Q1 Team Name
0 Points

Enciphered

Q2 Commands
10 Points

List the commands used in the game to reach the ciphertext

go/enter, enter, pluck, c, back, give, back, back, thrnxxtzy, read

Q3 Analysis
50 Points

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

Prime $p = 455470209427676832372575348833$

Given pair:

$$(429, 431955503618234519808008749742)$$
$$(1973, 176325509039323911968355873643)$$
$$(7596, 98486971404861992487294722613)$$

Mathematical expression behind this:

$x = g^{a_i} * password$ ($i \in \{1, 2, 3\}$)

Given pair can be expressed as:

$g^{429} * password =$
$431955503618234519808008749742 = x_1 \quad (1)$
$g^{1973} * password =$
$176325509039323911968355873643 = x_2 \quad (2)$
$g^{7596} * password =$
$98486971404861992487294722613 = x_3 \quad (3)$

Using three equation we get

=> Dividing (2) by (1)

$$g^{1973-429} = g^{1544} = x_2/x_1 \mod p = y_1\text{(say)} \quad (4)$$

=> Dividing (3) by (2)

$$g^{7596-1973} = g^{5623} = x_3/x_2 \mod p = y_2\text{(say)} \quad (5)$$

=> Dividing (3) by (1)

$$g^{7596-429} = g^{7167} = x_3/x_1 \mod p = y_3\text{(say)} \quad (6)$$

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

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

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 $t = t * g \mod p$
}
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,

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

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

$5623u_3 + 7167v_3 = 1$ 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 $i = 1$,

$$password = 4319555036182345198080087497742 * (g)^{429} \mod p$$

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=4554702094276768323725753488833;
x1= 4319555036182345198080087497742;
x2= 1763255090393239119683558736743;
x3= 9848697140486199248729472613;

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

z1=Mod(y1^ (-2298),p)   //z1=636733459191114829281180529557
z2= Mod((y2)^631,p)
//z2=3472670083898772983740176672300
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: 134721542097659029845273957;
==

Another Way: Using these above relation (4, 5, and 6) goal is to find g. Following computations help to find g.

$$z_1 = y_2/(y_1)^3 = g^{5623-3*1544} = g^{991}$$
$$z_2 = y_3/(z_1)^7 = g^{7167-7*991} = g^{230}$$
$$z_3 = z_1/(z_2)^4 = g^{991-4*230} = g^{71}$$
$$z_4 = z_2/(z_3)^3 = g^{230-3*71} = g^{17}$$
$$z_5 = z_1/(z_3)^{14} = g^{991-14*71} = g^{-3}$$
$$z_6 = z_4 * (z_5)^5 = g^{17+5*(-3)} = g^2$$
$$z_7 = z_5 * (z_6)^2 = g^{-3+2*2} = g$$

Hence $z_7 = g$. Modular reduction carried out after each step.

Like above from equation (1), compute

$$password = 431955503618234519808008749742 * (g)^{429} \mod p$$

We put the GP-PARI command to find g and password.

```
----------------------------------------------------------------------
p=455470209427676832372575348833;
x1= 431955503618234519808008749742;
x2= 176325509039323911968355873643;
x3= 98486971404861992487294722613;

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 = 52565085417963311027694339;
password: 134721542097659029845273957;

So, in two different approach we got the same result.

Reference:

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.

Note: Please go through the attached LaTexed pdf of this assignment (can be found in code section).

Q4 Password

10 Points

What was the final command used to clear this level?

Password: 134721542097659029845273957;

Q5 Codes

0 Points

Upload any code that you have used to solve this level

Enciphered_Assignment3.pdf

Download

Assignment 3

GRADED

GROUP

Utkarsh Srivastava

Gargi Sarkar
Anindya Ganguly
View or edit group

TOTAL POINTS

70 / 70 pts

QUESTION 1

Team Name **0** / 0 pts

QUESTION 2

Commands **10** / 10 pts

QUESTION 3

Analysis **50** / 50 pts

QUESTION 4

Password **10** / 10 pts

QUESTION 5

Codes **0** / 0 pts

✔	**+ 0 pts**	Correct