Assignment 2

Question 3

1. Keys for each service (Alice, TGS, Bob) and keys between Alice and TGS (K_{AS})
2. Session keys between Alice and Bob
3. **Alice key compromised**- The adversary could learn the session key between Alice and TGS, and then using that key could know the session key between Bob and Alice. So everything shared during that session between Alice and Bob is also known to the passive attacker who could use the keys learnt to decrypt the traffic between Alice and Bob (and also Alice and TGS- from where the adversary got the session key between Alice and Bob). Adversary could learn about all the session keys between Alice and other services (Bob), so basically using all the network traffic, they could learn about everything shared in those sessions.

   Bob key compromised- Using the Ticket sent by Alice, the adversary could know the session key between Alice and Bob and thus can learn the things shared between Alice and Bob during the session. Alice here would represent all the users that had previously connected with Bob.

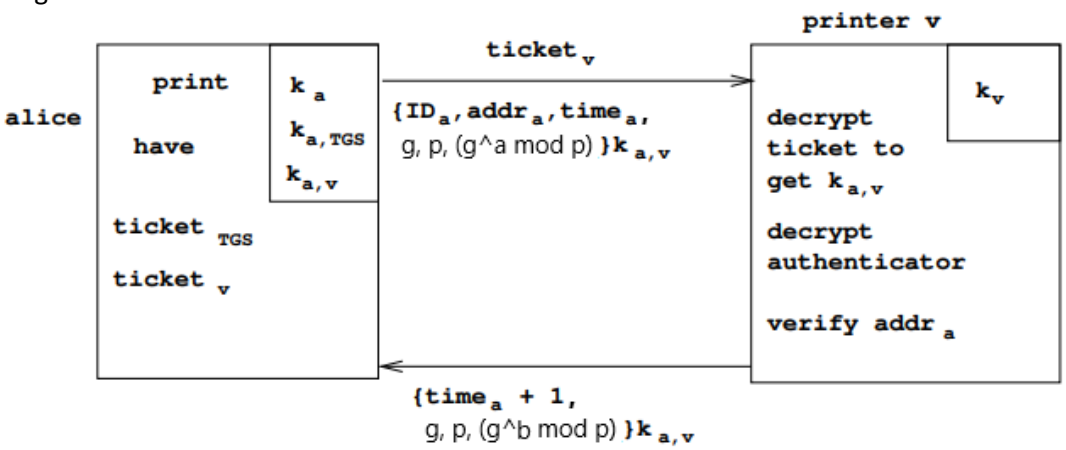
   TGS key compromised- The adversary could learn keys for all the services. Additionally, it could learn the session keys between Alice and TGS, and Alice and Bob. So every single thing shared between Alice and Bob could be learnt, where Alice could be any client and Bob could be any service.

   Key between Alice and TGS compromised- Knowing this key could enable the adversary to learn about the all the session keys between Alice and Bob (where Bob is all the services Alice connected with) until the ticket granting ticket (session key) expires. Further this leads to the adversary learning about all data shared between Alice and Bob(s) during that particular session between Alice and Bob (i.e. until the ticket expires).

   Alice and Bob session key compromised- Knowing this short term key could enable the adversary to learn about the data shared between Alice and Bob during that particular session (i.e. until the ticket expires).

4. Forward secrecy is basically that when the long term key is compromised, no data shared with the short term keys are compromised as a result of long term keys being compromised. Since from the previous question, we see that knowing any long term keys (i.e. Alice/TGS/Bob keys/key between Alice and TGS) results in the ability to learn the short term session keys which enables the adversary to decrypt the data shared during the session between Alice and Bob. Compromise of Alice or Bob results in enabling the adversary to decrypt all the data shared by(or with) Alice or Bob respectively while a compromise of TGS key results in getting all the long term keys (all clients (Alice) and all services (Bob)) being compromised. Since because of any compromised long term keys, the attacker could learn the session key between Alice and Bob, i.e. short term key (as was specified in part 3 of the question), the attacker could use this key to decrypt the data shared during that session. Therefore Kerberos does not have forward secrecy.

5. To add forward secrecy we can use Diffie-Hellman key exchange in Kerberos. So we could just add this concept at the very end, where Alice and Bob(printer) communicate with each other. The modified conversation between Alice and Bob is shown using the following diagram.



After this interaction, both Alice and printer could use the new keys *g^ab mod p* to communicate with each other using and *g^b mod p* or *g^a mod p*. Since the attacker would only know *g*, *p* and either *g^b mod p* or *g^a mod p* or even both, it would be infeasible for the attacker to find *a* and *b*. Hence it would be infeasible to compute *g^ab mod p* for the attacker and thus the data shared between Alice and Bob (printer) is not compromised if any long term keys are compromised.