characters, typically referred to as hash values. Hashes are used to ensure data integrity and are difficult to reverse-engineer, making them useful for creating digital signatures and verifying data.

1. What is a Certificate Chain? (2 Points)

A certificate chain is a sequence of certificates where each certificate is signed by the next higher-level certificate in the hierarchy. At the top of the chain is a root certificate that is trusted by all parties and intermediate certificates link the server’s certificate to the root.

A certificate chain generates a public key pair and certificate while erasing its own secrets and loads the new software, providing the new keypair and certificate as inputs. The hardware’s public key can be used to verify the certificate chain. Certificate chains prevent handoff attacks in the trusted boot. The trusted boot is necessary since we do not want to halt the system sometimes. Certificate chains cannot prevent privilege escalation attacks in order to maintain the certificate chain privilege code has to keep its private key.

1. What is a Hash Chain? (2 Points)

A hash chain is a series of hash values where each hash value is derived from previous ones. Hash chains can be used in cryptographic protocols to ensure data integrity and protect against tampering.

Hash chains can be more efficient in recording software measures as only a constant amount of secure memory is needed to record an arbitrarily long, append-only list of code identities. Hash chains are stored in secure memory and can defeat both privilege escalation and handoff attacks. The trusted platform module (TPM) uses hash chains to store measurements. Hardware-backed hash chains use a protected memory register initialized to a known value when the computer first boots and a hardware API is used to extend the identity of a code into the log. Given collision-resistant hash, the current register value guarantees the integrity of the append-only log. Malicious software cannot erase its identity from the log without rebooting the platform and losing control of the machine.

1. When are Certificate Chains preferable to Hash Chains? (7 Points)

Certificate chains are preferable when you need to establish trust in a digital identity or public key. The certificate chain allows you to verify that a certificate is valid and trusted by a root certificate authority. Hash chains are used more for ensuring data integrity rather than establishing trust in an identity.

