CS5285: Information Security for eCommerce Spring, 2024
Tutorial 8, Week 10 (November 12)

Questions:

1. **Symmetric key management using asymmetric crypto**
   i) How would you use asymmetric cryptographic to do key transport between only two parties.
   ii) How would you use asymmetric cryptographic to do key agreement between only two parties.

2. **PGP**
   PGP uses the "Anarchy Model" as the trust model. Users decide themselves which keys to trust by certifying them, i.e., signing them with their own key. PGP knows several trust levels for keys of other users: Trusted, marginally trusted and untrusted.

   The idea is, that if Alice does not know Bob's public key, she checks whether any of her friends have signed Bob's key. If either X of her friends who are *trusted* or Y of her friends who are *marginally trusted* have signed Bob's key, then Alice will also trust Bob's key. Assume that we require $X = 2$ trusted signatures or $Y = 3$ marginally trusted signatures to trust the key with confidence > 1. Consider whether in the following cases Alice will trust Bob's key:

   (a) Alice has 2 *trusted* friends who have signed Bob's key.
   (b) Alice has 2 *marginally trusted* friends who have signed Bob's key.
   (c) Alice has 1 *trusted* friend and 1 *marginally trusted* friend who signed Bob's key.
   (d) Alice has 1 *trusted* frend and 2 *marginally trusted* friends who signed Bob's key.

3. **Digital Certificates**

   (a) A certificate does not necessarily have to be signed directly by a CA you trust, there is something called a *certificate chain*. Can you imagine what a certificate chain is that allows you to trust a certificate?
   (b) Who signs the certificate of a CA?
   (c) Considering the default list of trusted certificates built into every browser, explain how one malicious CA could compromise *every* secure connection made by a particular user.

4. **PKI for Payment**
   In the lecture we discussed how good key management solve practical problems in open and closed payment systems. What would such a protocol look like? Design a protocol for allowing a merchant and a card to conduct on offline payment transaction (without help of a online third party). Sometimes a card also might want online verification before approving transaction, how would a card talk to its issuer?