CS 70 Discrete Mathematics and Probability Theory
Fall 2020 HW 5

1 RSA with Just One Prime

Given the message $x \in \{0, 1, \ldots, N-1\}$ and $N = pq$, where p and q are prime numbers, conventional RSA encrypts x with $y = E(x) \equiv x^e \pmod{N}$. The decryption is done by $D(y) \equiv y^d \pmod{N}$, where d is the inverse of $e \pmod{(p-1)(q-1)}$.

Alice is trying to send a message to Bob, and as usual, Eve is trying to decipher what the message is. One day, Bob gets lazy and tells Alice that he will now use $N = p$, where p is a 1024-bit prime number, as part of his public key. He tells Alice that it's okay, since Eve will have to try out 2^{1024} combinations to guess x. It is very likely that Eve will not find out the secret message in a reasonable amount of time! In this problem, we will see whether Bob is right or wrong. Assume that Eve has found out about this new setup and that she knows the public key.

Similar to the original method, for any message $x \in \{0, 1, \ldots, N-1\}$, $E(x) \equiv x^e \pmod{p}$, and $D(y) \equiv y^d \pmod{p}$. Choose e such that it is coprime with $p-1$, and choose $d \equiv e^{-1} \pmod{p-1}$.

(a) Prove that the message x is recovered after it goes through your new encryption and decryption functions, $E(x)$ and $D(y)$.

(b) Can Eve compute d in the decryption function? If so, by what algorithm and approximately how many iterations does it take for it to terminate?

(c) Given part (b), how would Eve recover x and what algorithm would she use? Approximately how many iterations does it take to terminate?

(d) Based on the previous parts, can Eve recover the original message in a reasonable amount of time? Explain.

Solution:

(a) We want to show x is recovered by $E(x)$ and $D(y)$, such that $D(E(x)) = x$. In other words, $x^{ed} \equiv x \pmod{p}$ $\forall x \in \{0, 1, \ldots, N-1\}$.
Proof: By construction of d, we know that $ed \equiv 1 \pmod{p-1}$. This means we can write $ed = k(p-1)+1$, for some integer k, and $x^{ed} = x^{k(p-1)+1}$.

- x is a multiple of p: Then this means $x = 0$, and indeed, $x^{ed} \equiv 0 \pmod{p}$.
- x is not a multiple of p: Then $x^{ed} \equiv x^{k(p-1)+1} \equiv x^{k(p-1)}x \equiv 1^k x \equiv x \pmod{p}$, by using FLT.

And for both cases, we have shown that x is recovered by $E(D(y))$.

(b) Since Eve knows the value of $N = p$, and the fact that $d \equiv e^{-1} \pmod{p-1}$, she can compute d using EGCD. Since EGCD decreases the largest number by at least a factor of two every two iterations, Eve needs at most $2n$ iterations, where n is the number of bits of the larger input. This means at most 2048 iterations.

(c) Since Eve now has d from part 3, and the encrypted message y, she can calculate x directly by using $D(y) = x \equiv y^d \pmod{p}$. She can now use exponentiation by repeated squaring, giving her no more than 1024 iterations.

(d) Assuming each recursive call in EGCD and exponentiation by squaring have reasonable operation time costs, Eve only needs at most 3×1024 iterations, which can easily be done with today's computing power.

2 Squared RSA

(a) Prove the identity $a^{p(p-1)} \equiv 1 \pmod{p^2}$, where a is coprime to p, and p is prime. (Hint: Try to mimic the proof of Fermat's Little Theorem from the notes.)

(b) Now consider the RSA scheme: the public key is $(N = p^2q^2, e)$ for primes p and q, with e relatively prime to $p(p-1)q(q-1)$. The private key is $d = e^{-1} \pmod{p(p-1)q(q-1)}$. Prove that the scheme is correct for x relatively prime to both p and q, i.e. $x^{ed} \equiv x \pmod{N}$. (Hint: Try to mimic the proof of RSA correctness from the notes.)

