assignment

1. Cryptography

a. What is the difference between symmetric cryptography and asymmetric cryptography?

The basic difference is that symmetric cryptography uses the same key to encrypt and decrypt, while asymmetric cryptography uses different keys, that is, public key and private key.

SYMMETRIC KEY CRYPTOGRAPHY	ASYMMETRIC KEY CRYPTOGRAPHY
It is also known as secret-key cryptography as the encryption and decryption process uses the same key.	It is also known as public-key cryptography. It works in the opposite way of symmetric cryptography.
A single key for both decryption and encryption.	Two key are required in which one key will encrypt and the other one used to decrypt.
The size of the cypher text is smaller or same.	The size of the cypher text is larger or same.
The encryption process is very extremely fast.	The encryption process is too slow.
It can transfer a huge amount of data.	It transfers only a small amount of data.
Symmetric key cryptography provides confidentiality.	It provides both authenticity and confidentiality.
Examples: AES, DES, 3DES and RC4	Examples are ECC, El Gamal, Diffie-Hellman, DSA and RSA
In comparison, resource utilization is low than asymmetric key encryption.	Resource utilization is higher.

b. Given that both types of cryptography can protect security, why should we still need both of them?

That's because each approach comes with advantages and disadvantages. Two big trade-offs exist between symmetric and asymmetric cryptography: Speed and security.

First, we have speed, where symmetric cryptography has an enormous advantage over asymmetric cryptography. Symmetric cryptography is faster to run (in terms of both encryption and decryption) because the keys used are much shorter than they are in asymmetric cryptography. Additionally, the fact that only one key gets used (versus two for asymmetric cryptography) also makes the entire process faster.

In contrast, the slower speed of asymmetric cryptography not only makes the process of sharing messages far less efficient, but it can also create performance issues as network processes get bogged down trying to encrypt and/or decrypt messages with asymmetric cryptography. This can result in slow processes, issues with memory capacity and fast drainage on batteries.

Second, we have security, where asymmetric cryptography presents an advantage over symmetric cryptography. Symmetric cryptography carries a high risk around key transmission, as the same key used to encrypt messages must be shared with anyone who needs to decrypt those messages. Every time the key gets shared, the risk of interception by an unintended third party exists.

Asymmetric cryptography offers better security because it uses two different keys -- a public key which only gets used to encrypt messages, making it safe for anyone to have, and a private key to decrypt messages that never needs to be shared. Since the private key never needs to be shared, it helps ensure only the intended recipient can decrypt encoded messages and creates a tamper-proof digital signature.

c. What is the algorithm framework of RSA?

Key generation

INPUT:
Two large prime numbers p and q.

OUTPUT:
Public Key Components:$\{e, n\}$
Private Key Components:$\{d, n\}$

PROCEDURE:
$n \leftarrow p * q$

/* Compute Euler phi value of n */

$\phi(n) \leftarrow (p-1) * (q-1)$

Find a random number e, satisfying $1 < e < \phi(n)$ and $\gcd(e, \phi(n)) = 1$.

Compute a random number d, such that, $d \leftarrow e^{-1} mod(\phi(n))$

Key distribution

Suppose that Bob wants to send information to Alice. If they decide to use RSA, Bob must know Alice's public key to encrypt the message and Alice must use her private key to decrypt the message.

To enable Bob to send his encrypted messages, Alice transmits her public key (n, e) to Bob via a reliable, but not necessarily secret, route. Alice's private key (d) is never distributed.

Encryption

After Bob obtains Alice's public key, he can send a message M to Alice.To do it, he first turns M into an integer m (strictly speaking, the padded plaintext), such that 0 ≤ m < n by using an agreed-upon reversible protocol known as a padding scheme. He then computes the ciphertext c, using Alice's public key e, corresponding $m^e \equiv c \pmod{n}$. Bob then transmits c to Alice.

Decryption

Alice can recover m from c by using her private key exponent d by computing $c^d \equiv (m^e)^d \equiv m \pmod{n}$. Given m, she can recover the original message M by reversing the padding scheme.

2. Cryptanalysis

a. Given an n-bit password, what is the average trying time for cracking the password using a brute force attack? Provide the detailed derivation.

