Arnel Adviento - 10130641
Introduction to Cryptography CPSC 418 Fall 2016
Department of Computer Science
University of Calgary

November 18, 2016

HOME WORK #3

Problem	Marks
1	
2	
3	
4	
5	
6	
7	
Total	

Problem 1. (Flawed hash function based MAC designs, 28 marks)

(a) i. We know

$$\begin{aligned} \boldsymbol{M_1} \\ \boldsymbol{MAC_1} &= \boldsymbol{H(K||M_1')} \\ \boldsymbol{M_2} &= \boldsymbol{M_1'||X} \end{aligned}$$

We want to get

$$\begin{aligned} \boldsymbol{MAC_2} &= \boldsymbol{H(K||M_2)} \\ &= \boldsymbol{H(K||M_1'||X)} \end{aligned}$$

without knowing $\boldsymbol{K}$

$$\begin{aligned} \boldsymbol{MAC_2} &= \boldsymbol{f(H(K||M_1')||X)} \\ &= \boldsymbol{f(MAC_1||X)} \end{aligned}$$

By doing this we showed we can compute the PHMAC of $\boldsymbol{M_2}$ without know $\boldsymbol{K}$ therefore defeating computation resistance.

ii. Since we know $\boldsymbol{H}$ is not weakly collision resistance. There exists:

$$\begin{aligned} \boldsymbol{M_1} &\neq \boldsymbol{M_2} \\ \boldsymbol{MAC_1} &= \boldsymbol{MAC_2} \end{aligned}$$

To generate a pair $\boldsymbol{(M_2, MAC_2)}$ first we just pick an arbitrary message $\boldsymbol{M_2}$.
We know that $\boldsymbol{MAC_2 = H(M_2'||K)}$ even if we don't know $\boldsymbol{K}$, since it is not weakly collision resistant, we can find X which you append at the end of $\boldsymbol{M_2}$ which will generate $\boldsymbol{MAC_2}$

(b) i. we know

$$\begin{aligned} \boldsymbol{M_1} \\ \boldsymbol{MAC_1} &= \boldsymbol{E_K(M_1)} \\ \boldsymbol{M_2} &= \boldsymbol{MAC_1} \\ \boldsymbol{MAC_2} &= \boldsymbol{E_K(M_2)} \\ &= \boldsymbol{E_K(MAC_1)} \end{aligned}$$

we want to know $\boldsymbol{MAC_3}$ given a two-block message $\boldsymbol{M_3 = M_1||0^n}$

$$\begin{aligned} \boldsymbol{MAC_3} &= \boldsymbol{E_K(M_3)} \\ \boldsymbol{C_1} &= \boldsymbol{E_K(M_1)} \\ \boldsymbol{C_2} &= \boldsymbol{E_K(0^n \oplus E_K(M_1))} \\ &= \boldsymbol{E_K(E_K(M_1))} \\ &= \boldsymbol{E_K(MAC_1)} \\ \boldsymbol{MAC_3} &= \boldsymbol{MAC_2} \end{aligned}$$

$$\begin{aligned} \boldsymbol{M_3} &\neq \boldsymbol{M_2} \\ \boldsymbol{MAC_3} &= \boldsymbol{MAC_2} \end{aligned}$$

This violates computation resistance since it is not weakly collision resistance since there exists $\boldsymbol{MAC_3 = MAC_2}$ but $\boldsymbol{M_3 \neq M_2}$.

ii. we know

$$\begin{aligned} \boldsymbol{M_1} & \\ \boldsymbol{MAC_1} &= \boldsymbol{E_K(M_1)} \\ \boldsymbol{M_2} & \\ \boldsymbol{MAC_2} &= \boldsymbol{E_K(M_2)} \\ \boldsymbol{M_3} &= \boldsymbol{M_1||X} \\ \boldsymbol{MAC_3} &= \boldsymbol{E_K(M_3)} \\ \boldsymbol{C_1} &= \boldsymbol{E_K(M_1)} \\ \boldsymbol{C_2} &= \boldsymbol{E_K(X \oplus (E_K(M_1)))} \\ \boldsymbol{MAC_3} &= \boldsymbol{E_K(X \oplus MAC_1)} \end{aligned}$$

we want to know $\boldsymbol{MAC_4}$ given a two-block message
$\boldsymbol{M_4 = M_2||(MAC_1 \oplus X \oplus MAC_2)}$

$$\begin{aligned} \boldsymbol{MAC_4} &= \boldsymbol{E_K(M_4)} \\ &= \boldsymbol{E_K(M_2||(MAC_1 \oplus X \oplus MAC_2)} \\ \boldsymbol{C_1} &= \boldsymbol{E_K(M_2)} \\ \boldsymbol{C_2} &= \boldsymbol{E_K(MAC_1 \oplus X \oplus MAC_2 \oplus E_K(M_2))} \\ &= \boldsymbol{E_K(MAC_1 \oplus X \oplus MAC_2 \oplus MAC_2)} \\ &= \boldsymbol{E_K(MAC_1 \oplus X)} \end{aligned}$$

$$\begin{aligned} \boldsymbol{M_4} &\neq \boldsymbol{M_3} \\ \boldsymbol{MAC_4} &= \boldsymbol{MAC_3} \end{aligned}$$