Solution:

(a) We mimic the proof of Fermat's Little Theorem from the notes.
Let S be the set of all numbers between 1 and $p^2 - 1$ (inclusive) which are relatively prime to p. We can write

$$S = \{1, 2, \ldots, p-1, p+1, \ldots, p^2-1\}$$

Define the set

$$T = \{a, 2a, \ldots, (p-1)a, (p+1)a, \ldots, (p^2-1)a\}$$

We'll show that $S \subseteq T$ and $T \subseteq S$, allowing us to conclude $S = T$:

- $S \subseteq T$: Let $x \in S$. Since $\gcd(a, p) = 1$, the inverse of a exists $\pmod{p^2}$. For ease of notation, we use a^{-1} to denote the quantity $a^{-1} \pmod{p^2}$. We know $\gcd(a^{-1}, p) = 1$, because a^{-1} has an inverse $\pmod{p^2}$ too. Combining this with the fact that $\gcd(x, p) = 1$, we have $\gcd(a^{-1}x, p) = 1$. This tells us $a^{-1}x \in S$, so $a(a^{-1}x) = x \in T$.
- $T \subseteq S$: Let $ax \in T$, where $x \in S$. We know $\gcd(x, p) = 1$ because $x \in S$. Since $\gcd(a, p) = 1$ as well, we know the product xs cannot share any prime factors with p as well, i.e. $\gcd(xs, p) = 1$. This means $xs \in S$ as well, which proves the containment.

We now follow the proof of Fermat's Little Theorem. Since $S = T$, we have:

$$\prod_{s_i \in S} s_i \equiv \prod_{t_i \in T} t_i \pmod{p^2}$$

However, since we defined $T = \{a, 2a, \ldots, (p-1)a, (p+1)a, \ldots, (p^2-1)a\}$:

$$\prod_{t_i \in T} t_i \equiv \prod_{s_i \in S} a s_i \equiv a^{|S|} \prod_{s_i \in S} s_i \pmod{p^2}$$

We can now conclude $(\prod_{s_i \in S} s_i) \equiv a^{|S|} (\prod_{s_i \in S} s_i) \pmod{p^2}$.
Each $s_i \in S$ is coprime to p, so their product $\prod_{s_i \in S} s_i$ is as well. Then, we can multiply both sides of our equivalence with the inverse of $\prod_{s_i \in S} s_i$ to obtain $a^{|S|} \equiv 1 \pmod{p^2}$. Since $|S| = p(p-1)$, we have gotten the desired result.

Alternate Solution: We can use Fermat's Little Theorem, combined with the Binomial Theorem, to get the result. Since $\gcd(a, p) = 1$ and p is prime, $a^{p-1} \equiv 1 \pmod{p}$, so we can write $a^{p-1} = \ell p + 1$ for some integer ℓ. Then,

$$(a^{p-1})^p = (\ell p + 1)^p = \sum_{i=0}^{p} \binom{n}{i} (\ell p)^i = 1 + p \cdot (\ell p) + \binom{p}{2} (\ell p)^2 + \cdots + (\ell p)^p,$$

and since all of the terms other than the first term are divisible by p^2, $a^{p(p-1)} \equiv 1 \pmod{p^2}$.

(b) By the definition of d above, $ed = 1 + kp(p-1)q(q-1)$ for some k. Look at the equation $x^{ed} \equiv x \pmod{N}$ modulo p^2 first:

$$x^{ed} \equiv x^{1+kp(p-1)q(q-1)} \equiv x \cdot (x^{p(p-1)})^{kq(q-1)} \equiv x \pmod{p^2}$$

where we used the identity above. If we look at the equation modulo q^2, we obtain the same result. Hence, $x^{ed} \equiv x \pmod{p^2 q^2}$.

Remark: The first part of the question mirrors the proof of Fermat's Little Theorem. The second and third parts of the question mirror the proof of correctness of RSA.

3 The CRT and Lagrange Interpolation

Let $n_1, \ldots n_k$ be pairwise co-prime, i.e. n_i and n_j are co-prime for all $i \neq j$. The Chinese Remainder Theorem (CRT) tells us that there exist solutions to the following system of congruences:

$$x \equiv a_1 \pmod{n_1} \quad (1)$$
$$x \equiv a_2 \pmod{n_2} \quad (2)$$
$$\vdots \quad (\vdots)$$
$$x \equiv a_k \pmod{n_k} \quad (k)$$

and all solutions are equivalent $\pmod{n_1 n_2 \cdots n_k}$. For this problem, parts (a)-(c) will walk us through a proof of the Chinese Remainder Theorem. We will then use the CRT to revisit Lagrange interpolation.

