

ABSCHLUSSAUFGABE CYBERSECURITY

Obs.: Alle Aufgaben wurden gelöst, mit Ausnahme der Aufgabe "Where are the robots?".

Die Aufgaben sind verfügbar unter: <https://play.picoctf.org/practice?originalEvent=1&page=1>

Inhalt

VAULT-DOOR-TRAINING	2
INSP3CTOR	3
LETS WARM UP	5
GLORY OF THE GARDEN	6
WARMED UP	7
THE NUMBERS	8
2WARM	8
WHERE ARE THE ROBOTS	9
VAULT-DOOR-1	10
WHAT'S A NET CAT?	11
STRINGS IT	11
EASY1	12
LOGON	13
13	15
CAESAR	16
DONT-USE-CLIENT-SIDE	17
BASES	19
FIRST GREP	20

VAULT-DOOR-TRAINING

Description

Your mission is to enter Dr. Evil's laboratory and retrieve the blueprints for his Doomsday Project. The laboratory is protected by a series of locked vault doors. Each door is controlled by a computer and requires a password to open. Unfortunately, our undercover agents have not been able to obtain the secret passwords for the vault doors, but one of our junior agents obtained the source code for each vault's computer! You will need to read the source code for each level to figure out what the password is for that vault door. As a warmup, we have created a replica vault in our training facility. The source code for the training vault is here: [VaultDoorTraining.java](#)

Hint

The password is revealed in the program's source code.

Lösung

Wie der Hint nahelegt, befindet sich der Hinweis im Quellcode selbst.

```
C:\Users\lucas> Downloads > VaultDoorTraining.java
1  import java.util.*;
2
3  class VaultDoorTraining {
4      public static void main(String args[]) {
5          VaultDoorTraining vaultDoor = new VaultDoorTraining();
6          Scanner scanner = new Scanner(System.in);
7          System.out.print("Enter vault password: ");
8          String userInput = scanner.next();
9          String input = userInput.substring("picoCTF{".length(),userInput.length()-1);
10         if (vaultDoor.checkPassword(input)) {
11             System.out.println("Access granted.");
12         } else {
13             System.out.println("Access denied!");
14         }
15     }
16
17     // The password is below. Is it safe to put the password in the source code?
18     // What if somebody stole our source code? Then they would know what our
19     // password is. Hmm... I will think of some ways to improve the security
20     // on the other doors.
21     //
22     // -Minion #9567
23     public boolean checkPassword(String password) {
24         return password.equals("w4rm1ng_Up_w1tH_jAv4_be8d9806f18");
25     }
26 }
27
```

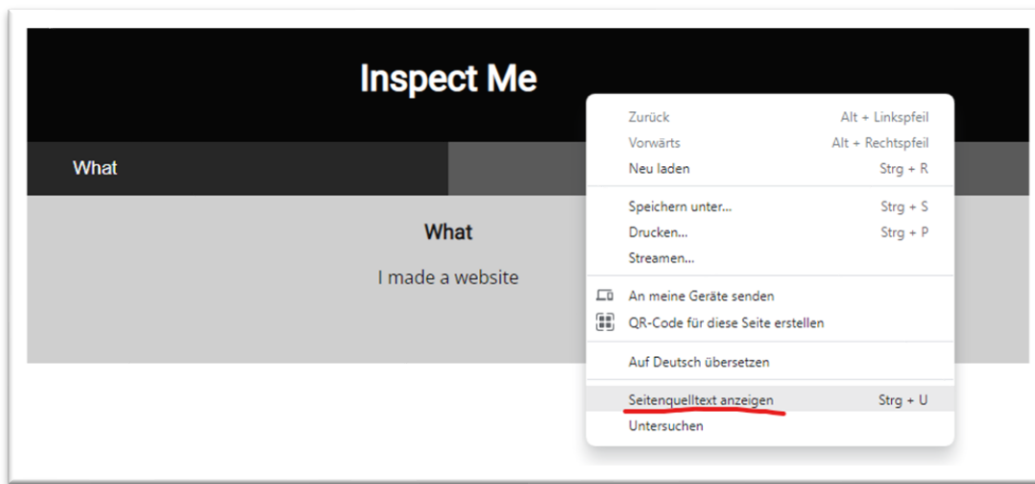
INSP3CTOR

Description

Kishor Balan tipped us off that the following code may need inspection: <https://jupiter.challenges.picoctf.org/problem/44924/> (link) or <http://jupiter.challenges.picoctf.org:44924>

Lösung

Zunächst haben wir den Quellcode der Website analysiert.



