Q1 Team Name
0 Points

Pasta_Sandwich

Q2 Commands
10 Points

List the commands used in the game to reach the ciphertext

go, enter, pick, 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)

After giving mushroom to the spirit in the hole, the spirit helped us by giving magic word “thrnxxtzy” to open the door in main chamber. Using the commands we reached the problem.
Hint was given that password is an element of multiplicative group Z_p^* where p = 455470209427676832372575348833 is a prime number. We were also given pairs number in the form (a, password* g^a).
Thus we were able to form 3 equations for 3 pairs
Suppose password = x
Eqn 1: (x * g^ 429) ≅ 431955503618234519808008749742(say n1) mod p
Eqn 2: (x * g^ 1973 ≅ 176325509039323911968355873643(say n2) mod p
Eqn 3: (x * g^ 7596) ≅ 98486971404861992487294722613(say n3) mod p

Dividing eqn2 by eqn1:-
g^1544 ≅(n2 * n1^(-1)) mod p → eqn4

Similarly, dividing eqn3 by eqn2 and eqn3 by eqn1:-
g^5623 ≅ (n3 * n2^(-1)) mod p →eqn5
g^7167 ≅ (n3 * n1^(-1)) mod p →eqn6

If we find n1^(-1) under modp i.e.modular inverse of n1 under modp then we can get the value of g^5623. After that we can repeatedly the powers of g and get the value of g. After we get value of g, we can get the value of our password x.

Since n1 and p are co-primes, using fermat’s little theorem we can write:-
n1^(p-1)≅ 1 mod p
Or,
1=n1^(p-1) mod p
by multiplying both sides by n1^(-1) :-
n1^(-1) =n1^(p-2) mod p .....using this we can find modular inverse of n1 under mod p
=70749967902234717329046816 40

By eqn4:-
g^1544 ≅(n2 * n1^(-1)) mod p
≅(176325509039323911968355873643 * 7074996790223471732904681640) mod p
=111590994894663139264552154672 - eqn 7

Similarly by eqn5:-
g^5623 ≅ (n3 * n2^(-1)) mod p
g^5623 = 420413074251022028027270785553 - eqn 8

Eqn 6:-
g^7167 ≅ (n3 * n1^(-1)) mod p
g^7167 = 110411376670918912626907526185 - eqn 9

Now, we repeatedly reduce the powers of g to get the value of g....
Multiplying both sides of eqn 8 by ((g^1544)^3)^-1
(g^5623 * ((g^1544)^3)^-1) = 111590994894663139264552154672 * ((g^1544)^3)^-1 mod p
g^991 = 161798558270556961732424822635 - eqn 10

Multiplying both sides of eqn 9 by ((g^991)^7)^-1
(g^7167 * ((g^991)^7)^-1) = 110411376670918912626907526185 * ((g^991)^7)^-1 mod p
g^230 = 263509268584013168241508095725 - eqn 11

Multiplying both sides of eqn 10 by ((g^230)^4)^-1
(g^991 * ((g^230)^4)^-1) = 161798558270556961732424822635 * ((g^230)^4)^-1 mod p
g^71 = 200335025748509210338477331839 - eqn 12

Multiplying both sides of eqn 11 by ((g^71)^3)^-1

Assignment 3 ● GRADED

GROUP
Manjyot Singh Nanra
Ayush Sahni
Sharanya Saha
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

Multiplying both sides of eqn 11 by ((g^71)^3)^-1
(g^230 * ((g^71)^3)^-1) = 263509268584013168241508095725 * ((g^71)^3)^-1 mod p
g^17 = 140738752429105879936732752189 - eqn 13

Multiplying both sides of eqn 12 by ((g^17)^4)^-1
(g^71 * ((g^17)^4)^-1) = 200335025748509210338477331839 * ((g^17)^4)^-1 mod p
g3 = 83679736938813925904466001390 - eqn 14

Multiplying both sides of eqn 13 by ((g^3)^5)^-1
(g^17 * ((g^3)^5)^-1) = 140738752429105879936732752189 * ((g^3)^5)^-1 mod p
g2 = 108044907665466013935627786069 - eqn 15

Finally,
g = g3 * g2^-1 mod p
g ≅ 52565085417963311027694339 mod p
Thus g can be any value of form (52565085417963311027694339 + p*k) for any integer k.

