Network Security I

Question 1 ***Internet layering*** **(10 min)**

(a) Complete the diagram showing the five layers of the internet protocol stack.

7	Application
4	Transport
3	Network
2	Link
1	Physical

(b) Write the number of the layer that corresponds to each protocol.

- ARP - Link (2)
- HTTP - Application (7)
- TCP/UDP - Transport (4)
- Voltage levels - Physical (1)
- IP - Network (3)

Question 2 ***DHCP*** **(10 min)**

Rafael gets home after a tiring day of lecturing CS 161. He opens up his laptop and goes on Twitter. From a networking and web perspective, what are the steps involved in loading the Twitter homepage?

Rafael's computer needs to connect to the wifi. What messages are exchanged in the 4 part handshake in order to achieve this?

Rafael's computer sends: _____.

This message is *broadcasted / unicasted*. Choose one and explain:

A DHCP server replies with a DHCP Offer. What does this message contain? What can a malicious attacker do at this step? Keep in mind that an attacker on the same subnet can hear the discovery message.

Rafael's computer sends: _____.

This message is *broadcasted / unicasted*. Choose one and explain:

The server then responds with: _____.

Solution: DHCP discover, broadcast. New host does not have an IP address and does not know what destination address to use.

The DHCP Offer message includes IP address, DNS server, gateway router, and lease time. Attacker can race the actual server; if they win, they can replace the DNS server and/or gateway router. Substitute a fake DNS server: Redirect any of a hosts lookups to a machine of attackers choice. Substitute a fake gateway router: Intercept all of a hosts off-subnet traffic (even if not preceded by a DNS lookup). Relay contents back and forth between host and remote server and modify however attacker chooses. An invisible Man In The Middle (MITM): Victim host has no way of knowing its happening (Cant necessarily alarm on peculiarity of receiving multiple DHCP replies, since that can happen benignly)

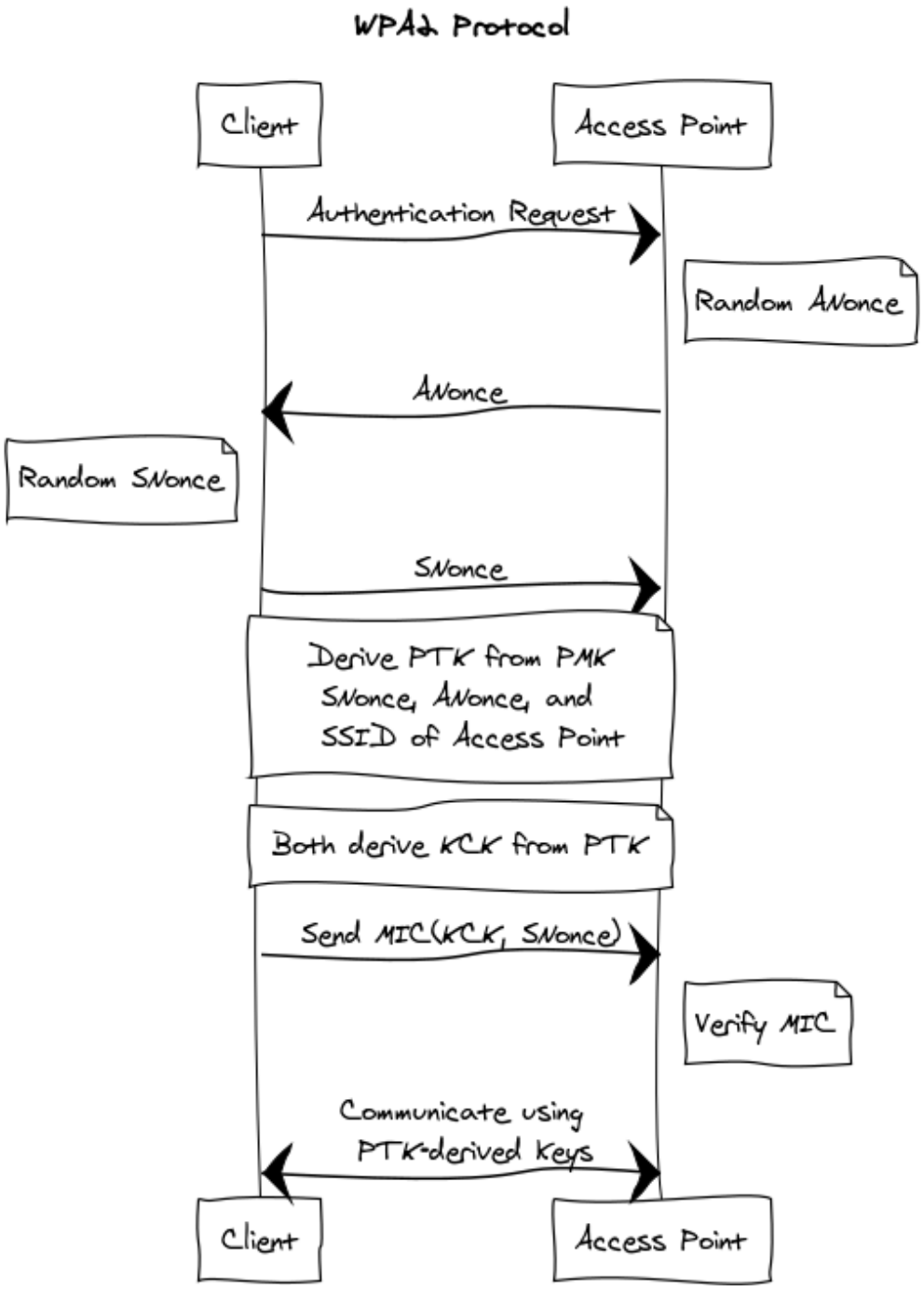
DHCP request, broadcast. Multiple DHCP servers can send out DHCP offers, but the client only accepts one. The broadcast tells the other DHCP servers which offer the client has accepted, and the rest of them can withdraw their offers that were not accepted and return the offered addresses to the pool of available addresses.

