**Assignment**

**Q2**

1. **Issue with the shared key scheme**

When user *A* and each recipient *B1…,Bn*​ share the same secret *key k*, any of the recipients *B1…,Bn* ​ can forge a valid *MAC* for a message because they have the same *key k* as everyone else. This means that if *B1…,Bn* ​ receives a message with a valid *MAC*, they cannot be sure that it came from *A*, as another user could have generated it using the same key. Thus, making this scheme insecure.

1. **Properties to obey so that part a’s mistake don’t repeat**

To handle this issue, the following properties should be obeyed

* No single user Bi can generate all the MACs required to validate the message for any other user.
* Specifically for each pair of users say Bi and Bj their sets should have some overlap, but neither of them should be able to generate the full set of MACs required to validate each other's messages

This way, a user cannot impersonate A to another user.

1. **When n=10 and m =5 ?**

If I have 5 keys and 10 users, then I can make the pairs by keeping in mind the properties described in part b.

* S1 = [k1 ,k2]
* S2 = [k2 ,k3]
* S3 = [k3 ,k4]
* S4 = [k4 ,k5]
* S5 = [k5 ,k1]
* S6 = [k1 ,k3]
* S7 = [k2 ,k4]
* S8 = [k3 ,k5]
* S9 = [k1 ,k4]
* S10 = [k2 ,k5]

Each recipient has a unique subset of 2 keys, and the overlap property ensures that no user can fully validate messages intended for others.

1. **If two users colluded, then?**

If two users Bi ​ and Bj ​ are allowed to collude, they could potentially share their keys and collectively create valid MACs that correspond to the keys in each other’s sets. For instance, if B1 andB2 are allowed to collude having subset S1 and S2 respectively from part c then,

S1 = [k1 ,k2] [k1, k2, k3] S6 =[k2, k3]



S2 = [k2 ,k3]



Using this combination of keys S6  keys are now available to the S1 and S2 . This would enable them to forge messages as if they came from A, thus breaking the integrity of the scheme.

The subsets Si were created to prevent individual users from being able to forge MACs. However, if collusion occurs, this protection is rendered ineffective. As a result, the scheme becomes entirely insecure in the presence of collusion.