COMPUTER SCIENCE & ENGINEERING
IIT KANPUR

Department of Computer Science and Engineering

Assignment #3

Enciphered

Roll: *21111261 —— 21111263 —— 21111063*
Class: *CS 641A*
Session: *Semester- II*
Email: *anindyag21@iitk.ac.in; anindyag@cse.iitk.ac.in*

Course: *CS 641A: Modern Cryptology* – Teacher: *Prof. Manindra Agrawal*
Submission date: *16th-Feb*

Problem 2

List the commands used in the game to reach the ciphertext.

Solution:

`go, enter, pluck, c, back, give, back, back, thrnxxtzy, read.`

Problem 3

Give a detailed analysis of how you figured out the password? (Explain in less than 500 words)

Solution:

Prime $p = 455470209427676832372575348833$ Given pair:

$(429, 431955503618234519808008749742)$
$(1973, 176325509039323911968355873643)$
$(7596, 984869714048619924872947226 13)$

Mathematical expression behind this: $x_{=}g^{a_i} * password$ (will run i from 1 to 3) Given pair can be expressed as:

$$g^{429} * password = 431955503618234519808008749742 = x_1 \quad (1)$$

$$g^{1973} * password = 176325509039323911968355873643 = x_2 \quad (2)$$

$$g^{7596} * password = 984869714048619924872947226 13 = x_3 \quad (3)$$

Using three equation we get

- Dividing (2) by (1)
$$g^{1973-429} = g^{1544} = \frac{x_2}{x_1} \mod p = y_1 \quad (say)$$
- Dividing (3) by (2)
$$g^{7596-1973} = g^{5623} = \frac{x_3}{x_2} \mod p = y_2 \quad (say)$$
- Dividing (3) by (1)
$$g^{7596-429} = g^{7167} = \frac{x_3}{x_1} \mod p = y_3 \quad (say)$$

Compute Modular Inverse: As per FLT
$$g^{p-1} = 1.$$
This implies
$$g^{-1} = g^{p-2} \mod p.$$
So, inverse computation converts to exponentiation. Square and multiply algorithm will help to exponentiation. It will takes $O(\log m)$ time to compute g^m[**CNT**]. So efficient.

Let $m = (m_{s-1}, m_{s-2}, \cdots, m_1, m_0)_2$ be the binary expression of the exponent m, where $m_i \in \{0, 1\}$.

Algorithm:

- initialize $t = 1 \mod p$
- for $(i = s - 1;\ i \geq 0;\ i - -)\{$
  - set $t = t^2 \mod p$
  - if $(m_i = 1)$ set $t = t \times g \mod p$

  $\}$
- Return t;

We try it two different manner. Let us illustrates the first technique. It is clearly observed that $1544, 5623, 7167$ are co-prime to each other and 5623 is a prime. So, by Bezout identity,

$$1544u_1 + 5623v_1 = 1 \text{ where } u_1 = -2298, v_1 = 631 \quad (7)$$

$$1544u_2 + 7167v_2 = 1 \text{ where } u_2 = -2929, v_2 = 631 \quad (8)$$

$$5623u_3 + 7167v_3 = 1 \text{ where } u_3 = 2929, v_3 = -2298 \quad (9)$$

We compute these u_i, v_i using Extended Euclidean Algorithm.

Running time is $O(\log \min(u_i, v_i))$. Choose equation (7) (you can choose anyone of them),

$$g^{1544u_1+5623v_1} = g \mod p$$

$$(g^{1544})^{-2298} \times (g^{5623})^{631} = g \mod p$$

Now from equations (1, 2 or 3) we can write
$$password = x_i * (g^{a_i})^{-1} \mod p$$

For $i = 1$,

$$password = 43195550361823451980800874974 2 * (g)^{429} \mod p$$

Now, we perform the computation using GP-PARI calculator. Other freely available number theoretic libraries are NTL, GMP library. We put the GP-PARI command to find g and password.

```
p=455470209427676832372575348833;
x1= 43195550361823451980800874974 2;
x2= 17632550903932391196835587 3643;
x3= 98486971404861992487294722613;

y1=Mod(x2/x1,p);
y2=Mod(x3/x2,p);
y3=Mod(x3/x1,p);

z1=Mod(y1^ (-2298),p)    //z1=6367334591911148292811805 2957
z2= Mod((y2)^631,p)      //z2=34726700838987729837401766 7230
z3=z1*z2;
g=z3;

t=Mod(g^429,p);
password=Mod(x1/t,p);
```

At the end of computation we got

```
g = 52565085417963311027694339;
password: 13472154209765902984527 3957;
```

2nd

Using these above relation (1,2, and 3) goal is to find g. Following computation helps to find g.

1. $z_1 = \dfrac{y_2}{(y_1)^3} = g^{5623-3\times1544} = g^{991}$
2. $z_2 = \dfrac{y_3}{(z_1)^7} = g^{7167-7\times991} = g^{230}$
3. $z_3 = \dfrac{z_1}{(z_2)^4} = g^{991-4\times230} = g^{71}$
4. $z_4 = \dfrac{z_2}{(z_3)^3} = g^{230-3\times71} = g^{17}$
5. $z_5 = \dfrac{z_1}{(z_3)^{14}} = g^{991-14\times71} = g^{-3}$
6. $z_6 = z_4 \cdot (z_5)^5 = g^{17+5\times(-3)} = g^2$
7. $z_7 = z_5 \times (z_6)^2 = g^{-3+2\times2} = g$
   Hence $z_7 = g$. Modular reduction carried out after each step.

Now from given equation we can write

$$password = x_i \cdot (g^{a_i})^{-1} \mod p$$

For $i = 1$,

$$password = 43195550361823451980800874 9742 * (g)^{429} \mod p$$

Now, we perform the computation using GP-PARI calculator[**GP**]. Other freely available number theoretic libraries are NTL, GMP library. We put the command to find g and password.

```
p=455470209427676832372575348833;
x1= 43195550361823451980800874 9742;
x2= 176325509039323911968355873643;
x3= 984869714048619924872947 22613;

y1=Mod(x2/x1,p);
y2=Mod(x3/x2,p);
y3=Mod(x3/x1,p);

z1=Mod(y2/(y1^3),p);
z2=Mod(y3/(z1^7),p);
z3= Mod(z1/(z2^4),p);
z4=Mod(z2/(z3^3),p);
z5=Mod(z1/(z3^14),p);
z6=Mod(z4*z5^5,p);
z7=Mod(z6^2*z5,p);

g=z7;

t=Mod(g^429,p);
password=Mod(x1/t,p);
```

At the end of computation we got

```
g = 525650854179633110276943 39;
password: 13472154209765902984527 3957;
```

Problem 4

What was the final command used to clear this level?

Solution:

Password:

```
13472154209765902984527 3957
```

References

[1] Das, Abhijit. Computational number theory. CRC Press, 2016.

[2] Kawamoto, Fuminori, and Koshi Tomita. "GP/PARI calculator GP/PARI calculator." Journal of the Mathematical Society of Japan 60.3 (2008): 865-903.