CPSC 418 / MATH 318 — Introduction to Cryptography
ASSIGNMENT 3

Name: Karmvir Singh Dhaliwal
Student ID: 30025474

Problem 1 — Flawed MAC designs (11 marks)

a. The attacker knows what *M1* and PHMAC($M1$) are. From this, the attacker simply needs to compute *f(*PHMAC($M1$)*, x)*. From the *ITHASH* algorithm, the attacker knows that the the first $L+1$ steps of computing PHMAC($M2$) will be exactly PHMAC($M1$), because the algorithm performs the function f on each block, and the first $L+1$ blocks, namely *K*-P_L, are in the exact same order. The last step of calculating PHMAC($M2$) is calculating *f(*PHMAC($M1$)*, x)*, as this is the only block remaining in *M2* that has not been used in function f yet. Since K is the first block used in the *ITHASH* algorithm, the attacker does not need to know what K is if PHMAC($M1$) is known, as K is in this calculation. After calculating *f(*PHMAC($M1$)*, x)*, the attacker now knows *M2* and PHMAC($M2$) without any knowledge of K, thereby defeating computation resistance for PHMAC.

b. Since *ITHASH* is not weak collision resistant, the attacker can find a message *M2* where *ITHASH(M1)* = ITHASH(M2) and *M1* $\neq$ *M2*. Now, since K is the same, and the attacker knows that AHMAC($M1$) $= f(ITHASH(M1), K)$, they can deduce that:

$$\text{AHMAC}(M1) = f(ITHASH(M1), K) = f(ITHASH(M2), K) = \text{AHMAC}(M2) \qquad (1)$$

And therefore, since AHMAC($M1$) is known, they have found a message pair and MAC pair (M2, AHMAC($M1$)) without knowing what K is.

Problem 2 — Fast RSA decryption using Chinese remaindering (7 marks)

In normal RSA, to decrypt we calculate:

$$\begin{gathered} C \equiv M^e \mod n \\ C^d \mod n \equiv M^{ed} \mod n \equiv M \mod n \end{gathered} \tag{2}$$

However, in this method we calculate:

$$\begin{aligned} & pxM_q + qyM_p \mod n \\ & \equiv (pxC^{d_q} \mod q + qyC^{d_p} \mod p) \mod n \\ & \equiv (pxM^{ed \mod q-1} \mod q + qyM^{ed \mod p-1} \mod p) \mod n \\ & \equiv (pxM^{ed} \mod n + qyM^{ed} \mod n) \mod n \\ & \equiv pxM^{ed} + qyM^{ed} \mod n \\ & \equiv (px + qy)M^{ed} \mod n \\ & \equiv M^{ed} \mod n \\ & \equiv M \mod n \end{aligned} \tag{3}$$

As required, thereby proving that this method does in fact decrypt messages correctly. □

Problem 3 — RSA primes too close together (21 marks)

a. To prove this, we begin by noting that n = pq, where p and q are prime. This means that the factors of n are simply 1, p, q and n. In pair form, we have the pairs (1,n) and (p,q). Now, we know $n = x^2 - y^2$, which can be factored into $n = (x - y)(x + y)$, by the difference of squares factoring rule. Now:

$$\begin{aligned} n &= (x-y)(x+y) \\ 1 \times n &= (x-y)(x+y) \end{aligned} \tag{4}$$

Since we know that n > 1, it logically follows that 1 = (x-y) and n = (x+y), because x and y are both positive integers and adding y to x will clearly be bigger than subtracting y from x. So:

$$\begin{aligned}
&1 = x - y \quad \text{-From above} \\
&y = x - 1 \quad \text{-Solving for y} \\
\\
&n = x + y \quad \text{-From above} \\
&n = x + x - 1 \quad \text{-Subbing in y} \\
&n = 2x - 1 \quad \text{-Collecting like terms} \\
&x = \frac{n+1}{2} \quad \text{-Solving for x} \\
\\
&y = x - 1 \\
&y = \frac{n+1}{2} - 1 \quad \text{-Subbing in x} \\
&y = \frac{n+1}{2} - \frac{2}{2} \quad \text{-Getting common denominator} \\
&y = \frac{n+1-2}{2} \\
&y = \frac{n-1}{2}
\end{aligned} \tag{5}$$

