Crypto HW2b

Milena Gonzalez

September 2018

Q1.

Users A and B use the Diffie-Hellman key exchange technique with a common prime q = 71 and a primitive root $\alpha = 7$.
a) If user A has a private key $X_a = 5$, what is A's public key Y_a?
$Y_a = \alpha^{X_a} \text{mod q} = 7^5 \text{mod } 71 = 51$

b) If user B has a private key $X_b = 12$, what is B's public key Y_b?
$Y_b = \alpha^{X_b} \text{mod q} = 7^{12} \text{mod } 71 = 4$

c) What is the shared secret key?
K = $(\alpha^{X_a})^{X_b} \text{mod q} = 7^{5*12} \text{mod } 71 = 30$

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?
The system would be insecure and can be broken easily. Since α is a public number, Eve will also know it and can easily get the key value by dividing the message by α

Q2.

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?
- type of brute force attack
- exploits mathematics behind the birthday parado and probability theory
- success depends on the likelihood of collisions found between random attack attempts and a fixed degree of permutations (pigeonhole theorem)

b) How much memory space does attacker need for an M-bit message?
Attacker generates $2^{m/2}$ variations of a valid message all with essentially the same meaning, but since it is a 64 bit message, will need $2^{m/2} * 2^6$ 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?
$\frac{2^{m/2} * 2^6}{2^{20}}$seconds

d) Answer (b) and (c) when 128-bit hash is used instead.
$2^{m/2} * 2^7$, $\frac{2^{m/2} * 2^7}{2^{20}}$seconds

Q3.

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

public key β = S*(a mod p)
β = S*{1097, 1175, 1409, 1877, 1009, 1194, 779, 456}

Ciphertext C = (0)(1097) + (1)(1175) + (0)(1409) + (1)(1877) + (0)(1009) + (1)(1194) + (1)(779) + (1)(456) = 5481

a^{-1}(mod p)

1999 = (1019)(1) + 980	980 = 1999 + 1019(-1)
1019 = 980(1) + 39	39 = 1019 + 910(-1)
980 = 39(25) + 5	5
39 = 5(7) + 4	4 = 39 + 5(-7)
5 = 4(1) + 1	1 = 5 + 4(1)

1 = 5 + 4(-1)
1 = 5 + (-1)[39+5(-7)]
1 = 39(-1) + 5(8)
1 = 39(-1) + 8[980 + 39(-25)] 1 = 980(8) + 39(-201)
1 = 980(8) + (-201)[1019+950(-1)]
1 = 1019(-201) + 980(209)
1 = 1019(-201) + (209)[1999 + 1019(-1)]
1 = 1019(-410) + 1999(0)
1999 - 410 = 1589 ($< -a^{-1}$)

C*a^{-1} mod p = 1665
Decompose:
1665 - 946 = 719
719 - 450 = 269
269 - 215 = 52
54 - 45 = 9
9 - 9 = 0