This violates computation resistance since it is not weakly collision resistance since there exists $\boldsymbol{MAC_4 = MAC_3}$ but $\boldsymbol{M_4 \neq M_3}$.

$\longrightarrow$ Answer

Problem 2. (A modified man-in-the-middle attack on Die-Hellman, 12 marks)

(a) Bob receives $(g^a)^q$ from Mallory and computes $((g^a)^q)^b$
Alice receives $(g^b)^q$ from Mallory and computes $((g^b)^q)^a$

$((g^a)^q)^b \mod p = g^{aqb} \mod p = q^{bqa} \mod p = ((g^b)^q)^a \mod p$
based on exponent power rules

(b) We know that g is a primitive root of P and its smallest positive exponent is k with $g^k \equiv 1$ (mod p) where $k = (p-1)$. We also know the equation $p = mq + 1$

So when Mallory raises g to the power of q this happens.

$$\begin{aligned} g^q &= g^{(p-1)/m} (\mathbf{mod} p) \\ &= g^{k/m} (\mathbf{mod} p) \end{aligned}$$

So we know that if we have g has a key-space of k and when we raise a power to g for example g^2 we half the key-space, essentially getting $k/2$. So when we raise g to the power of q we get $g^q = g^{k/m}$, which mean we divide our key-space by (k/m) which is $\frac{k}{(k/m)} = k \times \frac{m}{k} = m$ and we get a a new key-space of sized m.

(c) The man-in-the-middle attack discussed in class generates two different keys where Bob computes the key $(g^e)^b$ and Alice computes $(g^e)^a$. If Alice and Bob were to exchange the keys they computed they would notice that they would be different and will know that their keys have been tampered with. In this variation, we showed in part (a) that both Alice and Bob computed the same key and if they were to exchange those keys they would not notice the attack.

$\longrightarrow$ Answer

Problem 3. (Binary exponentiation, 12 marks)

(a) Base Case (i = 0):

$$\begin{aligned} s_i &= \sum_{j=0}^{i} b_j 2^{i-j} \\ s_0 &= \sum_{j=0}^{0} b_j 2^{0-j} \\ &= b_0 2^{0-0} \\ &= b_0 \cdot 1 \\ s_0 &= b_0 \end{aligned}$$

Inductive Hypothesis:
Suppose $s_i = \sum_{j=0}^{i} b_j 2^{i-j}$ we want to prove $s_{i+1} = 2s_i + b_{i+1}$

$$\begin{aligned} s_{i+1} &= 2s_i + b_{i+1} \\ &= 2(\sum_{j=0}^{i} b_j 2^{i-j}) + b_{i+1} \\ &= \sum_{j=0}^{i} b_j 2^{i-j+1} + b_{i+1} \\ &= (b_0 2^{i+1} + b_1 2^1 + b_2 2^{i-1} \dots b_j 2^1) + b_{i+1} \\ &= (b_0 2^{i+1} + b_1 2^1 + b_2 2^{i-1} \dots b_j 2^1) + b_{i+1} 2^0 \\ &= \sum_{j=0}^{(i+1)} b_j 2^{(i+1)-j} \end{aligned}$$

(b) Base Case (i=0)

$$\begin{aligned} r_i &\equiv a^{s_i} (\text{mod m}) \\ r_0 &\equiv a^{s_0} (\text{mod m}) \\ &\equiv a^{b_0} (\text{mod m}) \\ &\equiv a^1 (\text{mod m}) \end{aligned}$$

Inductive Hypothesis:
Suppose $r_i \equiv a^{s_i} (\text{mod m})$ we want to prove $r_{i+1} = r_i^2 (\text{mod m})$ and $r_{i+1} = r_i^2 a (\text{mod m})$

Case: $b_{i+1} = 0$

$$\begin{aligned} r_{i+1} &\equiv r_i^2 (\text{mod m}) \\ &\equiv (a^{s_i})^2 (\text{mod m}) \\ &\equiv (a^{2s_i}) (\text{mod m}) \end{aligned}$$

