Homework 6

1. **RSA Assumption (5+12+5)**. Consider RSA encryption scheme with parameters $N = 35 = 5 \times 7$.

   (a) Find $\varphi(N)$ and $\mathbb{Z}_N^*$.

   (b) Use repeated squaring and complete the rows $X, X^2, X^4$ for all $X \in \mathbb{Z}_N^*$ as you have seen in the class (slides), that is, fill in the following table by adding as many columns as needed.

   **Solution.**

<table>
<thead>
<tr>
<th>$X$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>9</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>16</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$X^4$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$X$</th>
<th>18</th>
<th>19</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>26</th>
<th>27</th>
<th>29</th>
<th>31</th>
<th>32</th>
<th>33</th>
<th>34</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X^2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$X^4$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
(c) Find the row $X^5$ and show that $X^5$ is a bijection from $\mathbb{Z}^*_N$ to $\mathbb{Z}^*_N$.

**Solution.**

\[
\begin{array}{cccccccccccc}
X & 1 & 2 & 3 & 4 & 6 & 8 & 9 & 11 & 12 & 13 & 16 & 17 \\
X^4 & & & & & & & & & & & & \\
X^5 & & & & & & & & & & & & \\
\end{array}
\]

\[
\begin{array}{cccccccccccc}
X & 18 & 19 & 22 & 23 & 24 & 26 & 27 & 29 & 31 & 32 & 33 & 34 \\
X^4 & & & & & & & & & & & & \\
X^5 & & & & & & & & & & & & \\
\end{array}
\]
2. Answer the following questions (7+7+7+7 points):

   (a) (7 points) Compute the three least significant (decimal) digits of $6251007^{1960404}$ by hand. Explain your logic.

   Solution.
(b) (7 points) Is the following RSA signature scheme valid? (Justify your answer)

\[(r||m) = 24, \sigma = 196, N = 1165, e = 43\]

Here, \(m\) denotes the message, and \(r\) denotes the randomness used to sign \(m\) and \(\sigma\) denotes the signature. Moreover, \((r||m)\) denotes the concatenation of \(r\) and \(m\).

The signature algorithm \(\text{Sign}(m)\) returns \((r||m)^d \mod N\) where \(d\) is the inverse of \(e\) modulo \(\varphi(N)\). The verification algorithm \(\text{Ver}(m, \sigma)\) returns \(\frac{(r||m) \equiv \sigma^e \mod N}{\text{Solution}}\).
(c) (7 points) Remember that in RSA encryption and signature schemes, \( N = p \times q \) where \( p \) and \( q \) are two large primes. Show that in a RSA scheme (with public parameters \( N \) and \( e \)), if you know \( N \) and \( \varphi(N) \), then you can efficiently factorize \( N \) i.e. you can recover \( p \) and \( q \).

**Solution.**

(d) (7 points) Consider an encryption scheme where \( Enc(m) := m^e \mod N \) where \( e \) is a positive integer relatively prime to \( \varphi(N) \) and \( Dec(c) := c^d \mod N \) where \( d \) is the inverse of \( e \) modulo \( \varphi(N) \). Show that in this encryption scheme, if you know the encryption of \( m_1 \) and the encryption of \( m_2 \), then you can find the encryption of \( (m_1 \times m_2)^5 \).

**Solution.**
(e) (7 points) Suppose $n = 11413 = 101 \cdot 113$, where 101 and 113 are primes. Let $e_1 = 8765$ and $e_2 = 7653$.

   i. (2 points) Only one of the two exponents $e_1, e_2$ is a valid RSA encryption key, which one?

   ii. (3 points) For the valid encryption key, compute the corresponding decryption key $d$.

   iii. (2 points) Decrypt the cipher text $c = 3233$. 

3. Euler Phi Function (30 points)

(a) (10 points) Let $N = p_1^{e_1} \cdot p_2^{e_2} \cdots p_t^{e_t}$ represent the unique prime factorization of a natural number $N$, where $p_1 < p_2 < \cdots < p_t$ are prime numbers and $e_1, e_2, \ldots, e_t$ are natural numbers. Let $\mathbb{Z}_N^* = \{ x : 0 \leq x < N - 1, \gcd(x, N) = 1 \}$ and $\phi(N) = |\mathbb{Z}_N^*|$. Using the inclusion exclusion principle, prove that

$$
\phi(N) = N \cdot \left(1 - \frac{1}{p_1}\right) \cdot \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_t}\right).
$$

Solution.
(b) (5 points) For any $x \in \mathbb{Z}_N^*$, prove that

$$x^{\phi(N)} = 1 \mod N.$$ 

Hint: Consider the subgroup generated by $x$.