(a) We start by proving the $k = 2$ case: Prove that we can always find an integer x_1 that solves (1) and (2) with $a_1 = 1, a_2 = 0$. Similarly, prove that we can always find an integer x_2 that solves (1) and (2) with $a_1 = 0, a_2 = 1$.

(b) Use part (a) to prove that we can always find at least one solution to (1) and (2) for any a_1, a_2. Furthermore, prove that all possible solutions are equivalent $\pmod{n_1 n_2}$.

(c) Now we can tackle the case of arbitrary k: Use part (b) to prove that there exists a solution x to (1)-(k) and that this solution is unique $\pmod{n_1 n_2 \cdots n_k}$.

For polynomials $p_1(x)$, $p_2(x)$ and $q(x)$ we say that $p_1(x) \equiv p_2(x) \bmod q(x)$ if $p_1(x) - p_2(x)$ is of the form $q(x) \times m(x)$ for some polynomial $m(x)$.

(d) Define the polynomials $x-a$ and $x-b$ to be co-prime if they have no common divisor of degree 1. Assuming that the CRT still holds when replacing x, a_i and n_i with polynomials (using the definition of co-prime polynomials just given), show that the system of congruences

$$\begin{aligned} p(x) &\equiv y_1 \pmod{(x - x_1)} && (1') \\ p(x) &\equiv y_2 \pmod{(x - x_2)} && (2') \\ &\vdots && (\vdots) \\ p(x) &\equiv y_k \pmod{(x - x_k)} && (k') \end{aligned}$$

has a unique solution $\pmod{(x-x_1)\cdots(x-x_k)}$ whenever the x_i are pairwise distinct. What is the connection to Lagrange interpolation?

Hint: To show that a unique solution exists, you may use the fact that the CRT has a unique solution when certain properties are satisfied.

Solution:

(a) Since $\gcd(n_1, n_2) = 1$, there exist integers k_1, k_2 such that $1 = k_1 n_1 + k_2 n_2$. Setting $x_1 = k_2 n_2 = 1 - k_1 n_1$ and $x_2 = k_1 n_1 = 1 - k_2 n_2$ we obtain the two desired solutions.

(b) Using the x_1 and x_2 we found in Part (a), we show that $a_1 x_1 + a_2 x_2 \pmod{n_1 n_2}$ is a solution to the desired equivalences:

$$\begin{aligned} a_1 x_1 + a_2 x_2 &\equiv a_1 \cdot 1 + a_2 \cdot 0 \equiv a_1 \pmod{n_1} \\ a_1 x_1 + a_2 x_2 &\equiv a_1 \cdot 0 + a_2 \cdot 1 \equiv a_2 \pmod{n_2}. \end{aligned}$$

Such result is also unique. Say that we have two difference solutions $x = c$ and $x = c'$, which both satisfy $x \equiv a_1 \pmod{n_1}$ and $x \equiv a_2 \pmod{n_2}$. This would give us $c \equiv c' \pmod{n_1}$ and $c \equiv c' \pmod{n_2}$, which suggests that $(c - c')$ is divisible by n_1 and n_2. Since n_1 and n_2 are coprime, $gcd(n_1, n_2) = 1$, $(c - c')$ is divisible by $n_1 n_2$. Writing it in modular form gives us $c \equiv c' \pmod{n_1 n_2}$. Therefore, all the result is unique with respect to $\pmod{n_1 n_2}$

(c) We use induction on k. Part (b) handles the base case, $k = 2$. For the inductive hypothesis, assume for $k \leq l$, the system (1)-(k) has a unique solution $a \pmod{n_1 \cdots n_k}$. Now consider $k = l+1$, so we add the equation $x \equiv a_{l+1} \pmod{n_{l+1}}$ to our system, resulting in

$$x \equiv a \pmod{n_1 \cdots n_l}$$
$$x \equiv a_{l+1} \pmod{n_{l+1}}.$$

Since the n_i are pairwise coprime, $n_1 n_2 \cdots n_l$ and n_{l+1} are coprime. Part (b) tells us that there exists a unique solution $a' \pmod{n_1 \cdots n_l n_{l+1}}$. We conclude that a' is the unique solution to (1)-($l+1$), when taken $\pmod{n_1 n_2 \cdots n_l n_{l+1}}$.

