Day 2

1. Encryption a. monoalphabetic ciphers c. Symmetric Encryption - caesar cipher (the key is how many letters to rotate it)
   - need secure algorithm, they can’t decipher ciphertext or key even if they have some examples of ciphertext along with decrypted version
   - Keys need to be distributed in secure manner
   - cryptanalysis
     - they know something (either plaintext, or algorithm to deduce the key)
   - brute force
     - try every possible combination to guess the key d. Stream Ciphers

2. Hash functions: a. MD5 b. sha1sum c. For message authentication. Encryption protects against passive attacks. Hash is used for active attacks (falsification of data and transactions). (Still falls under data integrity)

3. PKI
   - discuss PKI
     - Proposed in 1976 (diffie-hellman)
     - two separate keys
     - 6 ingredients to PKI
       - Plaintext
       - Encryption Algorithm
       - Public and private key
         - Each user generates a pair, public key is publicly available
         - encrypt message using persons public key, only corresponding private key can decrypt
         - private keys are never distributed
         - can ensure a person is who they say they are
         - when sending messages we can ensure confidentiality
         - when receiving messages we can ensure authentication and/or data integrity
       - Ciphertext
       - Decryption algorithm
   - look at /etc/moduli
   - diffie-hellman key exchange process
     - enables 2 users to securely reach agreement about shared secret that can be used as a secret key for symmetric encryption of messages
   - Asymmetric encryption algorithms
     - RSA = block cipher
     - currently uses 1024 bit key

4. Digital Signatures
   - bob creates message, generates hash value for the message, and encrypts hash code with private key, creating a digital signature
   - alice receives messages plus signature
     - recalculates hash value for message
     - decrypts signature using bobs public key
     - compares calculated hash value to decrypted hash value
     - the message is safe from alteration, but not from observation

5. Certificates
   - downside: some user could send their public key, purporting to be Bob.
   - solution is public key certificate
     - consists of public key, userid, plus signed by trusted 3rd party (ie verisign)