Ryan Stillings

Cryptography and Network Security I

HW2 Theory Part 2

1. Users A and B use the Diffie-Hellman key exchange technique with a common prime q = 71 and a primitive root α = 7.
   a) If user A has a private key $X_A = 5$, what is A's public key Y_A?
      7^5 mod 71 = **51**
   b) If user B has a private key $X_B = 12$, what is B's public key Y_B?
      7^{12} mod 71 = **4**
   c) What is the shared secret key?
      4^5 mod 71 = 51^{12} mod 71 = **30**
   d) In the Diffie-Hellman protocol, each participant selects a secret number x and sends the other participant (α^x mod q) for some public number α. What would happen if the participants sent each other (x^α mod q) instead?
      If the participants sent each other x^α mod q they would not end up with the same shared secret same. In normal DH, the shared secret is $(a^{x_a})^{x_b} = (a^{x_b})^{x_a}$, where each operation results in the same value. But with this modified DH the shared secret would be $({x_a}^a)^{x_b} = ({x_b}^a)^{x_a}$. Normal DH turns out to be equivalent since applying multiple exponents is a commutative operation. Modified DH, on the other hand, changes the number to which the exponent is being applied, which is not a commutative operation.

2. A network resource X is prepared to sign a message by appending the appropriate 64- bit hash code and encrypting that hash code with X's private key as described in class (also in the textbook, Page 330.
   a) Describe the Birthday Attack where an attacker receives a valid signature for his fraudulent message?
      As explained in the book, the birthday attack can be summarized as follows. First, the attacker generates $2^{m/2}$ variations of an original, valid message, with each of the variations having essentially the same meaning but a different hash. Next, the attacker does the same with their own malicious message, storing the hash of $2^{m/2}$ variations of their message. After that, the attacker compares the hashes of the original message variations with the hashes of his own message variations until he finds a match. Once a match is found, the attacker sends the matching original message variation out for verification. Once the attacker receives a signature, he sends out the malicious message variation with the matching hash code, attaching that signature to the malicious message instead.
   b) How much memory space does attacker need for an M-bit message?
      $2^{64/2}$ = 4,294,967,296 variations * 64 bits / message = 274,877,906,944 bits of memory
   c) Assuming that attacker's computer can process 2^{20} hash/second, how long does it take at average to find pair of messages that have the same hash?
      4294967296 hashes * 2 versions (original, malicious) / 2^{20} hash/second = 8192 seconds
   d) Answer (b) and (c) when 128-bit hash is used instead.

$2^{128/2}$ variations * 128 bits / message = $2.36*10^{21}$ bits ($2.36*10^{8}$ terabytes) of memory
2^{64} hashes * 2 versions / 2^{20} hash/sec = 35,184,372,088,832 seconds (1,115,689 years)

3. Use Trapdoor Oneway Function with following secrets as described in lecture notes to encrypt plaintext P = ’0101 0111’. Decrypt the resulting ciphertext to obtain the plaintext P back. Show each step to get full credit.

   S = {5, 9, 21, 45, 103, 215, 450, 946}
   a = 1019, p = 1999

   Encryption:

   Add up $a*s_i$ mod p ∀ s_i in s where s_i == 1

   (1019 * 9 mod 1999) + (1019 * 45 mod 1999) + (1019 * 215 mod 1999) + (1019 * 450 mod 1999) + (1019 * 946 mod 1999) = 5481 mod 1999 = **1483**

   Decryption:

   $c_0 = a^{-1} * 1483 \bmod 1999 = 1665$

i	s_i	c_i	$c_i - s_i$
8	946	1665	719
7	450	719	269
6	215	269	54
5	103	54	NA
4	45	54	9
3	21	9	NA
2	9	9	0
1	5	0	NA

Result = [0,1,0,1,0,1,1,1] ✓