Solution.
(c) **Replacing** $\phi(N)$ **with** $\frac{\phi(N)}{2}$ **in RSA.** (15 points)

In RSA, we pick the exponent $e$ and the decryption key $d$ from the set $\mathbb{Z}_{\phi(N)}^*$. This problem shall show that we can choose $e, d \in \mathbb{Z}_{\phi(N)/2}^*$ instead.

Let $p, q$ be two distinct odd primes and define $N = pq$.

i. (2 points) For any $e \in \mathbb{Z}_{\phi(N)/2}^*$, prove that $x^e : \mathbb{Z}_N^* \rightarrow \mathbb{Z}_N^*$ is a bijection.

ii. (7 points) Consider any $x \in \mathbb{Z}_N^*$. Prove that $x^{\frac{\phi(N)}{2}} = 1 \mod p$ and $x^{\frac{\phi(N)}{2}} = 1 \mod q$.

iii. (3 points) Consider any $x \in \mathbb{Z}_N^*$. Prove that $x^{\frac{\phi(N)}{2}} = 1 \mod N$.

iv. (3 points) Suppose $e, d$ are integers that $e \cdot d = 1 \mod \frac{\phi(N)}{2}$. Show that $(x^e)^d = x \mod N$, for any $x \in \mathbb{Z}_N^*$. 

4. **Understanding hardness of the Discrete Logarithm Problem.** (15 points)

Suppose \((G, \circ)\) is a group of order \(N\) generated by \(g \in G\). Suppose there is an algorithm \(A_{DL}\) that, when given input \(X \in G\), it outputs \(x \in \{0, 1, \ldots, N - 1\}\) such that \(g^x = X\) with probability \(p_X\).

Think of it this way: The algorithm \(A_{DL}\) solves the discrete logarithm problem; however, for different inputs \(X \in G\), its success probability \(p_X\) may be different.

Let \(p = \frac{\sum_{X \in G} p_X}{N}\) represent the average success probability of \(A_{DL}\) solving the discrete logarithm problem when \(X\) is chosen uniformly at random from \(G\).

Construct a new algorithm \(B\) that takes any \(X \in G\) as input and outputs \(x \in \{0, 1, \ldots, N - 1\}\) (by making one call to the algorithm \(A_{DL}\)) such that \(g^x = X\) with probability \(p\). This new algorithm that you construct shall solve the discrete logarithm problem for every \(X \in G\) with the same probability \(p\).

*(Remark: Intuitively, this result shows that solving the discrete logarithm problem for any \(X \in G\) is no harder than solving the discrete logarithm problem for a random \(X \in G\).* )
5. **Concatenating a random bit string before a message.** (15 points)

Let \( m \in \{0,1\}^a \) be an arbitrary message. Define the set

\[
S_m = \{(r \parallel m): r \in \{0,1\}^b\}.
\]

Let \( p \) be an odd prime. Recall that in RSA encryption algorithm, we encrypted a message \( y \) chosen uniformly at random from this set \( S_m \).

Prove the following

\[
\Pr_{y \leftarrow S_m} [p \text{ divides } y] \leq 2^{-b} \cdot \left\lceil \frac{2^b}{p} \right\rceil.
\]

(Remark: This bound is tight as well. There exists \( m \) such that equality is achieved in the probability expression above. Intuitively, this result shows that the message \( y \) will be relatively prime to \( p \) with probability (roughly) \( 1 - 1/p \).)
6. Challenging: Inverting exponentiation function. (20 points)

Fix $N = pq$, where $p$ and $q$ are distinct odd primes. Let $e$ be a natural number such that $\gcd(e, \phi(N)) = 1$. Suppose there is an adversary $\mathcal{A}$ running in time $T$ such that

$$\Pr[\mathcal{A}([x^e \mod N]) = x] = 0.01$$

for $x$ chosen uniformly at random from $\mathbb{Z}_N^*$. Intuitively, this algorithm successfully finds the $e$-th root with probability 0.01, for a random $x$.

For any $\varepsilon \in (0, 1)$, construct an adversary $\mathcal{B}_\varepsilon$ (which, possibly, makes multiple calls to the adversary $\mathcal{A}$) such that

$$\Pr[\mathcal{B}_\varepsilon([x^e \mod N]) = x] = 1 - \varepsilon,$$

for every $x \in \mathbb{Z}_N^*$. The algorithm $\mathcal{B}_\varepsilon$ should have running time polynomial in $T, \log N$, and $\log 1/\varepsilon$. 
Collaborators: