A-1: Motivate why the lecture notes define anonymity roughly as "the IP is unknown"

If the IP-address of a person, who you want to know who it is, is known. You could go to the ISP (Internet service provider) that has provided that IP address. They will know who is the customer of the internet connection with that IP (if they keep logs, or the person is connected). Depending who asks the ISP, the identity of the person will provided. (Different regulations in different countries, the police might get the information, or a warrant might be needed). (In some countries the ISP can be required by law to keep logs of who has what IP).

Grade: 1.0
Motivation:
A good answer that talks about the anonymity issues with IP addresses. A further motivation can be found here for example:
https://www.businessinsider.com/ip-address-what-they-can-reveal-about-you-2015-5?r=US&IR=T

A-2: Why is it important to have a large volume of traffic in anonymous communication?

The statement in the question can be read in two ways: That it's important for a individual to send a lot of data in a communication when you want to be anonymous. This have multiple reasons. If you pad every message you send (before encrypting) an adversary can not tell how long the original message was. The other case is that you continuously send data, that most of the time is just junk data (send as the content of a message, to make it look like a real message), this makes it so that an adversary can't see when you are communication for real.

The other way to read the question is that it's important in general to handle a lot of traffic when wanting to be anonymous. For example: If a mixer receives a to small amount of messages (volume of traffic) it can not mix the messages well. A mixer takes in multiple messages and sends them out in another order, whilst not giving away the senders information. But if there are too few messages, the mixer can only "mix" (scramble order) of so many messages, therefore the anonymity set becomes small. Meaning that if you send a message that an adversary (here assumed global passive adversary) want to determine to whom it is sent, they can look at the the inputs and outputs of the mixer. If there are only a few possible receivers for the message, and conversely only a few senders, there can only be so many people who sent the message.

A active adversary could then interject messages to the mixer to fill up 1 as large part of the mixers communication with (to the adversary) known communication, resulting in just a few or in the worst case just a single communication that was not send by the adversary (assuming mixer at some periodicity sends out all messages received, and don't keep some for later periods) therefore relieving without a doubt linking the sender and receiver.

(The anonymity set is the set of people in which you are anonymous.)

Grade: 0.7
Motivation:
In my opinion, the first “way” that is explained in the first paragraph does not answer the question. When they say “large volume of traffic", they mean many messages in a mix which you explain very well in the second and third paragraph.

A-6: What is the purpose of the random values R1 in a Mix?

R1 is used to break the link between sender and receiver. Without it an adversary could take the message send from the mixer to the receiver and encrypt it with the mixers public key to get the exact message that was sent into the mixer. Instead, when R1 is used the adversary could only do this if they somehow figured out R1 (for example if it was reused instead of generating a new random value every time, or if the random generator was bad).

Grade: 1.0
Motivation:
Clear and good answer. Similar information can be read in the lecture notes (page 5) about anonymity.

A-11: When returning a message using an untraceable return address, why does each Mix encrypt the return message with the randomness Ri?

By encrypting with Ri we ensure that the message into the mix and out from the mix can not be linked by an adversary, may it be Bob who replies to Alice or any other adversary (that does not control all mixes used).

When Alice send the message to Bob she knows what all parts of the path will be, and what the message in, and out will be. But we don’t want this to be possible on the return path. The message is received by a Mix and then encrypted with Ri . And since Alice knows all Ri she used, she can decrypt the message without needing to share any extra information to Bob (except a temporary return address).

In essence, when sending a message the sender puts many layers on top of the message, that are then removed one by one for every mix. When answering the opposite is happening, the message it put in an increasing amount of layers (one for each mix). In order for the message not to be traceable, not even from the return-message sender.

Grade: 1.0
Motivation:
Once again a good answer. I like the layer metaphor. Same information can be retrieved from the lecture notes and slides about anonymity.

A-13: It is straightforward to generalize the N − 1 attack to an N − k attack, 0 < k < N. Describe the N − k attack.

N-1: An adversary could send multiple messages to a Mix in the hope that all but 1 of the messages through the mix during a certain mixing period is by them. By looking at the traffic in and out form the Mix, the adversary can then deduct that the message reaching someone they have not sent a message to must be send by the only other person sending messages through the mix. Resulting in the person sending the message and the person receiving it being linked.