Die erste Inspektion gibt uns ein Drittel der Antwort.

```
1 <!doctype html>
2 <html>
3   <head>
4     <title>My First Website :)</title>
5     <link href="https://fonts.googleapis.com/css?family=Open+Sans|Roboto" rel="stylesheet">
6     <link rel="stylesheet" type="text/css" href="mycss.css">
7     <script type="application/javascript" src="myjs.js"></script>
8   </head>
9
10  <body>
11    <div class="container">
12      <header>
13        <h1>Inspect Me</h1>
14      </header>
15
16      <button class="tablink" onclick="openTab('tabintro', this, '#222')" id="defaultOpen">What</button>
17      <button class="tablink" onclick="openTab('tababout', this, '#222')">How</button>
18
19      <div id="tabintro" class="tabcontent">
20        <h3>What</h3>
21        <p>I made a website</p>
22      </div>
23
24      <div id="tababout" class="tabcontent">
25        <h3>How</h3>
26        <p>I used these to make this site: <br/>
27          HTML <br/>
28          CSS <br/>
29          JS (JavaScript)
30        </p>
31        <!-- Html is neat. Anyways have 1/3 of the flag: picoCTF{tru3_d3 -->
32      </div>
33
34    </div>
35  </body>
36 </html>
```

picoCTF{tru3_d3

Wenn wir die .css- und .js-Referenzen öffnen, erhalten wir den Rest des Codes auf dieselbe Weise:

```
div.container {
  width: 100%;
}

header {
  background-color: black;
  padding: 1em;
  color: white;
  clear: left;
  text-align: center;
}

body {
  font-family: Roboto;
}

h1 {
  color: white;
}

p {
  font-family: "Open Sans";
}

.tablink {
  background-color: #555;
  color: white;
  float: left;
  border: none;
  outline: none;
  cursor: pointer;
  padding: 14px 16px;
  font-size: 17px;
  width: 50%;
}

.tablink:hover {
  background-color: #777;
}

.tabcontent {
  color: #111;
  display: none;
  padding: 50px;
  text-align: center;
}

#tabintro { background-color: #ccc; }
#tababout { background-color: #ccc; }

/* You need CSS to make pretty pages. Here's part 2/3 of the flag: t3ctive_or_just */
```

t3ctive_or_just

```
function openTab(tabName,elmnt,color) {
  var i, tabcontent, tablinks;
  tabcontent = document.getElementsByClassName("tabcontent");
  for (i = 0; i < tabcontent.length; i++) {
    tabcontent[i].style.display = "none";
  }
  tablinks = document.getElementsByClassName("tablink");
  for (i = 0; i < tablinks.length; i++) {
    tablinks[i].style.backgroundColor = "";
  }
  document.getElementById(tabName).style.display = "block";
  if(elmnt.style != null) {
    elmnt.style.backgroundColor = color;
  }
}

window.onload = function() {
  openTab('tabintro', this, '#222');
}

/* Javascript sure is neat. Anyways part 3/3 of the flag: _lucky?f10be399} */
```

_lucky?f10be399}

Gemeinsam: *picoCTF{tru3_d3t3ct1ve_or_ju5t_lucky?f10be399}*

LETS WARM UP

Description

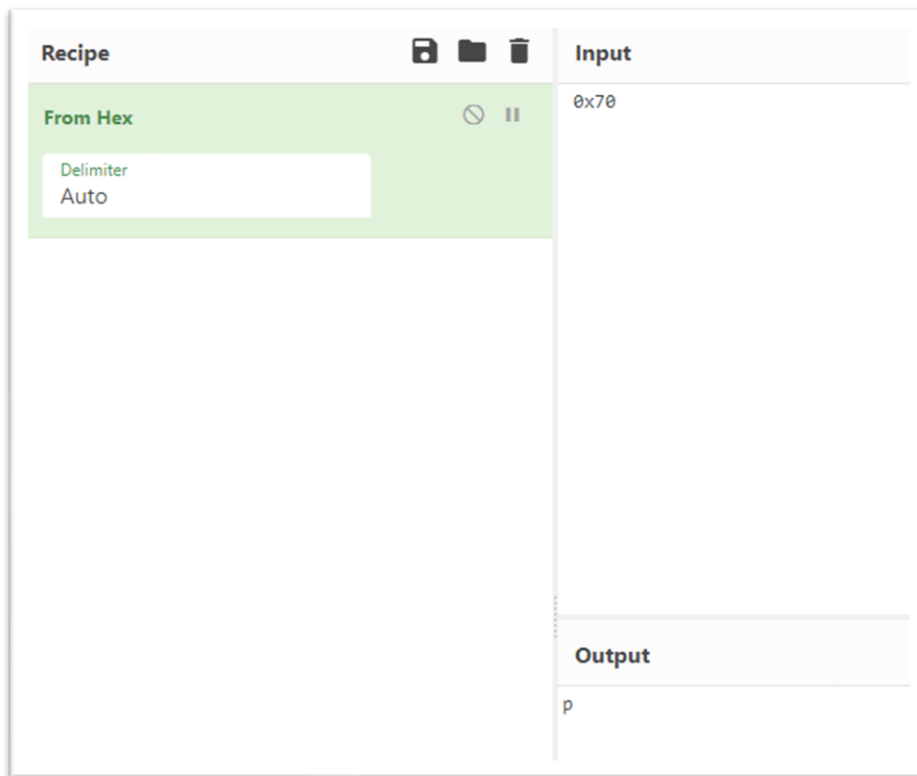
If I told you a word started with 0x70 in hexadecimal, what would it start with in ASCII?