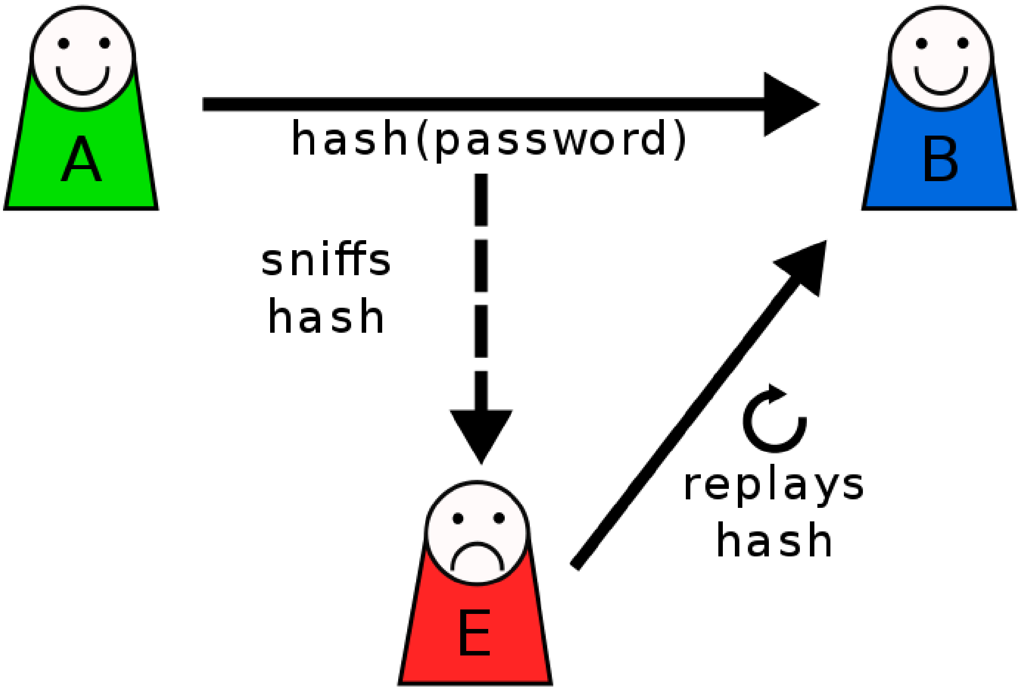
$\frac{1+2+3+...+2^n}{2^n}$

$= \frac{(1+2^n)2^n}{2^{n+1}}$

$= 2^{n-1} + \frac{1}{2}$

b. How does a replay attack work? How to address it?

Suppose Alice wants to prove her identity to Bob. Bob requests her password as proof of identity, which Alice dutifully provides (possibly after some transformation like hashing (or even salting) the password); meanwhile, Eve is eavesdropping on the conversation and keeps the password (or the hash). After the interchange is over, Eve (acting as Alice) connects to Bob; when asked for proof of identity, Eve sends Alice's password (or hash) read from the last session which Bob accepts, thus granting Eve access.



Replay attacks can be prevented by tagging each encrypted component with a session ID and a component number. This works because a unique, random session ID is created for each run of the program; thus, a previous run becomes more difficult to replicate. In this case, an attacker would be unable to perform the replay because on a new run the session ID would have changed.

We can also use session tokens, one-time passwords, nonces with MAC and timestamps to address it.

c. How does a man-in-the-middle attack work? How to address it?

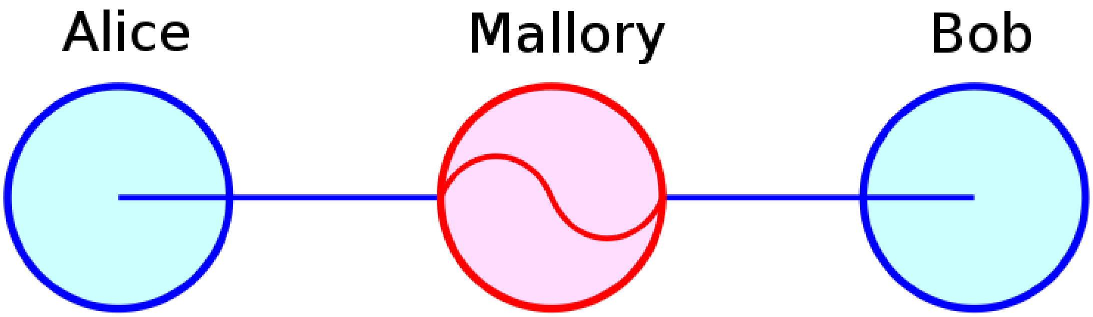
Suppose Alice wishes to communicate with Bob. Meanwhile, Mallory wishes to intercept the conversation to eavesdrop and optionally to deliver a false message to Bob.

First, Alice asks Bob for his public key. If Bob sends his public key to Alice, but Mallory is able to intercept it, an MITM attack can begin. Mallory sends Alice a forged message that appears to originate from Bob, but instead includes Mallory's public key.

Alice, believing this public key to be Bob's, encrypts her message with Mallory's key and sends the enciphered message back to Bob. Mallory again intercepts, deciphers the message using her private key, possibly alters it if she wants, and re-enciphers it using the public key she intercepted from Bob when he originally tried to send it to Alice. When Bob receives the newly enciphered message, he believes it came from Alice.

1. Alice sends a message to Bob, which is intercepted by Mallory:
   Alice "Hi Bob, it's Alice. Give me your key." → Mallory Bob
2. Mallory relays this message to Bob; Bob cannot tell it is not really from Alice:
   Alice Mallory "Hi Bob, it's Alice. Give me your key." → Bob
3. Bob responds with his encryption key:
   Alice Mallory ← [Bob's key] Bob
4. Mallory replaces Bob's key with her own, and relays this to Alice, claiming that it is Bob's key:
   Alice ← [Mallory's key] Mallory Bob
5. Alice encrypts a message with what she believes to be Bob's key, thinking that only Bob can read it:
   Alice "Meet me at the bus stop!" [encrypted with Mallory's key] → Mallory Bob
6. However, because it was actually encrypted with Mallory's key, Mallory can decrypt it, read it, modify it (if desired), re-encrypt with Bob's key, and forward it to Bob:
   Alice Mallory "Meet me at the van down by the river!" [encrypted with Bob's key] → Bob
7. Bob thinks that this message is a secure communication from Alice.

