CAS CS 357

Discussion worksheet 3

1. Suppose that m > 2 users want to communicate securely and confidentially. Suppose further that each of the m users wants to be able to communicate with every other user without the remaining m − 2 users being able to listen on their conversation. How many distinct keys are needed if we are using:
    a. A symmetric key cryptosystem, where two users use a shared secret key to communicate?

       Two people need the same shared key to communicate with each other.

       Ex) Alice shares Bob with key1, Bob shares Charlie with key2

       Therefore, we need to find the number of pairs since each pair requires a distinct key

       Answer: mC2 → m!/((2!)*(m-2!) = m*(m-1)/2
    b. A public key cryptosystem, where every user has a public key, KE and a private (secret) key, KD. How many keys are needed for each type of cryptosystems if m = 1000? If m = 10^5?

       Answer: 2m

       Alice has his own secret key, Bob has his own secret key.

       Each person has his own secret key and public key → two keys per person.

2. Anyone can submit articles to an online blog, even if those articles are utter nonsense. Each day at 7am, the blog owner signs and posts newly-submitted articles to the blog as follows:

   If articles $x_1, x_2, \ldots, x_n$ are posted on the blog, the owner creates manifest m and signature o as

   $$m = \{H(x_1), H(x_2), \ldots, H(x_n), t\} \qquad o = Sig_{sk}(m)$$

   H() is a hash function. (SK, PK) is the owner's (secret key, public key) pair. t is the time when the owner plans to update the blog next

   a. Users accessing the blog know PK, and download $x_1, x_2, \ldots, x_n$ and m and σ. How can users verify that the current version of the blog consists of articles $x_1, x_2, \ldots, x_n$?

      - Step 1: Compute hash of all x_i

        Check if $m = (h_1 = H(x_1), h_2 = H(x_2), \ldots, h_n = H(x_n))$

      - Step 2: verify if m and o (use the public key to verify)

   b. An attacker wants to trick the user into thinking that the blog currently contains some article that the owner did not post.

      This attacker can submit articles to the blog. This attacker can also man-in-middle communication between a user and the blog.

      - What property do we require from H to ensure this attack fails?

        Only thing we can change is the x values (article). However, the articles need to change in the way that the hash of it is the same, since messages will be different with different hash values.

Therefore, we need the property of collision resistance. H(x) given on the website does not equal H(x') that the adversary finds.

- Suppose H does Not have this property. Write pseudocode for the attack

  Find a x' such that for any x_i in blog $H(x_i) = H(x')$

3. Let (Sig_0, Ver_0) and (Sig_1, Ver_1) be two digital signature schemes with the same message space. Suppose you know that only one of the two schemes is CMA secure, but you do not know which one.

   Show how construct a new digital signature scheme (Sig, Ver) that is guaranteed to be CMA secure. (That is, present the key generation algorithm, the signing algorithm, and the verification algorithm for your new digital signature scheme.)

   keygen() = $keygen_0() \rightarrow (PK_0, SK_0)$

   $keygen_1() \rightarrow (PK_1, SK_1) \rightarrow (PK_0, PK_1), (SK_0, SK_1)$

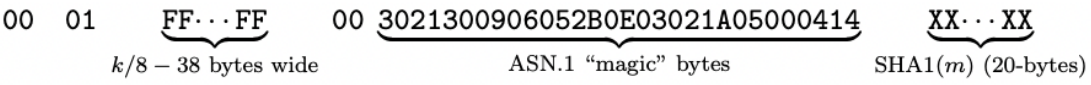
   Sig (m): $(sig_{0sk0}(m), Sig_{1sk1}(m)) = o$

   Ver(m): $(ver_{0pk0}(m), ver_{1pk1}(m))$

   Make a combination of both schemes. Even though one is not secure, one is secure so the adversary cannot forge the rest of the secure part. Therefore, the adversary might forge half of the new scheme, but not the rest of the secure part of the new scheme.

4. On February 23 2017, researchers announced that they found a collision in SHA1. The collision was two files f_1 and f_2 such that SHA1(f_1) = SHA1(f_2). See shattered.io.

Consider PKCS #1 v1.5 RSA digital signatures. To sign a message m, the message is hashed and padded as shown below to obtain the padded value p(m):



Then, the signature is

$$p(m)^d \mod N$$

where N is the RSA modulus, d is the secret RSA decryption exponent, and e is the public encryption exponent. Thus, the public key is (e, N) and the secret key is (d, N).

Present an attack that proves that PKCS #1 v1.5 RSA is not a secure digital signatures when SHA1 is used as the hash function. You must use the two files f_1 and f_2 in your attack.

SHA1 is the hash function.

Find f_1 and f_2 such that

Adversary needs to find SHA1(f_1) = SHA1(f_2) since if the adversary finds it, $RSA_{sk}(f_1) = RSA_{sk}(f_2)$