Case: $b_{i+1} = 1$

$$\begin{aligned} r_{i+1} &\equiv r_i^2 a (\text{mod m}) \\ &\equiv (a^{s_i})^2 a (\text{mod m}) \\ &\equiv a^{(2s_i)} a (\text{mod m}) \\ &\equiv a^{(2s_i+1)} (\text{mod m}) \end{aligned}$$

(c) Since in part b) we proved that $r_i \equiv a^{s_i} (\mathbf{mod} m)$ for $0 \leq i \leq k$ and in part a) we proved s_i hold since we proved s_{i+1} hold as well. Therefore $r_i \equiv a^{s_i} (\mathbf{mod} m)$ does compute $a^n (\mathbf{mod} m)$ where the answer is $r_k (\mathbf{mod} m)$.

$\longrightarrow$ Answer

Problem 4. (An RSA toy example for practicing binary exponentiation, 12 marks)
Consider the RSA encryption scheme with public key (e, n) = (11, 77).

(a)

$$C \equiv M^e \pmod{n}$$
$$C \equiv 17^{11} \pmod{77}$$

$$17^1 = 17 \mod 77$$
$$17^2 = 58 \mod 77$$
$$17^4 = (17^2)^2 = 58^2 = 53 \mod 77$$
$$17^8 = (17^4)^2 = 53^2 = 37 \mod 77$$

$$\begin{aligned} 17^{11} &= 17^8 \cdot 17^2 \cdot 17^1 \\ &= 37 \cdot 58 \cdot 17 \\ &= 37 \cdot 58 \cdot 17 \\ &= 67 \cdot 17 \\ &= 61 \end{aligned}$$

(b)

$$\begin{aligned} \boldsymbol{pq} &= \boldsymbol{n} \\ \boldsymbol{pq} &= \mathbf{77} \\ \mathbf{11} \times \mathbf{7} &= \mathbf{77} \end{aligned}$$

$$\begin{aligned} \boldsymbol{de} &\equiv \mathbf{1} (\text{mod } \boldsymbol{\phi n}) \\ \boldsymbol{\phi n} &= (\boldsymbol{p} - \mathbf{1})(\boldsymbol{q} - \mathbf{1}) \\ &= \mathbf{10} \times \mathbf{6} \\ &= \mathbf{60} \end{aligned}$$

Extended Euclidean

$$\begin{aligned} \mathbf{60} &= \mathbf{5} \cdot \mathbf{11} + \mathbf{5} \\ \mathbf{11} &= \mathbf{2} \cdot \mathbf{5} + \mathbf{1} \end{aligned}$$

$$\begin{aligned} \mathbf{1} &= \mathbf{11} - \mathbf{2} \cdot \mathbf{5} \\ &= \mathbf{11} - \mathbf{2}(\mathbf{60} \times (\mathbf{5} \cdot \mathbf{11})) \\ &= \mathbf{11} - (\mathbf{2} \cdot \mathbf{60}) + (\mathbf{10} \cdot \mathbf{11}) \\ &= \mathbf{11}(\mathbf{11}) - (\mathbf{2})(\mathbf{60}) \\ \boldsymbol{d} &= \mathbf{11} \end{aligned}$$

$$\begin{aligned} \boldsymbol{de} &\equiv \mathbf{1} (\text{mod } \boldsymbol{\phi n}) \\ \mathbf{11} \cdot \mathbf{11} &\equiv \mathbf{1} (\text{mod } 60) \end{aligned}$$

(c)

$$\begin{aligned} \boldsymbol{M} &\equiv \boldsymbol{C^d} \text{ (mod n)} \\ &\equiv \mathbf{32^{11}} (\text{mod } 77) \\ \text{binary of } 11 &= \mathbf{1011} \\ \boldsymbol{r_0} &= \mathbf{32} \text{ mod } 77 \\ \boldsymbol{r_1} &= \mathbf{32^2} \text{ mod } 77 = \mathbf{23} \text{ mod } 77 \\ \boldsymbol{r_2} &= \mathbf{23^2} \cdot \mathbf{32} \text{ mod } 77 = \mathbf{65} \text{ mod } 77 \\ \boldsymbol{r_3} &= \mathbf{65^2} \cdot \mathbf{32} \text{ mod } 77 = \mathbf{65} \text{ mod } 77 \\ \boldsymbol{M} &= \mathbf{65} \end{aligned}$$

$\longrightarrow$ Answer

Submitted by Arnel Adviento - 10130641 on November 18, 2016.