This is an example shows how man-in-the-middle attack works.



MITM attacks can be prevented or detected by two means: authentication and tamper detection. Authentication provides some degree of certainty that a given message has come from a legitimate source. Tamper detection merely shows evidence that a message may have been altered.

d. How does a relay attack work in wireless communication? How does distance bounding work against a relay attack?

- Radio signals or authentication messages between two devices (or people) may be hijacked

- An eavesdropping attacker may attempt to locate, intercept, and store a signal directly from a single device, e.g. a vehicle key fob, which constantly emits radio signals to check for the proximity of its owner's vehicle
- A criminal may send a signal to a victim's device in order to trick it into sending a response that they can then use to authenticate another device or application

Distance bounding could prevent the risk of relay attacks on contactless cards by measuring how long a card takes to respond to a request from a terminal for identification.

Since information cannot travel faster than the speed of light, the maximum distance between card and terminal can be calculated. By carefully designing the communication method cards use, this estimate can be made very accurate and ensure that relay attacks over even short distances (around 10m for our prototype) are detected.

3. Secure Routing

a. What are the key features of the five typical delivery schemes?

- unicast: deliver a message to a single node
- broadcast: deliver a message to all nodes in the network
- multicast: deliver a message to a group of nodes
- anycast: deliver a message to any one out of a group
- geocast: deliver a message to a group of nodes based on geographic location

b. What is the framework of the Dijkstra algorithm?

$c(i, j)$ - link cost from i to j (∞ if unknown)

$D(v)$ - current value of cost of path from source to destination v;

$p(v)$ - predecessor node along path from source to v;

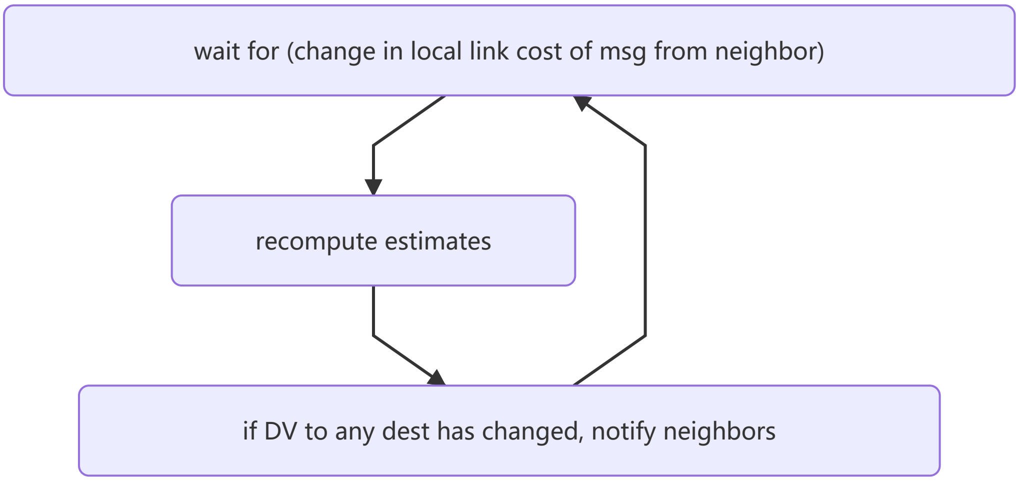
N' - set of nodes whose least cost path is already known;

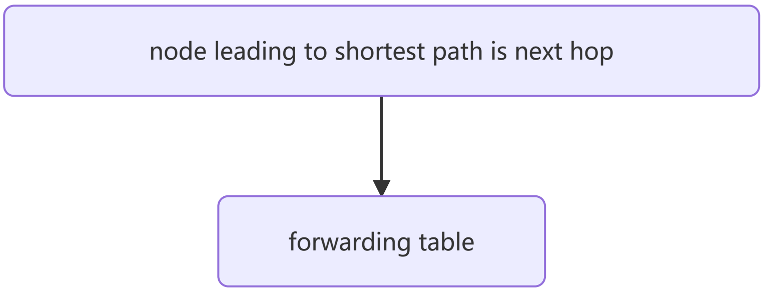
```
Initialization:
N' = {A}
for all nodes v
    if v adjacent to A
    then D(v) = c(A,v)
    else D(v) = ∞

Loop
find w not in N' such that D(w) is minimum
add w to N'
update D(v) for all v adjacent to w and not in N':
    D(v) = min(D(v), D(w) + c(w,v))
/* new cost to v is either the old cost, or known shortest path cost to w plus cost from w to v */
until all nodes in N'
```

c. What is the framework of the Bellman-Ford algorithm?

$D_x(y)$ - cost of least-cost path from x to y

$D_x(y) = \min\{c(x, v) + D_v(y)\}$ for all neighbors v of x


wait for (change in local link cost of msg from neighbor)
recompute estimates
if DV to any dest has changed, notify neighbors



d. How does prefix hijacking work?