As required. Now we do the same for n = pq. Again, we note that p>q, and therefore p must be (x+y), for the same reasons as mentioned above. Now:

$$\begin{aligned}
&q = x - y \quad \text{-From above} \\
&y = x - q \quad \text{-Solving for y} \\
\\
&p = x + y \quad \text{-From above} \\
&p = x + x - q \quad \text{-Subbing in y} \\
&p = 2x - q \quad \text{-Collecting like terms} \\
&x = \frac{p+q}{2} \quad \text{-Solving for x} \\
\\
&y = x - q \\
&y = \frac{p+q}{2} - q \quad \text{-Subbing in x} \\
&y = \frac{p+q}{2} - \frac{2q}{2} \quad \text{-Getting common denominator} \\
&y = \frac{p+q-2q}{2} \\
&y = \frac{p-q}{2}
\end{aligned} \tag{6}$$

Again, as required. Therefore we have proven that if x, y are integers with $x > y > 0$ and $n = x^2y^2$, then

$$\begin{aligned}
x = \frac{p+q}{2} &\text{ and } y = \frac{p-q}{2} \\
\text{or } x = \frac{n+1}{2} &\text{ and } y = \frac{n-1}{2} \ \square
\end{aligned} \tag{7}$$

b. We want to prove that n+1 > p+q. Now observe:

$$\begin{aligned}
&n+1 \\
&= pq + 1 \\
&> pq \\
&> 2p \quad \text{-Since q is an odd prime} \\
&= p + p \\
&> p + q \quad \text{-Since p > q} \\
n+1 &> p + q
\end{aligned} \tag{8}$$

As required. Therefore we have proven that n+1 > p + q, using the fact that p > q. □

c. To prove that $\sqrt{n} < \frac{p+q}{2} < p$, we will first prove that $p > \frac{p+q}{2}$. Now, observe:

$$\begin{aligned} p& \\ &= \frac{2p}{2} \\ &= \frac{p+p}{2} \\ &> \frac{p+q}{2} \quad \text{-Since p > q.} \\ p &> \frac{p+q}{2} \end{aligned} \tag{9}$$

As required. Now, we must show that $\frac{p+q}{2} > \sqrt{n}$, or equivalently prove that $2n < p^2 + q^2$. Now:

$$\begin{aligned} &p^2 + q^2 \\ &= p^2 + q^2 + 2n - 2n \\ &= 2n + (p^2 - 2n + q^2) \\ &= 2n + (p-q)^2 \quad \text{- By factoring} \end{aligned} \tag{10}$$

Now, since we know p > q, we can deduce that $(p - q) > 0$. So we have:

$$p^2 + q^2 = 2n + k \tag{11}$$

Where k = $(p - q)^2$, which is a positive integer. Therefore, we can conclude:

$$\begin{aligned} 2n + k &> 2n \\ &\text{And so:} \\ p^2 + q^2 &> 2n \\ \text{And equivalently,}& \\ \frac{p+q}{2} &> \sqrt{n} \end{aligned} \tag{12}$$

As required, thereby fully proving that $\sqrt{n} < \frac{p+q}{2} < p$. □

d. To prove that this algorithm terminates, we look to prove 3 things, namely: The "while" condition is satisfied when x = $\frac{p+q}{2}$, x = $\frac{p+q}{2}$ is the first value that satisfies the "while" condition, and the algorithm outputs. We begin by proving that the "while" condition is satisfied when x = $\frac{p+q}{2}$. To prove this, we must prove that when x = $\frac{p+q}{2}$ y is an integer.

Now:

$$x = \frac{p+q}{2}$$

$$y = \sqrt{x^2 - n}$$
$$y = \sqrt{\left(\frac{p+q}{2}\right)^2 - n} \quad \text{-Subbing in x}$$
$$y = \sqrt{\frac{p^2+2pq+q^2}{4} - n} \quad \text{-By expanding}$$
$$y = \sqrt{\frac{p^2+2pq+q^2}{4} - \frac{4n}{4}}$$
$$y = \sqrt{\frac{p^2+2pq+q^2-4n}{4}}$$
$$y = \sqrt{\frac{p^2+2pq+q^2-4pq}{4}} \quad \text{-Because n = pq}$$
$$y = \sqrt{\frac{p^2-2pq+q^2}{4}}$$
$$y = \sqrt{\left(\frac{(p-q)^2}{4}\right)} \quad \text{-By factoring}$$
$$y = \frac{(p-q)}{2}$$

Now, observe that both p and q are odd. This means:

$$\begin{aligned} p &= 2k+1 \quad \text{- For some integer k.} \\ \text{And,} & \\ q &= 2f+1 \quad \text{- For some integer f.} \\ \text{So:} & \\ p-q &= (2k+1)-(2f+1) \\ &= 2k+1-2f-1 \\ &= 2k-2f \\ &= 2(k-f) \quad \text{- An even number. So:} \\ \frac{(p-q)}{2} & \\ &= \frac{2(k-f)}{2} \\ &= k-f \end{aligned} \tag{13}$$

