Exercise 1. Factorize $n = 33$ given non-trivial square roots of unity 10 and 23.

Solution. Observe that

$$10^2 \bmod 13 = 1 \qquad\qquad 23^2 \bmod 33 = 1$$

Hence, $9 \cdot 11 \bmod 33 = 0$ and $22 \cdot 24 \bmod 33 = 0$. The factors are

$$\begin{aligned} \gcd(9, 33) &= 3 & \qquad \gcd(24, 33) &= 3 \\ \gcd(11, 33) &= 11 & \qquad \gcd(22, 33) &= 11 \end{aligned}$$

Hence, $33 = 3 \cdot 11$.

Exercise 2. Factorize $n = 1457$. Suppose you have learned that 1457 is a probable prime to base 187, and a strong pseudoprime to base 187.

Solution. If n is a probable prime, but a strong pseudoprime, then there exists a nontrivial square root of 1 modulo n. $1457 - 1 = 2^4 \cdot 91$.

$$\begin{aligned} 187^{91} \bmod 1457 &= 187 \\ 187^2 \bmod 1457 &= 1 \end{aligned}$$

It means that 187 is a nontrivial square root of unity, and

$$\gcd(186, 1457) = 31 \qquad\qquad \gcd(188, 1457) = 47$$

Hence, $1457 = 31 \cdot 47$.

Exercise 3. Factorize RSA modulus $n = 2491$, given that $e = 3$ and $d = 1595$.

Solution. By Fermat theorem, if $\gcd(a, n) = 1$, then

$$a^{\varphi(n)} = 1 \pmod{n} \ .$$

We also know that

$$e \cdot d \equiv 1 \pmod{\varphi(n)} \implies e \cdot d - 1 = \beta \cdot \varphi(n) \implies a^{e \cdot d - 1} = \left(a^{\varphi(n)}\right)^{\beta} \bmod n = 1 \ .$$

The value $e \cdot d - 1 = 4784 = 2^4 \cdot 299$. Choose a, i.e. 7.

$$\begin{aligned} 7^{299} \bmod 2491 &= 847 \\ 847^2 \bmod 2491 &= 1 \end{aligned}$$

847 is a square root of 1.

$$\gcd(846, 2491) = 47 \qquad\qquad \gcd(848, 2491) = 53$$

and it means that $2491 = 47 \cdot 53$.

Exercise 4. Show that textbook RSA is not secure against chosen plaintext attack. The IND-CPA game is defined as follows

1. The challenger generates a new key pair PK, SK and publishes PK to the adversary, the challenger retains SK.
2. The adversary may perform a polynomially bounded number of calls to the encryption oracle or other operations.
3. Eventually, the adversary submits two distinct plaintexts M_0 and M_1 to the challenger.
4. The chellenger selects a bit $b \in \{0, 1\}$ uniformly at random, and sends the challenge ciphertext $C = E(PK, M_b)$ back to the adversary.
5. The adversary is free to perform any number of additional computations.
6. Finally, the adversary outputs a guess for the value b.

A cryptosystem is indistinguishable under chosen plaintext attack (is IND-CPA secure) if every probabilistic polynomial time adversary has only a negligible advantage over random guessing.

Solution. The adversary knows the RSA public key (n, e), where n is the modulus and e is public exponent. During step 2 of the algorithm, the adversary can pre-compute values $C_0 = M_0^e \bmod n$ and $C_1 = M_1^e \bmod n$. Upon receiving the challenge ciphertext $C = M_b^e \bmod n$ the adversary can compare C to C_0 and C_1 and thus it will always win the game.

Exercise 5. Use homomorphic properties of RSA to show that textbook RSA is not secure against adaptive chosen ciphertext attack (CCA2). The IND-CCA2 game is defined as follows.

1. The challenger generates a new key pair PK, SK and publishes PK to the adversary, the challenger retains SK.
2. The adversary may perform any number calls to the encryption or decryption oracles, or other operations.
3. Eventually, the adversary submits two distinct chosen plaintexts M_0 and M_1 to the challenger.
4. The challenger selects a bit $b \in \{0, 1\}$ uniformly at random, and sends the challenge ciphertext $C = E(PK, M_b)$ back to the adversary.
5. The adversary is free to perform any number of additional computations, calls to the encryption and decryption oracles, but may not submit the challenge ciphertext C to the decryption oracle.
6. Finally, the adversary outputs a guess for the value b.

The plaintext RSA is homomorphic w.r.t. multiplication, meaning that

$$\begin{cases} C_1 = m_1^e \bmod n \\ C_2 = m_2^e \bmod n \end{cases} \Longrightarrow C_1 \times C_2 = m_1^e \cdot m_2^e \bmod n = (m_1 m_2)^e \bmod n \ .$$

Solution. Upon receiving the challenge ciphertext $C = M_b^e \bmod n$, the adversary selects a blinding factor, i.e. 2, computes ciphertext $2^e \bmod n$ and multiplies with ciphertext C as follows

$$2^e \cdot M_b^e \bmod n = (2 \cdot M_b)^e \bmod n \ .$$

The adversary submits $(2 \cdot M_b)^e \bmod n$ to the decryption oracle, which computes

$$\big((2 \cdot M_b)^e \bmod n\big)^d \bmod n = (2 \cdot M_b)^{ed} \bmod n = 2 \cdot M_b \ ,$$

and sends $2 \cdot M_b$ back to the adversary. All the adversary needs to do is to divide the obtained blinded plaintext by 2 and compare if $M_b = M_0$ or $M_b = M_1$.