El Gamal Crypto System

Kason Kershner

November 2023

1 Important Rules:

- It is really important that Bob picks a new ephemeral key b for every message. If Bob uses the same b value to encrypt another message m2:

Ciphertext:

r1 = α^2 (mod p)

t1 = m $\cdot$ B^b (mod p)

Then, the r value stays the same

r2 = α^2 (mod p)

t2 = m $\cdot$ B^b (mod p)

So r1 = r2.

- If Eve does a known plaintext attack on m2, she can find B^b = t2$\cdot$(m2)$^{-1}$ (mod p)

- She never has to solve the Discrete Log Problem.

- Notice that if Eve can make an educated guess as to what the plaintext is, She can't check to see if she was right because every ephemeral key produces a different value of t.

2 Contrast This With RSA:

- If Eve ever guesses m, she can check to see if she was right by computing m^e (mod n) to see if she computes the ciphertext.

- To defend against this with RSA, it is important to pad your messages.
  - Pad: append a random number to the end of the message.
  - This should be used whenever possible in applied cryptography .

3 Attacking the Discrete Log Problem:

Brute Force:

p is the modulus, where p is roughly 2^b
- b is the number of bits in p

Goal: Solve $y = a^x \pmod{p}$ for x. (We know y, a, and b)

Try all possibilities for x where $1 < x < p - 1$

- This has a running time of $O(p) = O(2^2)$. This is exponential time...

Baby-step Giant-step:

Goal: Solve $y = a^x \pmod{p}$ for x. (We know y, a, and b)

Let N = Ceiling($\sqrt{p}$)

Think about x as a number written in base N
- Note that $N^2 = \text{Ceiling}(\sqrt{p})^2 > p > x$

So $x = i \cdot N + j$
- i and j are digits in base N and between 0 and N

Goal: Solve for i and j:

y = $a^{i \cdot N+j}$ (mod p)
y=$a^{i \cdot N} \cdot a^{j}$ (mod p)

y$\cdot a^{-iN}$ = a^{j}

- This is the same trick as meet in the middle.

Create tables then find the entry that shows up in both tables. There will only be 1 value in both tables.

Table1:

Baby steps:
All possibilities for a^{j} (mod p) 0 <= j <= N

Table 2:

Giant steps:
y$\cdot^{-iN}$ (mod p) = y$\cdot(a^{-1})^{n}$ (mod p)

Running time is O(3N) = O(N) = O($\sqrt{p}$) = O($2^{b/2}$)

Example:

Find x where 7^{x} = 11 (mod 23) using baby-step giant-step

N = $\sqrt{23}$ = 5

Baby steps:

7^{i} (mod 23).

0 <= 1 < 5

$7^0 = 1 \pmod{23}$

$7^1 = 7 \pmod{23}$

$7^2 = 49 = 3 \pmod{23}$

$7^3 = 21 \pmod{23}$

$7^4 = -2 \cdot 7 = -14 = 9 \pmod{23}$

table:

0 — 1
1 — 7
2 — 3
3 — 21
4 — 9

Giant steps:

$11 \cdot 7^{-j \cdot 5} = 11 \cdot (7^{-5})^j \pmod{p}$
0 <= j < N

$7^{-5} = 10 \pmod{23}$ by the extended Euclidean algorithm.

Table:

$10^2 = 100 = 8 \pmod{23}$
$10^4 = 64 = 18 \pmod{23}$
$10^5 = 18 \cdot 10 = -50 = -4 = 19 \pmod{23}$

table:

0 — 11 (mod 23)
1 — 21 (mod 23)
2 — 15 (mod 23)

21 is in both tables, so j is 1 and i is 3. x = 3N + 1 = 15 + 1 = 16.