Prefix hijacking is the illegitimate takeover of groups of IP addresses by corrupting Internet routing tables maintained using the Border Gateway Protocol (BGP).

A prefix is announced using BGP with an IPV4 or IPV6 address block and also a path of AS numbers, indicating which ASNs the traffic must pass through to reach the announced address block. By maliciously manipulating BGP IP prefixes, an attacker (IP hijacker) can reroute traffic in order to intercept or modify traffic.

Internet-level BGP hijacking is performed by configuring an edge router to announce prefixes that have not been assigned to it. If the malicious announcement is more specific than the legitimate one, or claims to offer a shorter path, the traffic may be directed to the IP hijacker. Internet hijacking attacks will frequently target unused prefixes to hijack in order to avoid getting identified by the legitimate owner.

By broadcasting false prefix announcements, the new compromised router may poison the Routing Information Base (RIB) of its peers and could propagate to other peers in a short period of time, to other ASes, and onto the Internet, so identifying route IP hijacking as soon as possible is critical for the security of your network.

e. How does RPKI work? Why is it insufficient for secure routing?

IP addresses and AS numbers are made available to 'trust anchors' by RPKI. This is achieved through an arrangement that connects Internet Number Resource Information by RPKI. A systematic distribution system that is routed through the IANA to the RIRs reaches the Local Internet Registries. The end-user accesses it from these local internet registries. Each receiver has its security system to ensure that hijackers are detected and warded off during the very first attempt itself.

Even with RPKI validation enforced, a BGP actor could still impersonate your origin AS and advertise your BGP route through a malicious router configuration, so it is insufficient for secure routing.

4. DDoS

a. What is the difference between DoS attacks and DDoS attacks?

The key difference between DoS and DDoS attacks is that the latter uses multiple internet connections to put the victim's computer network offline whereas the former uses a single connection. DDoS attacks are more difficult to detect because they are launched from multiple locations so that the victim can't tell the origin of the attack. Another key difference is the volume of attack leveraged, as DDoS attacks allow the attacker to send massive volumes of traffic to the target network.

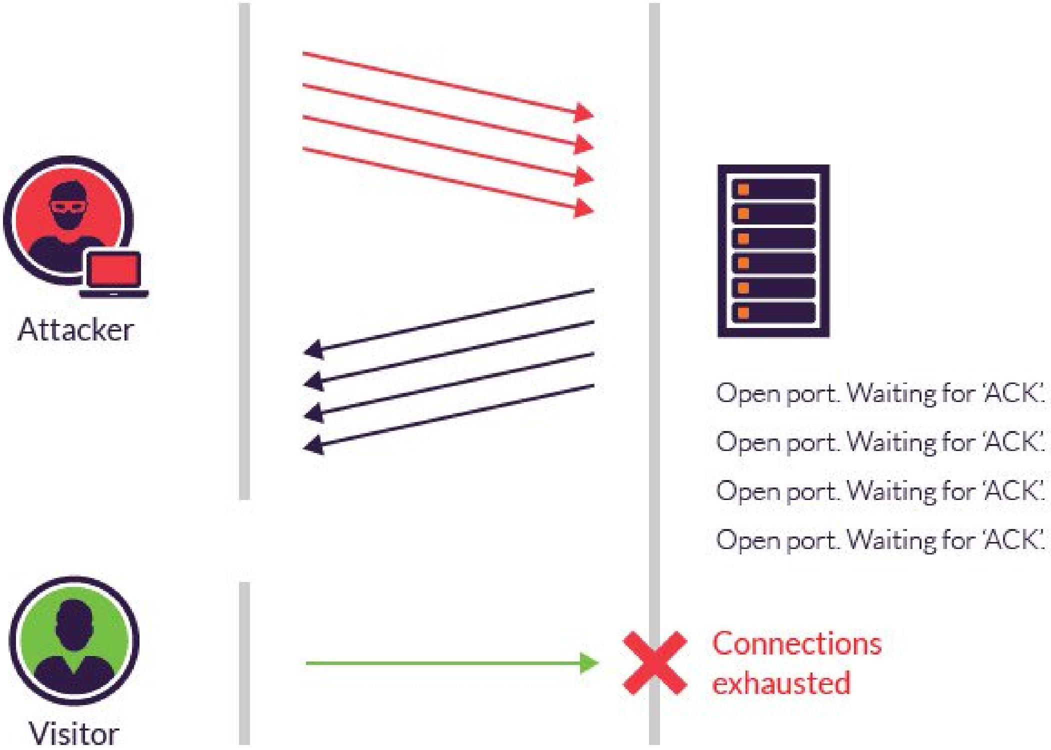
b. How does the TCP SYN Flood attack work?

When a client and server establish a normal TCP "three-way handshake," the exchange looks like this:

1. Client requests connection by sending SYN (synchronize) message to the server.
2. Server acknowledges by sending SYN-ACK (synchronize-acknowledge) message back to the client.
3. Client responds with an ACK (acknowledge) message, and the connection is established.

