CSE 40622 Cryptography
Writing Assignment 03 (Lecture 03-05)

Name & netID: Walker Bagley (wbagley)

1. (15 pts, page 6) State the converse of the Euler's Theorem and prove it.

   - Note that, when $x^{\varphi(n)} \equiv 1 \pmod{n}$, $x^{\varphi(n)} = kn + 1$ for some integer k.

   Answer:
   Converse: If $x^{\varphi(n)} \equiv 1 \pmod{n}$ then $\gcd(x, n) = 1$

   Proof.

   $$\begin{aligned} x^{\varphi(n)} \equiv 1 \pmod{n} &\Rightarrow x^{\varphi(n)} = kn + 1 \\ &\Rightarrow 1 = x^{\varphi(n)} - kn \\ &\Rightarrow 1 = x^{\varphi(n)-1} \cdot x - kn \end{aligned}$$

   Considering this mod n, $x^{\varphi(n)-1} \equiv x^{\varphi(n)}x^{-1} \equiv x^{-1} \pmod{n}$. Since x^{-1} and k are both integers, then there exists a linear combination of x, n that equals 1, so applying Bézout's Identity, $\gcd(x, n) = 1$. □

2. (20 pts, page 6) x^n is usually not congruent to 1 modulo n. Therefore, there should not be a correct proof showing that $x^n \equiv 1 \pmod{n}$. However, Taeho made a mistake and shows a proof that looks like this.

   > Let's list all distinct numbers in $\mathbb{Z}_n$ as follows:
   >
   > $$x_1, x_2, x_3, \cdots, x_{n-1}, x_n$$
   >
   > Then, we choose an arbitrary positive number $a \in \mathbb{Z}_n$ and multiply it using modular multiplication:
   >
   > $$a \times_n x_1, a \times_n x_2, \cdots, a \times_n x_{n-1}, a \times_n x_n$$
   >
   > Then, all these numbers are distinct since we multiplied a to distinct numbers. Furthermore, it also follows that all $ax_i \mod n$ belongs to $\mathbb{Z}_n$.
   >
   > Then, both sequences have distinct numbers in $\mathbb{Z}_n$, and it follows that two sequences of numbers are the same numbers in different orders. This means:
   >
   > $$\prod_{i=1}^{n} x_i \equiv \prod_{i=1}^{n} (ax_i) \pmod{n} \Rightarrow \prod_{i=1}^{n} x_i \equiv a^n \prod_{i=1}^{n} x_i \pmod{n} \Rightarrow a^n \equiv 1 \pmod{n}$$

   What mistake did Taeho make and why is the proof above incorrect?

   Answer:

   When Taeho says "Then, all these numbers are distinct since we multiplied a to distinct numbers," he is in fact incorrect. If a and n are not coprime, then some of these multiplied numbers are the same. Consider the case where $n = 6$ and $a = 3$. Then for elements $2, 4 \in \mathbb{Z}_6$, $2 \cdot 3 \equiv 4 \cdot 3 \equiv 0 \pmod{4}$. Clearly these products are not distinct and therefore the rest of the proof falls apart.

3. (15 pts, page 6) Prove that, if $\gcd(x, n) = 1$, then we have $x^k \mod n = (x \mod n)^{(k \mod \varphi(n))} \mod n$ for any integer k.

   - Note that $k \mod \varphi(n)$ can be represented using the equation "$k \mod \varphi(n) = k - q\varphi(n)$" for some integer q (i.e., quotient) according to the definition of the modular reduction mod .

Answer:

$$\begin{aligned}
(x \mod n)^{k \mod \varphi(n)} \mod n &= (x \mod n)^{k - q_1\varphi(n)} \mod n \\
&= (x \mod n)^{k}(x \mod n)^{-q_1\varphi(n)} \mod n \\
&= (x + q_2 n)^{k}(x + q_3 n)^{-q_1\varphi(n)} \mod n \\
&= x^k x^{-q_1\varphi(n)} \mod n \\
&= x^k (x^{\varphi(n)})^{(-q_1)} \mod n \\
&= x^k 1^{(-q_1)} \mod n \\
&= x^k \mod n
\end{aligned}$$

4. Suppose we have strong attackers as follows. Describe how he/she can universally break the RSA encryption.

   ** Anyone has access to the public key by default.

   4.1. (5 pts) The attacker can do the factoring of $n = pq$. That is, he/she can figure out p and q from $n = pq$.

   Answer:

   If the attacker can figure out the factoring of $n = pq$, then they can easily calculate the totient function $\varphi(n) = (p-1)(q-1)$ and find the private key d of the public key e by calculating e's inverse mod $\varphi(n)$. From here they just need the ciphertext and since they know n, can calculate $c^d \mod n$ to get the original message.

   4.2. (5 pts) The attacker cannot factor $n = pq$, but s/he can somehow calculate $\varphi(n)$ from n.

   Answer:

   This process involves the exact same calculations as above, p, q are just not known. At the end of the day, p, q are relevant only to their product n and the totient $\varphi(n)$, both of which are known.

5. (10 pts) Assuming that the factoring of $n = pq$ is hard. Explain why it is hard to infer m in RSA by performing the e-th root modulo n as follows, given that e is a public parameter.

$$\sqrt[e]{c} \mod n = c^{\frac{1}{e}} \mod n = (m^e)^{e^{-1}} \mod n = m^{e \cdot e^{-1}} \mod n = m^1 \mod n = m$$

   - Note that $x^k \mod n = x^{k \mod \varphi(n)} \mod n$.
   - Try with example parameters $n = 15, e = 4, c = 10$.

   ** This is why $\sqrt{x}$ or $\sqrt[y]{x}$ may not be calculated efficiently in integer domains.

   Answer:

   Since finding $m = \sqrt[e]{c}$ is the same as finding some m s.t. $m^e = c$ and we are in a modulo group, this is very difficult. In the real group we could find this by some kind of binary search for m that would close in on the correct value with increasing accuracy. However, in the current domain, m^e can be all over the place for different values of m. Thus in a large enough domain, which we generally have in RSA encryption, one would have to loop over and calculate every possible value of m for this to be easy.