Which is simply the subtraction of two integers, which will give us an integer. Thereby proving that y is an integer, so the while loop will stop when x = $\frac{p+q}{2}$. Next, we wish to show that x = $\frac{p+q}{2}$ is the first value that satisfies the while condition. For this, observe from part a that we know x can only be either $\frac{p+q}{2}$ or $\frac{n+1}{2}$. We know from part b that p+q < n+1, so it logically follows that $\frac{p+q}{2} < \frac{n+1}{2}$. From part c, we know that $\sqrt{n} < \frac{p+q}{2}$. From the algorithm, we know that the first value of x will be $\lceil\sqrt{n}\rceil$, and if this value does not lead to y being an integer, this value will be incremented by 1. In other words, we start at $\lceil\sqrt{n}\rceil$ and count up until we find a value that works. From part a we know the only values that will work are $\frac{p+q}{2}$ and $\frac{n+1}{2}$. From part b we know that $\frac{n+1}{2} > \frac{p+q}{2}$, so it logically follows that $\lceil\sqrt{n}\rceil < \frac{p+q}{2} < \frac{n+1}{2}$. This means that when counting up from $\lceil\sqrt{n}\rceil$, we will encounter $\frac{p+q}{2}$ before encountering $\frac{n+1}{2}$, and since from part a we know these are the only two possible values of x, we can conclude that $\frac{p+q}{2}$ will be the first value that satisfies the while condition. Now we simply need to prove that the output of this algorithm is q. We know the output is x-y. Now:

$$
\begin{aligned}
x - y& \\
&= \frac{p+q}{2} - \frac{p-q}{2} \quad \text{-Subbing in x and y from earlier} \\
&= \frac{p+q-p+q}{2} \\
&= \frac{2q}{2} \\
&= q
\end{aligned} \tag{14}
$$

As required. We have proven all the things we needed to prove, and therefore proving the termination of this algorithm. □

e. The algorithm begins at $\lceil\sqrt{n}\rceil$ and counts up until x. Each count is one "test" of the while loop. To calculate the number of counts, we take the difference of x and $\lceil\sqrt{n}\rceil$, or x - $\lceil\sqrt{n}\rceil$. This however, will not count the first run of the algorithm, when x = $\lceil\sqrt{n}\rceil$, so we must add 1 more to include this run. This is exactly x - $\lceil\sqrt{n}\rceil$ +1. □

f. We want to prove that $x - \lceil\sqrt{n}\rceil < \frac{y^2}{2\sqrt{n}}$. Now:

$$\begin{aligned}
&(x-\sqrt{n})(x+\sqrt{n}) \\
&= x^2 - n \\
&= \left(\frac{p+q}{2}\right)^2 - n \quad \text{-Subbing in x} \\
&= \frac{p^2+2pq+q^2}{4} - n \\
&= \frac{p^2+2pq+q^2}{4} - \frac{4n}{4} \\
&= \frac{p^2+2pq+q^2-4n}{4} \\
&= \frac{p^2+2pq+q^2-4pq}{4} \\
&= \frac{p^2-2pq+q^2}{4} \\
&= \frac{(p-q)^2}{4} \quad \text{-By factoring} \\
&= \left(\frac{(p-q)}{2}\right)^2 \\
&= y^2
\end{aligned}$$

$$\begin{aligned}
(x-\sqrt{n})(x+\sqrt{n}) &= y^2 \\
(x-\sqrt{n}) &= \frac{y^2}{(x+\sqrt{n})} \\
&< \frac{y^2}{(\sqrt{n}+\sqrt{n}} \quad \text{-Since we know x} > \sqrt{n}\text{, and x is in the denominator.} \\
&= \frac{y^2}{2\sqrt{n}}
\end{aligned} \tag{15}$$