A SYN flood attack works by not reacting to the server with the normal ACK code. The pernicious customer can either basically not send the normal ACK, or by satirizing the source IP address in the SYN, bringing about the server to send the SYN-ACK to a distorted IP address – which won't send an ACK on the grounds that it "knows" that it never sent a SYN.

The server will sit tight for the affirmation for quite a while, as straightforward system clog could likewise be the reason for the missing ACK. In any case, in an attack, the half-open connections made by the pernicious customer tie resources on the server and may in the long run surpass the resources accessible on the server. By then, the server can't be access by any customers.


Attacker
Visitor
Open port. Waiting for 'ACK'.
Open port. Waiting for 'ACK'.
Open port. Waiting for 'ACK'.
Open port. Waiting for 'ACK'.
Connections exhausted

c. How does the solution of SYN Cookies against TCP SYN Flood attacks work?

SYN cookies using cryptographic hashing, the server sends its SYN-ACK response with a sequence number (seqno) that is constructed from the client IP address, port number, and possibly other unique identifying information. When the client responds, this hash is included in the ACK packet. The server verifies the ACK, and only then allocates memory for the connection.

c. How does the DNS Amplification Attack work? How to defend against it?

Attackers craft DNS requests in a way that substantially amplifies the size of the response. One way to do this is by requesting not just the IP address for a site, but information about the entire domain (for example, using DNS requests for the record type "ANY"), so the response might include details about subdomains, backup servers, mail servers, aliases, and more. Suddenly, a 10-byte DNS request could generate a response that's 10, 20, even 50 times larger.

We can defend against it by reducing the number of open resolvers and source IP verification to stop spoofed packets leaving network.

5. Blockchain

a. What are the key cryptographic techniques used in blockchain? What are they used for therein?

Asymmetric-key algorithms: Asymmetric-key cryptography is where the private key generally needs to be produced by a random number algorithm, and the public key is calculated by executing an irreversible algorithm. The asymmetric encryption algorithm has the advantage of having separate public and private keys, which can be transferred over unsecured channels.

One of the major parts of asymmetric-key cryptography is **digital signatures**. Digital signatures provide integrity to the process; they are easily verifiable and cannot be corrupted. They also hold the quality of non-repudiation, making them similar to the signatures in the real-world. The digital signatures ensure that the blockchain is valid and the data is verified and correct.

Hash functions: Hash functions are used to provide the functionality of a single view of blockchain to every participant. Blockchains generally use the SHA-256 hashing algorithm as their hash function. Hash functions have a major role in linking the blocks to one another and also to maintain the integrity of the data stored inside each block. Any alteration in the block data can lead to inconsistency and break the blockchain, making it invalid. This requirement is achieved by the property of the hash functions, called the 'avalanche effect'.

b. How is double spending addressed in blockchain?

The blockchain prevents double-spending by timestamping groups of transactions and then broadcasting them to all of the nodes in the bitcoin network. As transactions are time-stamped on the blockchain and mathematically related to the previous ones, they are irreversible and impossible to tamper with.

b. How does Proof of Stake work and save blockchain from intensive computation?

Proof-of-work is a necessary part of adding new blocks to the Bitcoin blockchain. Blocks are summoned to life by miners, the players in the ecosystem who execute proof-of-work**.** A new block is accepted by the network each time a miner comes up with a new winning proof-of-work.

Finding the winning proof-of-work is so difficult the only way to provide the work miners need to win bitcoin is with expensive, specialized computers. Miners will earn bitcoin if they guess a matching computation. The more computations they churn out, the more bitcoin they are likely to earn.

The goal of the miners is to create a hash matching Bitcoin's current "target." They must create a hash with enough zeroes in front. The probability of getting several zeros in a row is very low. But miners across the world are making trillions of such computations a second, so it takes them about 10 minutes on average to hit this target.

Whoever reaches the goal first wins a batch of bitcoin cryptocurrency. Then the Bitcoin protocol creates a new value that miners must hash, and miners start the race for finding the winning proof-of-work all over again.