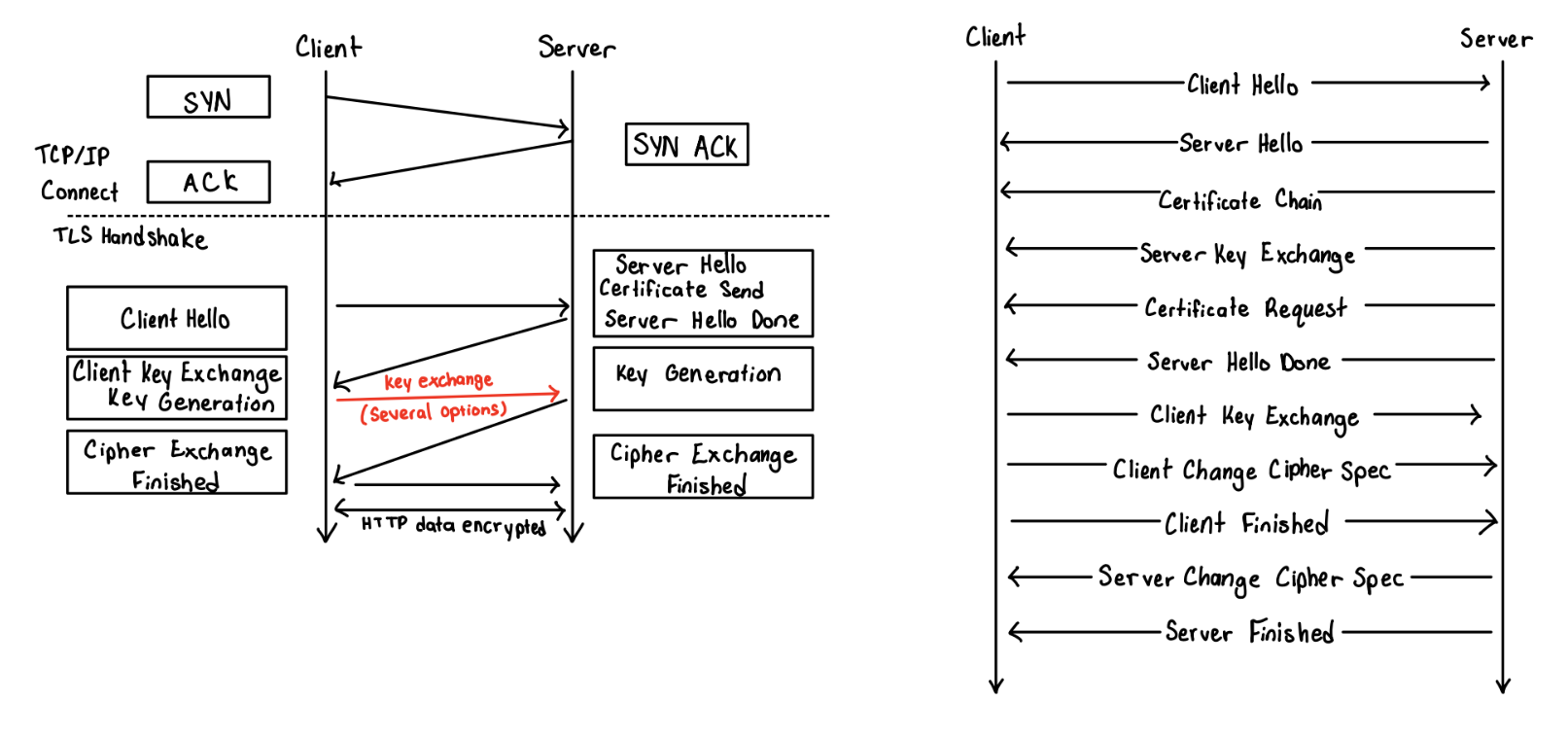
## TSL/SSL

1. What is a Man-in-the-Middle attack and how can it occur even with Asymmetric Encryption? (5 Points)

A Man-in-the-Middle (MitM) attack occurs when an attacker intercepts and potentially alters communication between two parties. This can happen even with asymmetric encryption if the attacker can intercept the public key in the exchange process, the attacker can replace the legitimate public key with their own and decrypt and re-encrypt the communication to allow them to read or modify messages.

Two private/public keys are created by a third party. The public key is transmitted from the sender to the receiver. This transmission is then intercepted by the third party using one of the created public keys while also calculating a shared key with the sender. The Receiver gets the public key calculates the shared key with the third party instead of the sender, and then transmits their public key to the sender. The third-party intercepts the transmission with the second public key to the sender while also calculating a shared key with the receiver. The sender receives the key and calculates the shared key with the third party instead of the receiver. The third-party can now intercept, decrypt, re-encrypt, and forward all messages between the sender and receiver. The Man-in-the-Middle attack is also known at the bucket brigade attack as it needs a reliable way to associate public keys with principal. The public key infrastructure (PKI) is one way. PGP web of trust is another. In some circumstances, it may be possible to use scheduling/timing to prevent MitM.

1. Please draw the process of a client and server authenticating each other through TLS/SSL. (7 Points)



## MQTT

1. Why is MQTT often referred to as a ‘Hub and Spoke’ model? (3 Points)

The MQTT is often referred to as a ‘Hub and Spoke’ model because all communication in the system is managed by a central broker (the hub). Clients (spokes) send and receive messages through the broker without directly communicating with each other. Brokers are also called servers that accept messages from clients and deliver the messages to any interested clients. Clients can publish, subscribe, or both to a topic ( namespace for massages on a broker)

1. Can Client A directly send a message to Client B in MQTT? (5 Points)

No, Client A cannot directly send a message to Client B in MQTT. In MQTT, messages are published topics and clients subscribe to these topics. The broker manages the distribution of messages, ensuring that only subscribed clients receive them.

1. How might Client A get a message to Client B in MQTT? (5 Points)

Client A can publish a message on a specific topic that Client B is subscribed to. When Client B subscribes to that topic, it will receive a message from the broker.

1. What are the two types of Man-in-the-middle Attacks possible in MQTT? (10 Points)

The two types of MitM attacks are message interception and replay. Message interception is when an attacker can intercept a message between the client and broker, generally what can happen if the connection is not encrypted (no TLS/SSL). Interception attack occurs when a third party obtains and changes a message, the message needs to be changed to be considered an attack otherwise it is considered sniffing where messages are obtained but not changed. Targets confidentiality, integrity, and accessibility. In confidentiality, it requests to send for public information, MitM changes the request to ask for private information, the server trusts the client and sends the information and the MitM receives the sensitive information. For integrity, it changes the sensor reading/command and the system reacts inappropriately. For accessibility, it changes all messages to nonsense and prevents access to it. In order to defend against interception there is traceroute, timing, encryption, and attestation. Traceroute identifies each node along the path a packet travels and ensures all are trusted which is very difficult to impossible for internet communications. In timing the intercepting and changing message takes time into consideration for authentication. For encryption, the messages are encrypted, which protects against confidentiality and integrity threats, but accessibility can still be blocked by changing messages to nonsense. Attestation is confirming the identity of remote parties. MitM replay attacks are when the save and resend valid communication and potential responses to security questions. Encryption in this case may not help much since ciphertext is still the same and plaintext isn’t needed to get a correct response. Can also relate the message to system behavior and resent messages that certain desired behaviors. Additionally, it can be devastating even without knowing what is said. Inorder to defend against replay there are physical unblinable functions, unique and unpredictable responses to challenges, and nonce is another solution. Nonce as another solution can include randomly numbers when creating a connection, different ciphertexts when creating a connection, different ciphertexts with the same keys, prevent simple replay, not all algorithms support it, and part of attestation.

## Closing Thoughts

1. How might Client A send a message to Client B through MQTT such that no other Client than B, even if properly subscribed to the appropriate topic can understand the content of the message? (Hint: Look to Asymmetric Encryption). (15 Points)

A client can encrypt messages using Client B’s public key before sending it. Even though other clients might be subscribed to the same topic, only client B, who has the corresponding private key, would be able to decrypt the message.

1. Draw a system boundary for the IoT Board if deployed as an IoT Device. (15 Points)

