# Q1

a) In simple essence, Tor anonymize the browsing experience by using multiple relays, called nodes, and layers of encryption between Client and a Server.

b) The onion routing comes from the exchanging the message like an onion (has many layers), hence a message will be encrypted with k1, k2, k3… kn based on n number of relays by the Client. Note that Each node will only know the next and previous nodes they communicated with, not all the nodes in the circuit of communication. The client will establish communication links between each node to exchange his keys (using Diffie-Hellman for example), which is done through a proxy beginning from node 1. Each node will have a key, for example node 1 has key 1, node 2 has key 2, etc. Then, the Client will encrypt his message like an onion, with all the keys he has, and the message will be passed through all the nodes, and every time a node gets a message, it will decrypt one layer until the exit node (last node). As for the replay from the server, the same process happens but it is done using encryption instead of decryption. The nodes remember the node they have to communicate with by using a locally stored Circuit table, which indicates the next destination.

c) Tor uses a special type of domain for its hidden services, .onion, that can only be reached using Tor network. These addresses are not DNS names, but they can be accessed through a proxy using Tor network. Compared to WWW, they make it difficult to trace both the server and the client due previous reason. Hence, it provides mutual anonymity.

d) No. First, as how the internet goes, security issues remain because sites that are reachable through internet or Tor can be attacked. Second, Tor does not encrypt the traffic between exit node and the server. Hence, if somebody could sniff information from the last node (exit node), he could be able to get any personal information sent by the Client. To solve this problem, SSL or TLS should be used between exit node and the server. But how could you trust the exit node? There is no insurance. Furthermore, Human error could lead to breaches when browsing non-https sites, torrenting with Tor, enabling data collecting plugins, download data through Tor, and use unsafe bridge of communication, or providing your real information. Also, anonymity could be compromised too in Tor if an attacker could be able to identify the messages sent between Client and guard node (first node) and the message between exit node (last node) and the server, which the attacker could establish a relationship between them and destroy the idea of anonymity in Tor.

# Q2

1) PIIs from user side are Phone Number, Name, Age, ZIP Code, Phone ID (or serial number), Social Graph, Location Data, and COVID-19 Disease Status (medial status). Shareholders are public health care system hospitals and agencies as they have a role in mitigating the disease, everybody in all nations could get infected by the disease and we want to avoid that (whose data is being collected and they worry about their data), the role of governments to make sure that everybody is obeying the rules and regulations to facilitate the work of public health care system and make sure that Contact Tracing applications also follow the rules, organizations that help in developing the Contact Tracing applications, and finally the society to encourage the people to download and use these Contact Tracing applications. Hence, every individual and entity are considered to be a shareholder in Contact Tracing application system. The main three players that Contact Tracing focus on are the infected people, the people who come into contact with infected people, and everybody else who is not under the previous two cases.

2) Some of the attacks are:

a. Spoofing Identity: a malicious user trying to behave as different person to show himself as that person to the server to gain information on the PIIs.

b. Information disclosure: private or sensitive data that is stored in the phone or the server gets leaked.

c. Linkage Attacks: match different data sets of a certain user to gain more information about that user. For example, malicious actors in the authority could breach the privacy of individuals.

d. Location Disclosure: Malicious actors try to sniff Bluetooth packets sent by Contact Tracing application to record and know the locations of individuals. When a user gets broadcasted by the server, the malicious actor can search their database then query the database using the identifier of the user to get the IDs and map-movements of that individual. Moreover, with background information, it might get worse since he can confirm a user location by only the received information from the other phone (If he knows serial number that corresponds to Iphone X, and he also knows somebody that is close has an Iphone X).

3) Solutions for the attacks:

a. For centralized architecture, authenticate the user's identity that is trying to register. As for decentralized architecture, identify the users that are trying to register with minimal information, or better, do not identify and store the information of users other than needed functionalities.

b. Since privacy and security comes hand to hand, use better encryption to mitigate the risk of information leakage on both ends. Furthermore, as the same solution for a, do not store PII if they are not needed as Google/Apple API implementation states.

c. Data linkage attacks could be prevented using the principals of privacy technologies. The use of Week 4 topics (K-anonymization, l-diversity, t-closeness) or/and differential privacy when storing the data will prevent and mitigate any data linkage attacks.

d. To prevent location disclosure, we should stop the root problem. First, to prevent background information problem, the phone should not send PII such as device serial number to other phones, but instead it sends pseudonym only known to the application developer. This will prevent the sniffer from knowing the phone of the subject. As for the second main problem, we aim to stop the sniffers from constructing a map-like of user’s movements and interactions. The main issue with this problem is the ID being used to identify the packet. If the ID has not been changed for a long time, the adversary could learn a lot of information about user’s location. Hence, a better way of token/ID exchange algorithm should be implemented. First, the timer ID of each user should be minimized. For example, instead of using 30 minutes changing-IDs, use 5 minutes changing-IDs. Second, instead of using fixed timer, we can use an interval timer that will confuses the adversary who is collecting the IDs and constructing the location tree. Thirdly, the IDs should be exchanged anonymously, for example using an anonymous communication, we can prevent the adversary from learning the ID belong to whom by constructing a token exchange model the simulates onion network. That could be done I believe by having 3 nodes, where the middle node will act as the proxy that will let node 1 (phone 1) send his ID securely in a tunnel to node 3 (phone 2), where node 2 can be any other device. Note that the third method is costly and requires the 3 nodes to work, which decreases utility of Contact Tracing Application.