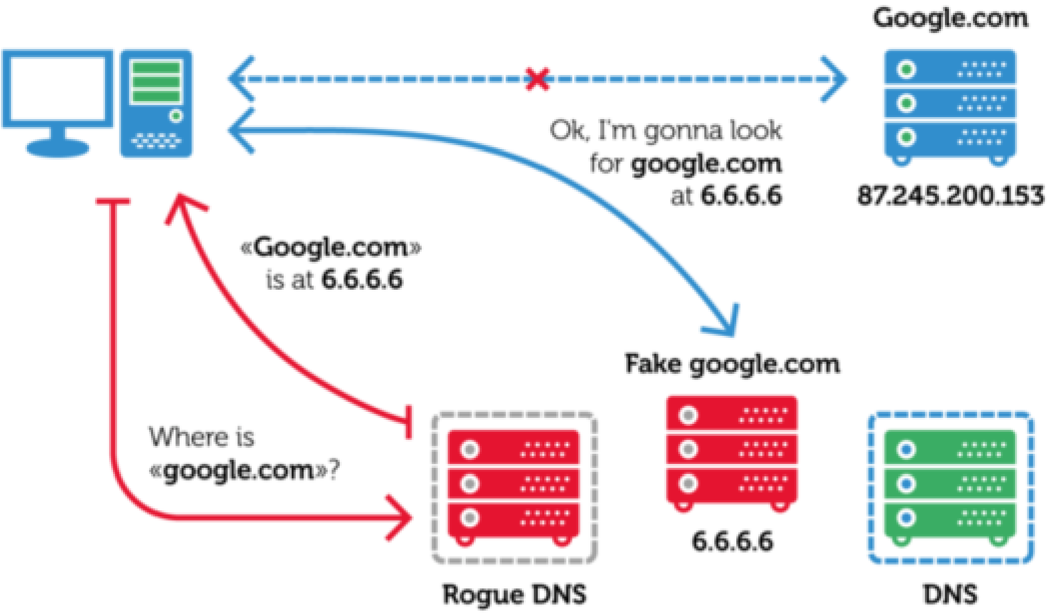
6. Secure Connection

a. How does a DNS hijacking attack affect network security?

Domain Name Server (DNS) hijacking, also named DNS redirection, is a type of DNS attack in which DNS queries are incorrectly resolved in order to unexpectedly redirect users to malicious sites. To perform the attack, perpetrators either install malware on user computers, take over routers, or intercept or hack DNS communication.

DNS hijacking can be used for pharming or for phishing (displaying fake versions of sites users access and stealing data or credentials).

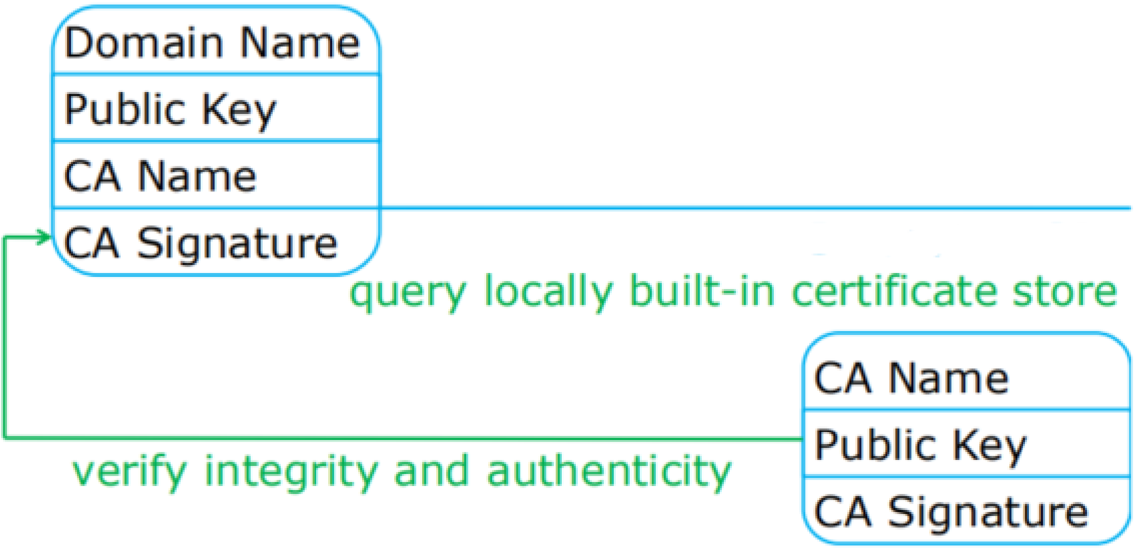
Many Internet Service Providers (ISPs) also use a type of DNS hijacking, to take over a user's DNS requests, collect statistics and return ads when users access an unknown domain. Some governments use DNS hijacking for censorship, redirecting users to government-authorized sites.



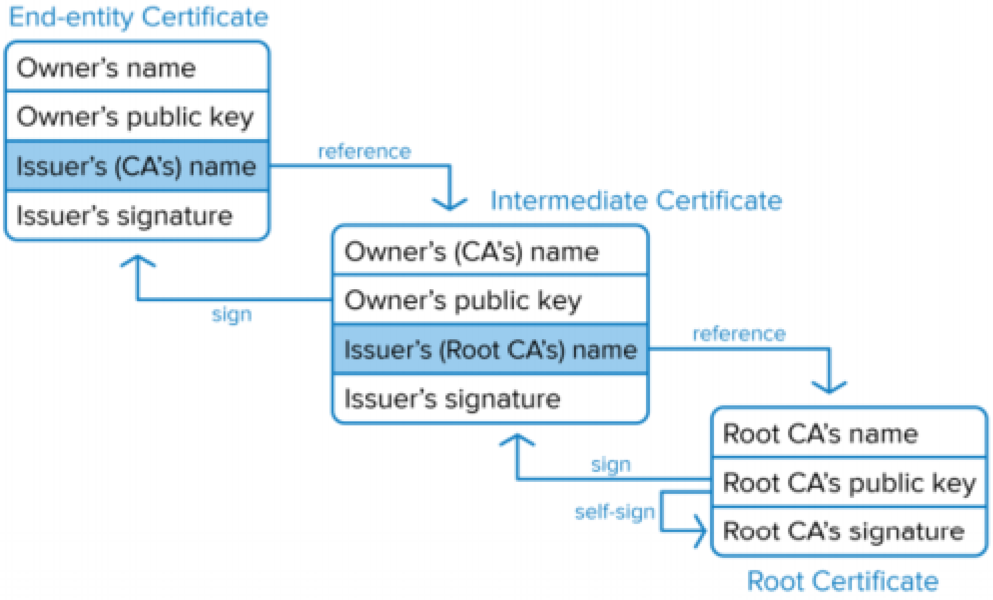
b. What is the protocol framework of HTTPS?

