RSA, Polynomials, Secret Sharing, Erasure Errors 4

Computer Science Mentors 70

February 12-14, 2018

1 RSA

1.1 Introduction

Main idea: find an encryption function E and an decryption function D such that $D(E(x)) = x$. In other words, two people can encrypt and decrypt a message x (an integer) if they know E and D.

Mechanism:

Public key: (N, e)

Private key: d

Assumption: Given N, there is no efficient algorithm to determine $(p-1)(q-1)$.

Encryption function: $E(x) = x^e$ mod N

Decryption function: $D(x) = x^d$ mod N, where $d = e^{-1}$ mod$(p-1)(q-1)$

1.2 Proof

Goal: To show $(x^e)^d \equiv x \bmod (N)$, $\forall x \in \{0, 1, \cdots, N-1\}$.
Since $ed \equiv 1 \bmod (p-1)(q-1)$ (that's how we define d), $ed = 1 + k(p-1)(q-1)$ for some integer k. Therefore, $x^{ed} = x^{1+k(p-1)(q-1)}$

$\implies x^{ed} - x = x^{1+k(p-1)(q-1)} - x = x(x^{k(p-1)(q-1)} - 1)$.

Claim: $p \mid x^{ed} - x$
Case 1 ($x \not\equiv 0 \bmod p$):
According to Fermat's Little Theorem, $x^{p-1} \equiv 1 \bmod p$.
$\implies (x^{p-1})^{k(q-1)} \equiv 1^{k(q-1)} \mod p$
$\implies x^{k(p-1)(q-1)} - 1 \equiv 0 \bmod p$
$\implies x^{k(p-1)(q-1)+1} - x \equiv 0 \bmod p$
$\implies p \mid x^{ed} - x$
Case 2 ($x \equiv 0 \bmod p$):
$p \mid x(x^{ed-1} - 1)$
$\implies p \mid x^{ed} - x$
By symmetry, $q \mid x^{ed} - x$. Thus, $pq \mid x^{ed} - x$, which means $(x^e)^d \equiv x \bmod (N)$.

1.3 Questions

1. **How does RSA work?**

   a. Alice wants to send Bob a message $m = 5$ using his public key ($n = 26$, $e = 11$). What cipher text E(m) will Alice send?

   b. What is the value of d (Bobs private key) in this scheme? Note that traditional RSA schemes use much larger prime numbers, so its harder to break n down into its prime factors than it is in this problem.

2. In RSA, if Alice wants to send a confidential message to Bob, she uses Bobs public key to encode it. Then Bob uses his private key to decode the message. Suppose that Bob chose $N = 77$. And then Bob chose $e = 3$ so his public key is (3, 77). And then Bob chose $d = 26$ so his private key is (26, 77).
   Will this work for encoding and decoding messages? If not, where did Bob first go wrong in the above sequence of steps and what is the consequence of that error? If it does work, then show that it works.

1.4 Extra Practice

1. **Coin tosses over text messages**
   You and one of your friends want to get your hands on the new gadget thats coming out. One of you has to wait in line overnight so that you have a chance to get the gadgets while they last. In order to decide who this person should be, you both agree to toss a coin. But you wont meet each other until the day of the actual sale and you have to settle this coin toss over text messages (using your old gadgets). Obviously neither of you trusts the other person to simply do the coin toss and report the results.

   How can you use RSA to help fix the problem?

2 Polynomials

2.1 Introduction

We can use two representations for polynomials: value representation and coefficient representation.

1. Value representation: if polynomial $P(x)$ has degree $n-1$ then we can uniquely reconstruct it from any n distinct points
2. Coefficient representation: if a polynomial $P(x)$ has degree $n-1$ then it can be uniquely described by its n coefficients

2.2 Questions

1. 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 $P_3(x) = x^3 - 5x^2 - 12x + 5$.) Show that $P_n(7) \equiv 0 \pmod{19}$ for every $n \in N$.

2. Let $P(x)$ be a polynomial of degree 2 over GF(7).
   1. Suppose $P(0) = 3$ and $P(1) = 2$. How many values can $P(5)$ have? How many distinct polynomials are there?

   2. Suppose we only know $P(0) = 3$. How many possible pairs of $(P(1), P(5))$ are there? How many different polynomials are there?

   3. If we only know k different points, where $k \leq d$, of a degree d polynomial over GF(p), how many possible polynomials are there?

3 Secret Sharing

3.1 Introduction

1. Set up: n officials indexed from 1 to n. Secret s is an integer. Use $GF(q)$: q is a prime number such that $q > n$ and $q > s$.
2. Scheme: Pick a random $p(x)$ of degree $k-1$ such that $p(0) = s$. Thus,
   (a) Any k officials can use Lagrange Interpolation to find $p(x)$, therefore x.
   (b) Any $k-1$ officials have no information about s.

3.2 Questions

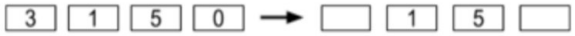
1. Suppose the Oral Exam questions are created by 2 TAs and 3 Readers. The answers are all encrypted and we know that:
   (a) Together, both TAs should be able to access the answers
   (b) All 3 Readers can also access the answers
   (c) One TA and one Reader should also be able to do the same

   Design a secret sharing scheme to make this work.

4 Erasure Errors

4.1 Introduction

We want to send n packets and we know that k packets could get lost. We use a polynomial under GF(q).



How many more points does Alice need to send to account for k possible errors? __
What degree will the resulting polynomial be? ___
How large should q be if Alice is sending n packets with k erasure errors, where each packet is a integer between 0 and m?

What would happen if Alice instead send $n+k-1$? Why will Bob be unable to recover the message?

4.2 Questions

1. Suppose $A = 1$, $B = 2$, $C = 3$, $D = 4$, and $E = 5$. Assume we want to send a message of length 3. Recover the lost part of the message, or explain why it can not be done.
   1. C_AA
   2. CE_ _

2. Suppose we want to send n packets, and we know $p = 20\%$ of the packets will be erased. How many extra packets should we send? What happens if p increases (say to 90%)?