We see that for k=0, g is matching with the hint number provided in the problem statement.
Thus, we can be sure that the value of g = 52565085417963311027694339.
Now that we have g, we can easily calculate the password.
Substituting value of g in eqn 1:
x*g^429 ≅ n1 mod p
Multiplying both sides with (g^429)^(-1)
x ≅ (n1 * (g^429)^-1) mod p
Substituting the value of n1 and g:-
x = 134721542097659029845273957
Thus the final password is 134721542097659029845273957
(Note the above calculations have been done using python code written by us which is attached below)

Q4 Password

10 Points

What was the final command used to clear this level?

134721542097659029845273957

Q5 Codes

0 Points

Upload any code that you have used to solve this level

assignment3.ipynb Download

In [1]:

```
import math

def mod_inverse(y, p):
    # p will be a prime,
    # thus y,p will be coprime
    # Calculating using Fermat Little theorem
    return pow(y, p-2, p)
```

In [2]:

```
# Assigning values
n1, n2, n3 = 431955503618234519808008749742,
176325509039323911968355873643, 98486971404861992487294722613
p = 455470209427676832372575348833
```

In [3]:

```
# finding value (n2* n1^-1) mod p
n1_inv = mod_inverse(n1, p)
print("n1 inverse is", n1_inv)
g1544 = (n2* n1_inv) % p
print("g^1544 =",g1544)
```

```
n1 inverse is 70749996790223471732904681640
g^1544 = 111590994894663139264552154672
```

In [4]:

```
# finding value of (n3 * n2^-1) mod p
n2_inv = mod_inverse(n2, p)
print("n2 inverse is", n2_inv)

g5623 = (n3* n2_inv) % p
print("g^5623 = ",g5623)
```

```
n2 inverse is 228947149478752602606353685125
g^5623 =  420413074251022028027270785553
```

In [5]:

```
# finding value of (n3 * n1^-1) mod p

g7167 = (n3* n1_inv) % p
print("g^7167 = ", g7167)
```

```
g^7167 =  110411376670918912626907526185
```

In [6]:

```
# 5623 - (1544 * 3)
# finding g^991
g991 = (g5623 * mod_inverse(pow(g1544, 3, p), p)) % p
print("g^991=",g991)
```

g^991= 161798558270556961732424822635

In [7]:

```
# 7167 - 991 * 7
# finding g^230
g230 = (g7167 * mod_inverse(pow(g991, 7, p), p)) % p
print("g^230=",g230)
```

g^230= 263509268584013168241508095725

In [8]:

```
# 991 - 230 * 4
# finding g^71
g71 = (g991 * mod_inverse(pow(g230, 4, p), p)) % p
print("g^71=",g71)
```

g^71= 200335025748509210338477331839

In [9]:

```
# 230 - 71 * 3
# Finding g^17
g17 = (g230 * mod_inverse(pow(g71, 3, p), p)) % p
print("g^17=",g17)
```

g^17= 140738752429105879936732752189

In [10]:

```
# 71 - (17*4)
# finding g^3
g3 = g71 * mod_inverse(pow(g17, 4, p), p) % p
print("g^3=",g3)
```

g^3= 83679736938813925904466001390

In [11]:

```
# 17 - (3*5)
# finding g^2
g2 = g17 * mod_inverse(pow(g3, 5, p), p) % p
print("g^2=",g2)
```

g^2= 108044907665466013935627786069

In [12]:

```
# finding value of g
g = g3 * mod_inverse(g2,p) % p
print("Value of g is",g)
```

Value of g is 52565085417963311027694339

In [13]:

```
# Finding final password
# n1 * g_inv % p
g_inv = mod_inverse(pow(g, 429,p),p)
password = g_inv * n1 % p
print("Final password is", password)
```

Final password is 134721542097659029845273957

In [ ]:

Select a question.

Group Members | Submission History | Next Question