As required, thereby proving that $x - \lceil\sqrt{n}\rceil < \frac{y^2}{2\sqrt{n}}$. $\square$

g. We want to show that the algorithm terminates in at most $\frac{B^2}{2} + 1$ steps. We know from part e the algorithm takes exactly x - $\lceil \sqrt{n} \rceil$ +1 steps. Now:

$$
\begin{aligned}
\text{Number of steps} &= x - \lceil \sqrt{n} \rceil + 1 \\
&< \frac{y^2}{2\sqrt{n}} + 1 \quad \text{-Proven in part f} \\
&= \frac{\left(\frac{p-q}{2}\right)^2}{2\sqrt{n}} + 1 \quad \text{-Subbing in y} \\
&< \frac{\left(\frac{2B\sqrt[4]{n}}{2}\right)^2}{2\sqrt{n}} + 1 \quad \text{-From problem statement} \\
&= \frac{4B^2\sqrt[4]{n}^2}{4 \times 2\sqrt{n}} + 1 \\
&= \frac{B^2\sqrt[4]{n}^2}{2\sqrt{n}} + 1 \\
&= \frac{B^2\sqrt{n}}{2\sqrt{n}} + 1 \\
&= \frac{B^2}{2} + 1 \\
x - \lceil \sqrt{n} \rceil + 1 &< \frac{B^2}{2} + 1
\end{aligned}
\tag{16}
$$

As required. Thereby proving that the algorithm terminates in at most $\frac{B^2}{2} + 1$ steps. $\square$

Problem 4 — El Gamal is not semantically secure (12 marks)

We want to prove each of Mallory's assertions are correct, we begin with the first assertion, that if $\left(\frac{y}{p}\right) = 1, \left(\frac{C_2}{p}\right) = 1$ then C = E(M_1). Now:

$$\begin{aligned}
&\text{If } \left(\frac{y}{p}\right) = 1 \text{ and } \left(\frac{C_2}{p}\right) = 1 \text{ then:} \\
\left(\frac{C_2}{p}\right) &= \left(\frac{M}{p}\right)\left(\frac{y}{p}\right)^k \\
1 &= \left(\frac{M}{p}\right)(1)^k \\
1 &= \left(\frac{M}{p}\right)(1) \\
1 &= \left(\frac{M}{p}\right)
\end{aligned} \tag{17}$$

So $\left(\frac{M}{p}\right)$ is a quadratic residue of p, and therefore M must be M_1, since M_1 is a quadratic residue of p. Next we prove that if $\left(\frac{y}{p}\right) = 1, \left(\frac{C_2}{p}\right) = -1$ then C = E(M_2). Now:

$$\begin{aligned}
&\text{If } \left(\frac{y}{p}\right) = 1 \text{ and } \left(\frac{C_2}{p}\right) = -1 \text{ then:} \\
\left(\frac{C_2}{p}\right) &= \left(\frac{M}{p}\right)\left(\frac{y}{p}\right)^k \\
-1 &= \left(\frac{M}{p}\right)(1)^k \\
-1 &= \left(\frac{M}{p}\right)(1) \\
-1 &= \left(\frac{M}{p}\right)
\end{aligned} \tag{18}$$

So $\left(\frac{M}{p}\right)$ is a quadratic non-residue of p, and therefore M must be M_2, since M_2 is a quadratic non-residue of p. Next we prove that if $\left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = 1, \left(\frac{C_2}{p}\right) = 1$ then C = E(M_1). Now:

$$
\begin{aligned}
&\text{if } \left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = 1, \left(\frac{C_2}{p}\right) = 1 \text{ then:}\\
\left(\frac{C_2}{p}\right) &= \left(\frac{M}{p}\right)\left(\frac{y}{p}\right)^k\\
1 &= \left(\frac{M}{p}\right)(-1)^k\\
&\text{We need to know if k is even or odd.}\\
&\text{Observe that } \left(\frac{C_1}{p}\right) = 1\text{, this means that } g^k \text{ is a quadratic residue of p, so } g^k = (g^x)^2 \text{ for some integer x}\\
&\text{and by the power rules } g^k = g^{2x} \text{ for some integer x.}\\
&\text{Therefore, k is even. So:}\\
1 &= \left(\frac{M}{p}\right)(1)\\
1 &= \left(\frac{M}{p}\right)
\end{aligned}
\tag{19}
$$

So $\left(\frac{M}{p}\right)$ is a quadratic residue of p, and therefore M must be M_1, since M_1 is a quadratic residue of p. Next we prove if $\left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = 1, \left(\frac{C_2}{p}\right) = -1$ then C = E(M_2):

$$
\begin{aligned}
&\text{if } \left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = 1, \left(\frac{C_2}{p}\right) = -1 \text{ then:}\\
\left(\frac{C_2}{p}\right) &= \left(\frac{M}{p}\right)\left(\frac{y}{p}\right)^k\\
-1 &= \left(\frac{M}{p}\right)(-1)^k\\
&\text{Again, we need to know if k is even or odd.}\\
&\text{Observe that } \left(\frac{C_1}{p}\right) = 1\text{, this means that } g^k \text{ is a quadratic residue of p, so } g^k = (g^x)^2 \text{ for some integer x}\\
&\text{and by the power rules } g^k = g^{2x} \text{ for some integer x.}\\
&\text{Therefore, k is even. So:}\\
-1 &= \left(\frac{M}{p}\right)(1)\\
-1 &= \left(\frac{M}{p}\right)
\end{aligned}
\tag{20}
$$