1. connection request
2. server response
3. certifificate verifification
4. key exchange
5. secure communication
6. end

c. How does a user verify a certificate for determining the authenticity of the website it connects to?


Domain Name
Public Key
CA Name
CA Signature
query locally built-in certificate store
CA Name
Public Key
CA Signature
verify integrity and authenticity

If signed by “branch” CA:



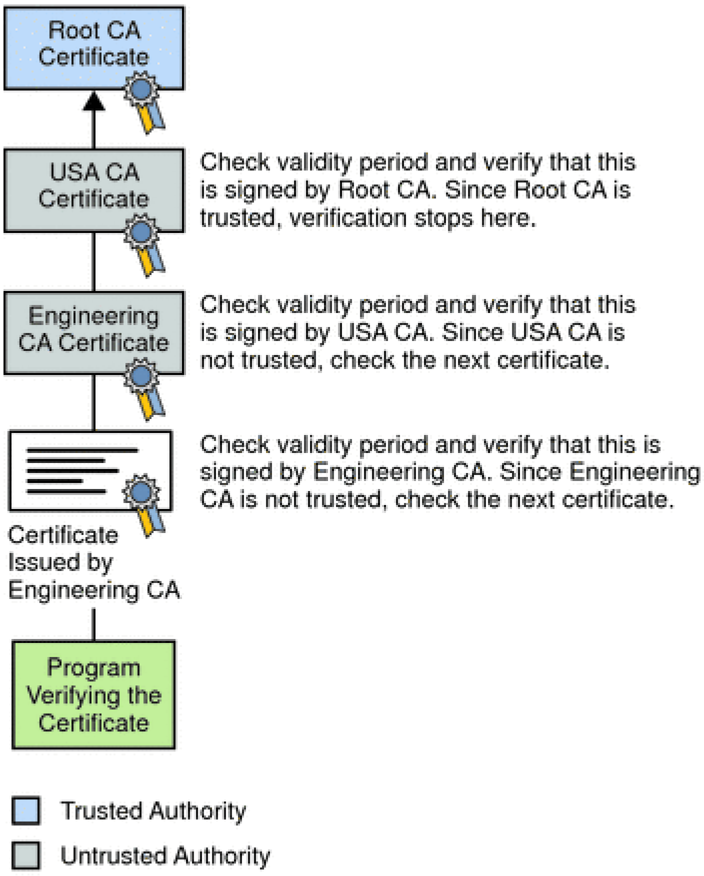
And enable CRL check.

1. Check if the certificate is issued by a trusted Certificate Authority (CA).
2. Check if the fully qualified hostname in the HTTPS request URL and the certificate owner match.
3. Check if the certificate is within its valid date range.
4. Check if the certificate is on a revocation list.
5. Checks if 1-4 are recursively applied to every certificate in the trust chain.

d. When is a certificate chain required? How to authenticate a certificate chain?

When signed by “branch” CA, a certificate chain is required. In general, a chain of multiple certificates may be needed, comprising a certificate of the public key owner (the end entity) signed by one CA, and zero or more additional certificates of CAs signed by other CAs. Such chains, called certification paths, are required because a public key user is only initialized with a limited number of assured CA public keys.

1. The certificate validity period is checked against the current time provided by the verifier’s system clock.
2. The issuer’s certificate is located. The source can be either the verifier’s local certificate database (on that client or server) or the certificate chain provided by the subject (for example, over an SSL connection).
3. The certificate signature is verified using the public key in the issuer certificate.
4. If the issuer’s certificate is trusted by the verifier in the verifier’s certificate database, verification stops successfully here. Otherwise, the issuer’s certificate is checked to make sure it contains the appropriate subordinate CA indication in the Directory Server certificate type extension, and chain verification returns to step 1 to start again, but with this new certificate.


Root CA Certificate
USA CA Certificate
Check validity period and verify that this is signed by Root CA. Since Root CA is trusted, verification stops here.
Engineering CA Certificate
Check validity period and verify that this is signed by USA CA. Since USA CA is not trusted, check the next certificate.
Check validity period and verify that this is signed by Engineering CA. Since Engineering CA is not trusted, check the next certificate.
Certificate Issued by Engineering CA
Program Verifying the Certificate
Trusted Authority
Untrusted Authority

7. Wi-Fi Security

a. What key properties of wireless communication make it more vulnerable to attacks than wired communication?

- Broadcast Communication

  Wireless networking typically involves broadcast communication, which is far more susceptible to eavesdropping and jamming than wired networks.