(d) We only need to check that $q_i(x) = (x - x_i)$ and $q_j(x) = (x - x_j)$ are coprime whenever $x_i \neq x_j$; that is, that they don't share a common divisor of degree 1. If $d_i(x) = a_i x + b_i$ is a divisor of $q_i(x)$, then $q_i(x) = q'(x)(a_i x + b_i)$ for some polynomial $q'(x)$. But since $q_i(x)$ is of degree 1, $q'(x)$ must be of degree 0 and hence a constant, so $d_i(x)$ must be a constant multiple of $q_i(x)$. Similarly, any degree 1 divisor d_j of $q_j(x)$ must be a constant multiple of $q_j(x)$, and if $x_i \neq x_j$, then none of these multiples overlap, so $q_i(x)$ and $q_j(x)$ are coprime.

From our result in part (d), the congruences (1′)-(k') assert that we are looking for a polynomial $p(x)$ such that $p(x_i) = y_i$. The CRT then establishes the existence of $p(x)$, and that it is unique modulo a degree k polynomial. That is, $p(x)$ is unique if its degree is at most $k-1$. Lagrange interpolation finds $p(x)$.

4 Polynomials in Fields

Define the sequence of polynomials by $P_0(x) = x + 12$, $P_1(x) = x^2 - 5x + 5$ and $P_n(x) = xP_{n-2}(x) - P_{n-1}(x)$.

(For instance, $P_2(x) = 17x - 5$ and $P_3(x) = x^3 - 5x^2 - 12x + 5$.)

(a) Show that $P_n(7) \equiv 0 \pmod{19}$ for every $n \in \mathbb{N}$.

(b) Show that, for every prime q, if $P_{2017}(x) \not\equiv 0 \pmod{q}$, then $P_{2017}(x)$ has at most 2017 roots modulo q.

Solution:

(a) Prove by strong induction. Base cases:

$$P_0(7) \equiv 7 + 12 \equiv 19 \equiv 0 \pmod{19}$$
$$P_1(7) \equiv 7^2 - 5 \cdot 7 + 5 \equiv 49 - 35 + 5 \equiv 19 \equiv 0 \pmod{19}$$

Inductive step: Assume $P_n(7) \equiv 0 \pmod{19}$ for every $n \leq k$. Then

$$\begin{aligned} P_{k+1}(7) &\equiv xP_{k-1}(7) - P_k(7) \pmod{19} \\ &\equiv x \cdot 0 - 0 \pmod{19} \\ &\equiv 0 \pmod{19}. \end{aligned}$$

Hence, we have $P_n(7) \equiv 0 \pmod{19}$ for all naturnal numbers n.

(b) This question asks to prove that, for all prime numbers q, if $P_{2017}(x)$ is a non-zero polynomial $\pmod{q}$, then $P_{2017}(x)$ has at most 2017 roots $\pmod{q}$.

The proof of Property 1 of polynomials (a polynomial of degree d can have at most d roots) still works in the finite field GF(q). Therefore we need only show that P_{2017} has degree at most 2017. We prove that $\deg(P_n) \leq n$ for $n > 1$ by strong induction. Base cases:

$$\begin{aligned}
\deg(P_0) &= \deg(x+12) = 1 \\
\deg(P_1) &= \deg(x^2 - 5x + 5) = 2 \\
\deg(P_2) &= \deg(xP_0(x) - P_1(x)) \leq 2 \\
\deg(P_3) &= \deg(xP_1(x) - P_2(x)) \leq 3
\end{aligned}$$

Assuming degree of $P_n \leq n$ for all $2 \leq n \leq k$, then

$$\begin{aligned}
\deg(P_{k+1}(x)) &\leq \max\{\deg(xP_{k-1}(x)), \deg(P_k(x))\} \\
&= \max\{1 + \deg(P_{k-1}(x)), \deg(P_k(x))\} \\
&\leq \max\{1 + k - 1, k\} \\
&\leq k \\
&\leq k + 1.
\end{aligned}$$

Thus the proof holds for all $n \geq 2, n \in \mathbb{N}$.

5 Secrets in the United Nations

A vault in the United Nations can be opened with a secret combination $s \in \mathbb{Z}$. In only two situations should this vault be opened: (i) all 193 member countries must agree, or (ii) at least 55 countries, plus the U.N. Secretary-General, must agree.