So $\left(\frac{M}{p}\right)$ is a quadratic non-residue of p, and therefore M must be M_2, since M_2 is a quadratic residue of p. Next we prove if $\left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = -1, \left(\frac{C_2}{p}\right) = 1$ then C = E(M_2):

$$\text{if } \left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = -1, \left(\frac{C_2}{p}\right) = 1 \text{ then:}$$

$$\left(\frac{C_2}{p}\right) = \left(\frac{M}{p}\right)\left(\frac{y}{p}\right)^k$$

$$1 = \left(\frac{M}{p}\right)(-1)^k$$

Again, we need to know if k is even or odd.

Observe that $\left(\frac{C_1}{p}\right) = -1$, this means that g^k is a quadratic non-residue of p, so $g^k \neq (g^x)^2$ for any integer x. This means k is not a multiple of 2, so k must be odd. Now:

$$1 = \left(\frac{M}{p}\right)(-1)$$

$$-1 = \left(\frac{M}{p}\right) \tag{21}$$

So $\left(\frac{M}{p}\right)$ is a quadratic non-residue of p, and therefore M must be M_2, since M_2 is a quadratic residue of p. Finally, we prove that if $\left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = -1, \left(\frac{C_2}{p}\right) = -1$ then C = E(M_1):

$$\text{if } \left(\frac{y}{p}\right) = -1, \left(\frac{C_1}{p}\right) = -1, \left(\frac{C_2}{p}\right) = -1 \text{ then:}$$

$$\left(\frac{C_2}{p}\right) = \left(\frac{M}{p}\right)\left(\frac{y}{p}\right)^k$$

$$-1 = \left(\frac{M}{p}\right)(-1)^k$$

Again, we need to know if k is even or odd.

Observe that $\left(\frac{C_1}{p}\right) = -1$, this means that g^k is a quadratic non-residue of p, so $g^k \neq (g^x)^2$ for any integer x. This means k is not a multiple of 2, so k must be odd. Now:

$$-1 = \left(\frac{M}{p}\right)(-1)$$

$$1 = \left(\frac{M}{p}\right) \tag{22}$$

So $\left(\frac{M}{p}\right)$ is a quadratic residue of p, and therefore M must be M_1, since M_1 is a quadratic residue of p. We have now proven all of Mallory's assertions, thereby proving that the El Gamal system is not secure. □

Problem 5 — An IND-CPA, but not IND-CCA secure version of RSA (12 marks)

If Mallory sends in the cipher text C' = (s——t) $\oplus M_1$ for decryption, she can compute what C is very easily by following the decryption process. First, observe that C $\neq$ C', because C = (s——t) where as C' = (s——(t $\oplus M_1$)). The only case where these could be the same is if M_1 was a string of all zeros, so we will not allow M_1 to be a string of all zeros. Now by following the decryption process, we get:

$$\begin{aligned}
&\quad \text{C = (s——(t } \oplus M_1\text{)), so to decrypt:}\\
&M \equiv H(s^d \mod n) \oplus (t \oplus M_1)\\
&M \equiv H(r^{ed} \mod n \oplus (H(r) \oplus M_i \oplus M_1)\\
&M \equiv H(r) \oplus H(r) \oplus M_i \oplus M_1 \quad \text{-Because r} \equiv r^{ed} \mod n \text{ from the RSA rules in class.}\\
&M \equiv M_i \oplus M_1
\end{aligned} \tag{23}$$

We know that in xor, $0 \oplus 0 = 0$ and $1 \oplus 1 = 0$. This means that if M_i is M_1, all the bits being xor-ed will be the same, and therefore the M we get will simply be a string of all 0s. This is how Mallory can easily detect what M_i is; if the M she gets back from decrypting C' is all 0s, she knows immediately that M_1 was the message that was encrypted. If M is not all 0s, that is if there are any 1s in M, then Mallory instantly knows that the message encrypted was M_2. Since Mallory can easily figure out which message was encrypted, we have proven that this version of RSA is not IND-CCA secure. □