CS 70 Discrete Mathematics and Probability Theory
Spring 2021 **DIS 4A**

# 1 RSA Practice

Bob would like to receive encrypted messages from Alice via RSA.

(a) Bob chooses $p = 7$ and $q = 11$. His public key is $(N, e)$. What is $N$?

(b) What number is $e$ relatively prime to?

(c) $e$ need not be prime itself, but what is the smallest prime number $e$ can be? Use this value for $e$ in all subsequent computations.

(d) What is $\gcd(e, (p-1)(q-1))$?

(e) What is the decryption exponent $d$?

(f) Now imagine that Alice wants to send Bob the message 30. She applies her encryption function $E$ to 30. What is her encrypted message?

(g) Bob receives the encrypted message, and applies his decryption function $D$ to it. What is $D$ applied to the received message?

**Solution:**

(a) $N = pq = 77$.

(b) $e$ must be relatively prime to $(p-1)(q-1) = 60$.

(c) We cannot take $e = 2, 3, 5$, so we take $e = 7$.

(d) By design, $\gcd(e, (p-1)(q-1)) = 1$ always.

(e) The decryption exponent is $d = e^{-1} \pmod{60} = 43$, which could be found through Euclid's extended GCD algorithm.

(f) The encrypted message is $E(30) = 30^7 \equiv 2 \pmod{77}$. We can obtain this answer via repeated squaring.

$$30^7 \equiv 30 \cdot 30^6 \equiv 30 \cdot (30^2 \bmod 77)^3 \equiv 30 \cdot 53^3 \equiv (30 \cdot 53 \bmod 77) \cdot (53^2 \bmod 77) \equiv 50 \cdot 37$$
$$\equiv 2 \pmod{77}.$$

(g) We have $D(2) = 2^{43} \equiv 30 \pmod{77}$. Again, we can use repeated squaring.

$$\begin{aligned} 2^{43} &\equiv 2 \cdot 2^{42} \equiv 2 \cdot (2^2 \bmod 77)^{21} \equiv 2 \cdot 4^{21} \equiv (2 \cdot 4 \bmod 77) \cdot 4^{20} \equiv 8 \cdot (4^2 \bmod 77)^{10} \\ &\equiv 8 \cdot 16^{10} \equiv 8 \cdot (16^2 \bmod 77)^5 \equiv 8 \cdot 25^5 \equiv (8 \cdot 25 \bmod 77) \cdot 25^4 \equiv 46 \cdot (25^2 \bmod 77)^2 \\ &\equiv 46 \cdot (9^2 \bmod 77) \equiv 46 \cdot 4 \equiv 30 \pmod{77}. \end{aligned}$$

# 2 RSA with Multiple Keys

Members of a secret society know a secret word. They transmit this secret word $x$ between each other many times, each time encrypting it with the RSA method. Eve, who is listening to all of their communications, notices that in all of the public keys they use, the exponent $e$ is the same. Therefore the public keys used look like $(N_1, e), \ldots, (N_k, e)$ where no two $N_i$'s are the same. Assume that the message is $x$ such that $0 \leq x < N_i$ for every $i$.

(a) Suppose Eve sees the public keys $(p_1 q_1, 7)$ and $(p_1 q_2, 7)$ as well as the corresponding transmissions. Can Eve use this knowledge to break the encryption? If so, how? Assume that Eve cannot compute prime factors efficiently. Think of $p_1, q_1, q_2$ as massive 1024-bit numbers. Assume $p_1, q_1, q_2$ are all distinct and are valid primes for RSA to be carried out.

(b) The secret society has wised up to Eve and changed their choices of $N$, in addition to changing their word $x$. Now, Eve sees keys $(p_1 q_1, 3)$, $(p_2 q_2, 3)$, and $(p_3 q_3, 3)$ along with their transmissions. Argue why Eve cannot break the encryption in the same way as above. Assume $p_1, p_2, p_3, q_1, q_2, q_3$ are all distinct and are valid primes for RSA to be carried out.

(c) Let's say the secret $x$ was not changed ($e = 3$), so they used the same public keys as before, but did not transmit different messages. How can Eve figure out $x$?

**Solution:**

(a) Normally, the difficulty of cracking RSA hinges upon the believed difficulty of factoring large numbers. If Eve were given just $p_1 q_1$, she would (probably) not be able to figure out the factors.

However, Eve has access to two public keys, so yes, she will be able to figure it out. Note that $\gcd(p_1 q_1, p_1 q_2) = p_1$. Taking GCDs is actually an efficient operation thanks to the Euclidean Algorithm. Therefore, she can figure out the value of $p_1$, and from there figure out the value of $q_1$ and $q_2$ since she has $p_1 q_1$ and $p_1 q_2$.

(b) Since none of the $N$'s have common factors, she cannot find a GCD to divide out of any of the $N$s. Hence the approach above does not work.

(c) Eve observes $x^3 \pmod{N_1}$, $x^3 \pmod{N_2}$, $x^3 \pmod{N_3}$. Since all $N_1, N_2, N_3$ are pairwise relatively prime, Eve can use the Chinese Remainder Theorem to figure out $x^3 \pmod{N_1 N_2 N_3}$. However, once she gets that, she knows $x$, since $x < N_1$, $x < N_2$, and $x < N_3$, which implies $x^3 < N_1 N_2 N_3$. Uh oh! (Binary search can compute compute $x$ from $x^3$ in the integers since one can tell if some number is too large or too small.)

## 3 RSA for Concert Tickets

Alice wants to tell Bob her concert ticket number, $m$, which is an integer between 0 and 100 inclusive. She wants to tell Bob over an insecure channel that Eve can listen in on, but Alice does not want Eve to know her ticket number.

(a) Bob announces his public key $(N = pq, e)$, where $N$ is large (512 bits). Alice encrypts her message using RSA. Eve sees the encrypted message, and figures out what Alice's ticket number is. How did she do it?

(b) Alice decides to be a bit more elaborate. She picks a random number $r$ that is 256 bits long, so that it is too hard to guess. She encrypts that and sends it to Bob, and also computes $rm$, encrypts that, and sends it to Bob. Eve is aware of what Alice did, but does not know the value of $r$. How can she figure out $m$?

**Solution:**

(a) There are only 101 possible values for Alice's ticket number, so Eve can try encrypting all 101 values with Bob's public key and find out which one matches the one Alice sent.

(b) Alice sends $x = r^e \pmod{pq}$, as well as $y = (rm)^e = r^e m^e = x m^e \pmod{pq}$. We can find $x^{-1} \pmod{N}$ using the Extended Euclidean Algorithm, and multiplying this value by $y$ gives us $m^e \pmod{N}$. Now we proceed as in the previous part to find $m$.