DHCP ACK.

Question 3 ***WPA2*** **(20 min)**

Let's review WPA2. You might find some of the definitions below helpful.

- PMK is the *premaster key*, also known as "the WiFi password".
- PTK is the *pairwise transient key*, which is used to derive symmetric keys.
- KCK is the *key confirmation key*, which helps the client and the access point confirm they've agreed on the same keys.



(a) Louis Reasoner proposes that we don't generate ANonce or SNonce, and instead derive the PTK directly from the SSID and PMK. What sort of attack does this fail to prevent?

Solution:

Replay attacks! Nonces always stop replay attacks.

(b) WPA2 has an interesting pattern which is common in cryptographic protocols. Both parties agree on a shared secret, which they use to derive keys. Which other protocol have we seen which follows this motif?

Solution:

TLS–both parties agree on a premastered secret.

(c) Alyssa P. Hacker wants to compromise a WPA2 WiFi network. In order to do so, she performs the handshake many times. She bruteforces possible PMK against the Access Point many times, until the access point eventually accepts it. If the password has 28 bits of entropy[1] and the attacker can make 10 guesses a second, how long will it take to bruteforce the password?

Solution:

$2^{28}/10 \approx 310$ days.

(d) Ben Bitdiddle has an alternate idea. Ben waits until Louis attempts to connect to the network. While this happens, he records all of the messages that Louis sends over the network. How can Ben use this to bruteforce possible PMKs? Why do we expect this to be faster than Alyssa's method?

Solution:

Ben can attempt to bruteforce PMK and derive keys the same way the access point and Louis would. Once he gets the same key confirmation key (which he can check by looking at a MIC computed with the key-confirmation key), then he knows that he's probably generated the same keys and hence has the right PMK.

This is significantly faster than Alyssa's method because it can be computed offline.

[1] As per this XKCD comic, a password which looks like `Tr0ub4dor&3` has roughly 28 bits of entropy.