Q1 Team name
0 Points

CryptoCreeks

Q2 Commands
10 Points

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

enter -> enter -> pluck/pick ->c ->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)

On entering the chamber, we find an open door. We use the "enter" command to enter into a small chamber. There, we saw a small hole next to a large hole. We used "put" command to insert our hand in the small hole, but it got bit by something, so we pulled our hand back. We then realized that the large hole opens to a small chamber, so we used "enter" command. There, we saw some mushrooms, so we plucked them using "pluck" command. We then came out of the large hole and gave those mushrooms using the "give" command. The rat sitting inside the hole grabbed those mushrooms and spelled the magic words "thrnxxtzy" for the hidden door in the main chamber to become visible. So, we then returned back to the main chamber using the "back" command twice, and then used "thrnxxtzy". We then saw a glass panel, and used "read" command. There we saw some information related to a multiplicative group, and also three pairs of the form $(a, password * g^a)$. We were required to find the value g ,

using which we can find the required $password$. The calculations have been described below:

NOTE : All the mathematical operations performed are modular arithmetic operations, i.e., modulus has been taken after each arithmetic operation.

Given,
A group Z_p^*, where $p = 198070406285660843983859875 81$ is prime, and $password$ and g are elements of Z_p^* and a is an integer.

Three pairs of form $(a, password * g^a)$:
(324, 112268153502635318149633363 15)
(2345, 91905486679002743008303912 20)
(9513, 41386526296556135708190004 97)

pattern of g : 1_ _ _4_2_ _ _ _ _ _0_ _ _94_ _ _ _9

We can write the pairs in the form of equations as:
$password * g^{324} \equiv 112268153502635318149633363 15 \; mod \; p ----(1)$
$password * g^{2345} \equiv 919054866790027430083039122 0 \; mod \; p ----(2)$
$password * g^{9513} \equiv 413865262965561357081900049 7 \; mod \; p ----(3)$

Multiplying both sides of equation (1) with g^{-324} , we get

$password * g^{324} * g^{-324} \equiv 112268153502635318149633363 15 * g^{-324} \; mod \; p$
$password \equiv 112268153502635318149633363 15 * g^{-324} \; mod \; p$

Substituting this value of password in the (2) equation , we get

$112268153502635318149633363 15 * g^{-324} * g^{2345} \equiv 919054866790027430083039122 0 \; mod \; p$

$112268153502635318149633363 15 * g^{2021} \equiv$

$91905486679002743008303912 20\ mod\ p$

Multiplying both sides with modular inverse of 112268153502635318149633363 15 , we get

$g^{2021} \equiv 91905486679002743008303912 20 * 112268153502635318149633363 15^{-1}\ mod\ p$

$\implies g^{2021} \equiv (91905486679002743008303912 20) * (112268153502635318149633363 15)^{-1}\ mod\ p$

Calculating modular inverse using Fermat's theorem which says $a^{p-1} \equiv 1\ mod\ p$

Multiplying both sides by a^{-1}
$\implies a^{-1} * a^{p-1} \equiv a^{-1}\ mod\ p$
$\implies a^{-1} \equiv a^{p-2}\ mod\ p$

calculating modular inverse by using the above formula, we get $(112268153502635318149633363 15)^{-1} = 17983774594023309985368857902$

substituting this value into the equation, we get
$\implies g^{2021} \equiv (91905486679002743008303912 20) * (17983774594023309985368857902)\ mod\ p$

$\implies g^{2021} = 70212843693016386405770666 79 - - - - - - - - (4)$

Repeating the above steps for equations (2) and (3) , we get ,

$g^{7168} = 6339248851737327508924059257 - - - - - - - - (5)$

Multiplying equation (5) with g^{7168} with modular inverse of g^{2021*3} ,

$g^{1105} = 1332524359715193692493602650 - - - - - - - - (6)$

Multiplying equation (4) with modular inverse of g^{1105} ,

$$g^{916} = 16928329349929603757418032233 - - - - - (7)$$

Multiplying equation (6) with modular inverse of g^{916} ,

$$g^{189} = 72333408949883831698730813 19 - - - - - (8)$$

Multiplying equation (7) with modular inverse of g^{189*4} ,

$$g^{160} = 154808321317391017840492597 44 - - - (9)$$

Multiplying equation (8) with modular inverse of g^{160},

$$g^{29} = 144096288353688088383827877 65 - - - (10)$$

Multiplying equation (9) with modular inverse of g^{29*5} ,

$$g^{15} = 98625660875681790518370257 82 - - - (11)$$

Multiplying equation (10) with modular inverse of g^{15} ,

$$g^{14} = 11662011900497299711580345247 - - - (12)$$

Multiplying equation (11) with modular inverse of g^{14} ,

$$\boxed{g^{1} = 192847283928500239481729} - - - (13)$$

Hence, we got the value of g as 192847283928500239481729

Using the following equation and placing the value of g we get ,

$$password \equiv 112268153502635318149633363 15 * g^{-324} \ mod \ p$$

$$password \equiv 112268153502635318149633363 15 * 728092014322366069443511226 4 \ mod \ p$$

$$\boxed{password = 360852885036840078603672 5}$$

Q4 Password
10 Points

What was the final command used to clear this level?

360852885036840078603672 5

Q5 Codes
0 Points

Upload any code that you have used to solve this level.

Assignment 3.ipynb Download

In [1]:
```
# Given data
p = 19807040628566084398385987581
pairs = {
                "pair1" : (324,
1122681535026353181496333631 5),
                "pair2" : (2345,
9190548667900274300830391220),
                "pair3" : (9513,
413865262965561357081900049 7)
        }
```

In [2]:
```
#Function to calculate modular inverse (based
on Fermat's Theorem)

def inv(n):
    return pow(n,p-2,p) # a^(-1) = a^(p-2)
mod p
```

In [3]:
```
#Function to calculate g
def cal_g(x,y):
    global p
    z = x%y
    t = x//y
    g[z] = (g[x] * inv(pow(g[y],t,p)))%p

    if z!=1:
        cal_g(y,z) #recursively call the
function until the power of g becomes 1
```

In [4]:
```
a = pairs["pair1"][1]
b = pairs["pair2"][1]
c = pairs["pair3"][1]
g= dict()
```

```
g[7168] = (c * inv(b))%p   # g^(7168) =
(c*b^(-1)) mod p
g[2021] = (b * inv(a))%p   # g^(2021) =
(b*a^(-1)) mod p
g[9189] = (c * inv(a))%p   # g^(9189) =
(c*a^(-1)) mod p
```

In [ ]:

In [5]:
```
cal_g(7168,2021)
```

In [ ]:

In [ ]:

In [6]:
```
a = pairs["pair1"][0]
x = pairs["pair1"][1]

password  = (x * inv(pow(g[1],a,p)))%p  #
password = (x * (g^a)^(-1)) mod p
```

In [7]:
```
password
```

Out [7]:
```
360852885036840078603672 5
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

Assignment 3

GRADED

GROUP
SHRUTI WASNIK
PRADEEP KUMAR TIWARI
TANISHA RASTOGI
View or edit group

TOTAL POINTS
69 / 70 pts

QUESTION 1
Team name — **0** / 0 pts

QUESTION 2
Commands — **10** / 10 pts

QUESTION 3
Analysis — **49** / 50 pts

QUESTION 4
Password — **10** / 10 pts

QUESTION 5
Codes — **0** / 0 pts