A N-k attack is the same thing except that the adversary only manages to get "N-k" messages through the mix during one mixing period. This does not prove who is communicating with who. But it does reduce the "victims" (for the attack) anonymity set (to k *). In the case where k is 3, any of the three senders could have sent the message to anyone of the 3 receivers severely reducing the anonymity set.

* Assuming the messages periodically is sent of from the mix, and is not with a certain probability delayed to the next batch.

Grade: 1.0
Motivation:
I had the same question and your answer was much better. It was difficult to find information about the n-k attack except for the slides and lecture notes (which did not mention n-k but only n-1), so it is difficult to truly know if the answer is correct. However, the answer provided here sounds very reasonable.

A-14: Consider the long term intersection attack. Explain how the number of users and the sizes of each batch will affect the efficiency of the attack.

The long term intersection attack uses the fact that each time a user of, for example, a forum (or like service) publishes a message with a pseudonym, they must have sent a message to the server. If a adversary with high probability can predict when a user sent in their message to Mix to be forwarded to the server (potentially via more Mixes), they can store the information of everyone sending in a message to the Mix at that time (during that batch), and by storing this information every time a certain user on a forum publishes something. By checking the intersection of the people who where connected to the Mix at those instances, after a certain amount of times only one user will be left in the resulting set, proving that that user is the searched for forum user.

The more messages in each Mix-batch the more people will by chance repeatedly send messages in the same batch as the person that the adversary wants to deanonymize, making it harder (take more attempts) to succeed with a long term intersection attack. If there are many users but few messages in each batch, then the likelihood that two or more users repeatedly are in the same batch is reduced. Therefore more users will increase the probability of success of the attack. (Assuming there are enough users in the first case, since if there only are very few users, the anonymity set is small, and therefore making the attack more likely.).

Grade: 1.0
Motivation:
Good answer. The answer can also be tied together with A-2. The information agrees with the part about long term intersection attack in the lecture notes about anonymity (page 7).

A-16: When negotiating a symmetric key with an onion router, what is the purpose of the H(K1) message sent from the router to Alice?

OR = Onion router

The hash of the key is used to so that Alice can verify that she has calculated the correct key. Since Alice sends her g x (mod p) encrypted with the onion routers public key. She ensures that only that OR can read her it. The onion router then answers without encrypting the answer. This means that any person could send that "answer" without Alice knowing. Except that since Alice receives the g y (mod p) along with the hash of the key, another person that's not the onion router Alice started communicating with could not have make the correct hash of the key, since for building the key g x·y (mod p) the g x (mod p) would have to been known, which only the correct OR knows.

Therefore Alice can be ensured she is communicating with the correct onion router without having a public key available for the router to verify her identity.

Grade: 0.7
Motivation:
While the answer was mainly correct, it was poorly written. The author could have for example said that K1 is the *negotiated* key. The section about negotiating symmetric keys (section 4.2) explains it well.

A-25: Explain what the point of the recognized field in a Tor cell is and how it makes communication more efficient.

OR = Onion router

The point of the recognised field is for a OR to determine whether or not they should send the message forward (if it's a relay node) or interpret it itself to then send it out to the internet (if it's an end node), and to do it in an efficient way.

In the recognised field, in the Tor cell, Alice put two bytes of zeros before she encrypts the cell. When the message is received by an onion router it checks whether those two bytes are zero (as they will be after the last decryption). If it's not zero, the message gets forwarded without calculating the hash of the message. Otherwise the the hash is calculated and compared with the data in the digest field (which will be the same, if the data is decrypted, in other words if the end node (last OR) is reached). If the hash differs, the message is not in the end node, and is therefore forwarded again to the next node.

The reason to use the recognized field instead of checking the hash directly is that it’s costly to calculate the hash for every message routed through a OR. By just checking if there are zeros, these calculations are just needed to be done at the last node, or in 1 out of every 2 16 messages for all other nodes (since the probability of a message having 2 bytes of zeros in the place of recognized field when hashed is 2 −16). Therefore the amount of hashes needed to be calculated are greatly reduced and communication is more efficient.

Grade: 1.0
Motivation:
Very well written answer. Answers both questions perfectly. The answer agrees with the information on page 10 in the lecture notes about anonymity.

Total score: 7.4