- Higher Mobility

  far more portable and mobile, thus resulting in a number of risks;

- Constrained Resource

  sophisticated OS but limited memory and processing resources to counter threats, including DoS and malware.

- Greater Accessibility

  may be left unattended in remote and/or hostile locations, thus greatly increasing their vulnerability to physical attacks.

b. Why is WEP insecure?

WEP can be attacked by message modification and message injection.

- Message modification takes advantage of CRC's linearity and unkeyed nature.

  C is the original cyphertext

  c is the CRC-32 function

  Δ is the change in the message

  $C' = C \oplus \langle \Delta, c(\Delta) \rangle$

- Message injection takes advantage of CRC's unkeyed nature and IV reuse.

  C is the original cyphertext

  P is the original plaintext

  $RC4(v, k)$ is the keystream for IV v

  M' is the new message

  c is the CRC-32 function

  Δ is the change in the message

c. How does IEEE 802.11i provide a higher security guarantee than WEP?

- Authentication
  mutual authentication enforced
  STA ←→ AP
- Access Control
  enforces authentication, routes messages properly, facilitates key exchange
- Privacy with Message Integrity

8. Anonymous Communication

a. Why is current Internet communication vulnerable to anonymity or privacy leakage?

For users to communicate via Internet, their devices assigned with IP addresses, which are usually fixed within a communication session or more. This can be used to infer critical privacy of users.

b. In which scenarios do users require the communication anonymity or privacy as concerned in sub-question a?

For Mortals:

- Unmonitored access to health and medical information
- Preservation of democracy: anonymous election/jury
- Censorship circumvention: anonymous access to otherwise restricted information

For Attackers:

- Misbehaviors without getting caught:
  - Terrorism
  - Darknet
  - Spam
  - Pirate

c. How to use proxies to secure communication anonymity? What are the possible limitations?

- intermediary between sender & receiver

- sender relays all traffic through proxy

- Encrypt destination and payload

- Asymmetric technique: receiver not involved (or informed of) anonymity

Limitations

- Require trusted third party proxy may release logs, or sell them, or blackmail sender
- Anonymity largely depends on the (likely unknown) location of attacker

d. How does Onion Routing provide a better guarantee for anonymity?

- Connect to Tor entry
- Randomly select a series of Tors
- Relay messages across them
- Tor exit relays messages to destination
- Reply traffic from destination traverses the reverse path
- Maintains a bidirectional persistent multi-hop path between source and destination

e. How to infer anonymity or privacy of Onion Routing traffic?

- Path Selection Attack
    - Tor path selection algorithm: weight nodes by selfreported bandwidth; select each node using weighted probability distribution;
- Counting Attack
    - Correlate incoming and outgoing flflows by counting the number of packets
- Low Latency Attack
    - Tor router assigns each anonymous circuit its own queue
    - Dequeue one packet from each queue in round-robin fashion
- Cross Site Attack
    - Search the accounts on public websites

9. Authentication Efficiency

Consider a time-consuming authentication scenario where a database records all secret keys of a large number of users. When the system authenticates a user, it first issues a challenge message to the user. The user then uses his/her key to encrypt the challenge and then returns the encrypted challenge to the system. The system then encrypts the challenge using one key in the database after another and compares the result with the received encrypted message. Once a match is found, the system accepts the user. Otherwise, the user is denied. This authentication protocol surely takes a lot of time and computation. Design a possible solution to speed up the authentication process.

After the user uses his/her key to encrypt the challenge, he/her can return the encrypted challenge and his/her id to the system. The system can then query by the given id and record the message. In this way, we reduce the traverse time.

666. SHINE YOUR WAY

Share your thoughts on the course project.

a. Do you aim for a research output from the course project? To what extent do you devote your time and energy to it? How do you overcome the associated challenges?

Emm, yes, I do aim for a research output from the course project and I wish I can devote as much time and energy to it as I have. I overcome the associated challenges by discussing with my group members, and thanks to their great intelligence, I think challenges are no more challengers. Most importantly, Google is the best teacher.

b. Do you think that you have gradually cultivated a research/security mindset? What is the most useful idea that you learned during this process?

Maybe?

Knowledge is infinite. Accept how 🥬 you are and just lie down.

c. Provide an example to showcase how you leverage that useful idea to facilitate problem solving in study or life.

When encountered with tough problems I can't solve at that moment, I choose to go sleep and let it be.

Design a question that you think is feasible as an exam question.

a. Which topic among the lectures you would like to consider?

WiFi Security

b. Describe a (sufficiently complex) question;

Describe the process of WEP encryption and decryption.

c. Provide also a *correct* sample solution, thanks.

Encryption

- Compute CRC for the message
- Compute the keystream
  - IV is concatenated with the key
  - RC4 encryption algorithm is used on the 64 or 128 bit concatenation
- Encrypt the plaintext
  - The plaintext is XORed with the keystream to form the ciphertext
  - The IV is prepended to the ciphertext

Decryption

- Build the keystream
  - Extract the IV from the incoming frame
  - Prepend the IV to the key
  - Use RC4 to build the keystream
- Decrypt the plaintext and verify
  - XOR the keystream with the ciphertext
  - Verify the extracted message with the CRC