(a) Propose a scheme that gives private information to the Secretary-General and all 193 member countries so that the secret combination s can only be recovered under either one of the two specified conditions.

(b) The General Assembly of the UN decides to add an extra level of security: each of the 193 member countries has a delegation of 12 representatives, all of whom must agree in order for that country to help open the vault. Propose a scheme that adds this new feature. The scheme should give private information to the Secretary-General and to each representative of each country.

Solution:

(a) Create a polynomial of degree 192 and give each country one point. Give the Secretary General $193 - 55 = 138$ points, so that if she collaborates with 55 countries, they will have a total of

192 points and can reconstruct the polynomial. Without the Secretary-General, the polynomial can still be recovered if all 192 countries come together. (We do all our work in GF(p) where $p \geq d+1$).

Alternatively, we could have one scheme for condition (i) and another for (ii). The first condition is the secret-sharing setup we discussed in the notes, so a single polynomial of degree 192 suffices, with each country receiving one point, and evaluation at zero returning the combination s. For the second condition, create a polynomial f of degree 1 with $f(0) = s$, and give $f(1)$ to the Secretary-General. Now create a second polynomial g of degree 54, with $g(0) = f(2)$, and give one point of g to each country. This way any 55 countries can recover $g(0) = f(2)$, and then can consult with the Secretary-General to recover $s = f(0)$ from $f(1)$ and $f(2)$.

(b) We'll layer an *additional* round of secret-sharing onto the scheme from part (a). If t_i is the key given to the ith country, produce a degree-11 polynomial f_i so that $f_i(0) = t_i$, and give one point of f_i to each of the 12 delegates. Do the same for each country (using different f_i each time, of course).

6 Secret Sharing with Spies

An officer stored an important letter in her safe. In case she is killed in battle, she decides to share the password (which is a number) with her troops. However, everyone knows that there are 3 spies among the troops, but no one knows who they are except for the three spies themselves. The 3 spies can coordinate with each other and they will either lie and make people not able to open the safe, or will open the safe themselves if they can. Therefore, the officer would like a scheme to share the password that satisfies the following conditions:

- When M of them get together, they are guaranteed to be able to open the safe even if they have spies among them.
- The 3 spies must not be able to open the safe all by themselves.

Please help the officer to design a scheme to share her password. What is the scheme? What is the smallest M? Show your work and argue why your scheme works and any smaller M couldn't work. (The troops only have one chance to open the safe; if they fail the safe will self-destruct.)

Solution:

The key insight is to realize that both polynomial-based secret-sharing and polynomial-based error correction work on the basis of evaluating an underlying polynomial at many points and then trying to recover that polynomial. Hence they can be easily combined.

Suppose the password is s. The officer can construct a polynomial $P(x)$ such that $s = P(0)$ and share $(i, P(i))$ to the i-th person in her troops. Then the problem is: what should the degree of $P(x)$ be and what is the smallest M?

First, the degree of polynomial d should not be less than 3. It is because when $d < 3$, the 3 spies can decide the polynomial $P(x)$ uniquely. Thus, n will be at least 4 symbols.

Let's choose a polynomial $P(x)$ of degree 3 such that $s = P(0)$. We now view the 3 spies as 3 general errors. Then the smallest $M = 10$ since n is at least 4 symbols and we have $k = 3$ general errors, leading us to a "codeword" of $4 + 2 \cdot 3 = 10$ symbols (or people in our case). Even though the 3 spies are among the 10 people and try to lie on their numbers, the 10 people can still be able to correct the $k = 3$ general errors by the Berlekamp-Welch algorithm and find the correct $P(x)$.

Alternative solution:

Another valid approach is making $P(x)$ of degree $M - 1$ and adding 6 public points to deal with 3 general errors from the spies. In other words, in addition to their own point $(i, P(i))$, everyone also knows the values of 6 more points, $(t+1, P(t+1)), (t+2, P(t+2)), \ldots, (t+6, P(t+6))$, where t is the number of the troops. The spies have access to total of $3 + 6 = 9$ points so the degree $M - 1$ must be at least 9 to prevent the spies from opening the safe by themselves. Therefore, the minimum M is 10.