Number Theory Preliminaries

Definition 1. Let n be a positive integer. We denote $\phi(n)$ (denoted as the *totient of* n) as

$$\phi(n) = |\{x \in \mathbb{N} \mid x \leqslant n, \gcd(x, n) = 1\}|$$

Theorem 1 (Chinese Remainder Theorem)**.** If we have k integers $n_1, \ldots, n_k$ greater than 1, denoted moduli, such that the n_i are pairwise coprime ($\gcd(n_i, n_j) = 1$ for all i, j) and any integers $a_1, \ldots, a_k$, then there exists one and only one x such that $0 \leqslant x < N$, where N is the product of all n_i, such that the following holds:

$$x \equiv a_i \pmod{n_i} \text{ for all } i$$

In fact, we can efficiently construct this x using the Extended Euclidean Algorithm, but we leave this as an exercise to the reader. For this problem session, the reader may consider CRT as a black box that both proves the existence of and constructs an x satisfying the above properties.

Problem 1: Prove that for $a, b \in \mathbb{Z}$ where $\gcd(a, b) = 1$,

$$\phi(ab) = \phi(a)\phi(b).$$

Is this true for general a, b?

RSA Problems

Definition 2. The *RSA cryptosystem*, a public-key (asymmetric) scheme, operates as follows:

1. Alice generates two random primes p, q, and computes $N = pq$, $\phi(N) = (p-1)(q-1)$.
2. She then chooses public exponent e, such that $\gcd(e, \phi(N)) = 1$.
3. Finally, she computes $d \equiv e^{-1} \pmod{\phi(N)}$. The public key is (e, N), and the private key is (p, q, d) (although only d is required for decryption).
4. Encryption on a plaintext M is performed as $C \equiv M^e \pmod{N}$, and decryption can likewise be done as $M \equiv C^d \pmod{N}$.

Problem 2: We've intercepted the following encryption from Neil and Alvin! It contains Neil's favorite number. Alvin's public key is $N = 65$, $e = 5$, and the encrypted message Neil sent is $C = 6$. What's Neil's favorite number?

Theorem 2 (Euler's Totient Theorem)**.** If a and n are coprime ($\gcd(a, n) = 1$), then

$$a^{\phi(n)} \equiv 1 \pmod{n}.$$

Problem 3: Using Euler's Totient Theorem, prove the correctness of RSA, that is, that for (almost[1]) every plaintext M, it is true that $M \equiv (M^e)^d \pmod{N}$. Which M does this fail for?

[1] This proof fails in some cases, but we need more machinery to prove those cases, so this will suffice for now.

Problem 4: Using the Chinese Remainder Theorem, derive Håstad's broadcast attack, which states the following: Suppose the same message M is sent to k different people, using different public keys, but all with the same public exponent e. Then, if an attacker Eve intercepts $C_1, \ldots, C_k$, they can efficiently[2] recover M as long as $k \geqslant e$.

Definition 3. Another interesting application of RSA is in digital signatures. In the *RSA digital signature scheme*, if you want to prove to someone that you're the original owner of a public key, they send you a challenge, you "decrypt" it, generating a signature, which you then send back. Anyone can then verify using the public key that your signature encrypts to the original challenge.

However, because RSA decryption can be somewhat computationally intensive, we have something called RSA-CRT. In RSA-CRT, we construct the signature modulo p and q instead of modulo $N = pq$, and then we can use the Chinese Remainder Theorem to construct the final signature from these partial signatures:

$$\left.\begin{aligned} s_1 &= m^d \pmod{p} \\ s_2 &= m^d \pmod{q} \end{aligned}\right\} \implies s = m^d \pmod{N}$$

Problem 5: Prove the following *fault attack* on the RSA-CRT signature scheme. Suppose a fault happens in calculating s_2, and the signer computes a $\tilde{s}_2$ such that $\tilde{s}_2 \neq m^d \pmod{q}$. Then, the signer uses CRT and obtains a faulty signature $\tilde{s}$. Show that using $\tilde{s}$, m, e, and N, you can recover the private key.

Definition 4. An encryption scheme is said to be *perfectly secret* if for every probability distribution over the message space $\mathcal{M}$, every message $m \in \mathcal{M}$, and every ciphertext $c \in \mathcal{C}$ for which $\Pr[C = c] > 0$, the following holds:

$$\Pr[M = m \mid C = c] = \Pr[M = m]$$

Definition 5. The *One-Time Pad* (OTP) encryption scheme operates as follows: for a given bit sequence m, the plaintext, we require the secret key to be any uniformly random bit sequence k such that $|k| \geqslant |m|$. The ciphertext (encryption) is then constructed as $c_i = m_i \oplus k_i$, where $\oplus$ is the XOR operation. To decrypt, the same operation is performed: $m_i = c_i \oplus k_i$.

Problem 6: Show that the One-Time Pad is perfectly secret. That is, a ciphertext by itself reveals no information about the plaintext.

Problem 7: Show that for any scheme to be perfectly secret, the size of key space must be at least the size of the message space, that is, $|\mathcal{K}| \geqslant |\mathcal{M}|$. Is this sufficient for a scheme to be perfectly secret?

[2] Polynomial time; you can assume CRT runs in polynomial time as well. The exact computational model does not matter that much for this question.