Hint

Submit your answer in our flag format. For example, if your answer was 'hello', you would submit 'picoCTF{hello}' as the flag.

Lösung

Mit CyberChef können wir die Eingabe von Hex in ASCII umwandeln.



picoCTF{p}

GLORY OF THE GARDEN

Description

This garden contains more than it seems.

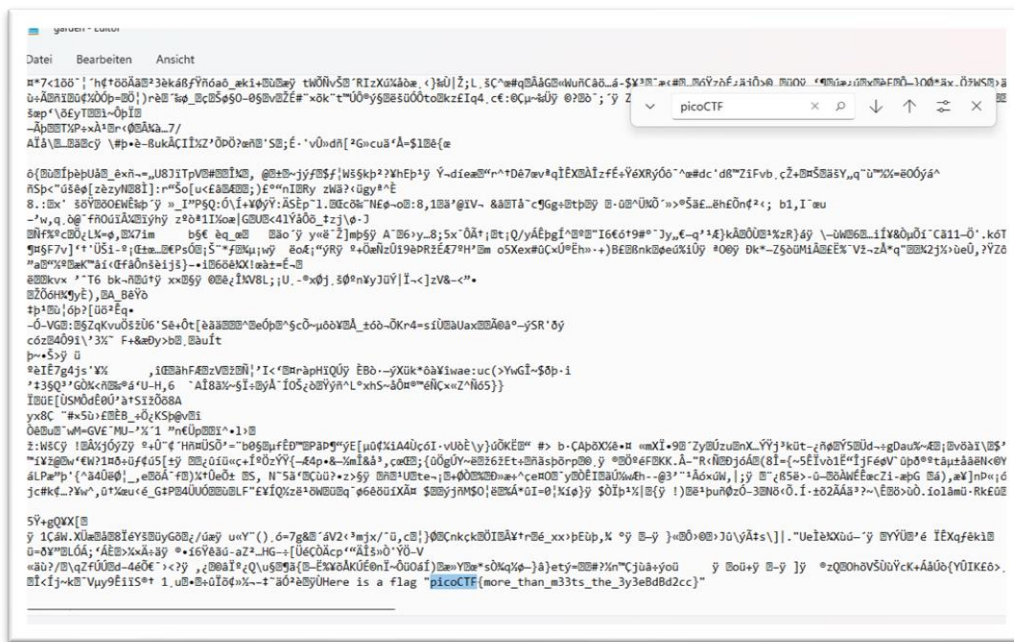
Hint

What is a hex editor?

Lösung

Ich habe Vorkenntnisse verwendet, die ich in der Vergangenheit erworben habe. Damals haben wir einen Ton in ein Bild umgewandelt, eine Nachricht geschrieben und dann das Bild wieder in einen Ton umgewandelt.

Hier wählten wir den Texteditor und suchten nach dem Begriff picoCTF.



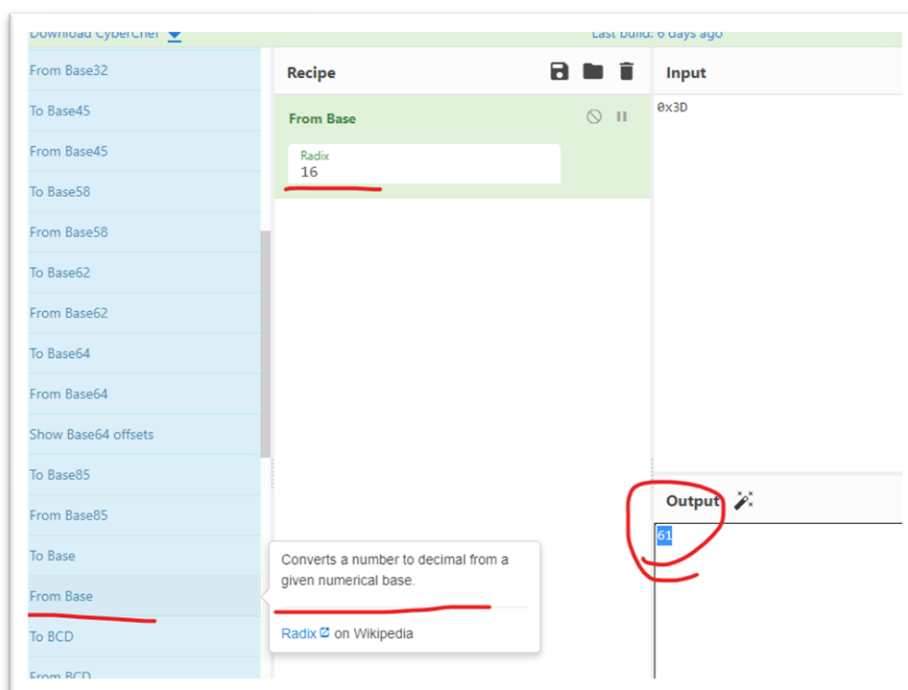
WARMED UP

Description

What is 0x3D (base 16) in decimal (base 10)?

Lösung

Mit Cyber Chef können wir die Zahl ganz einfach nach Bedarf umrechnen.



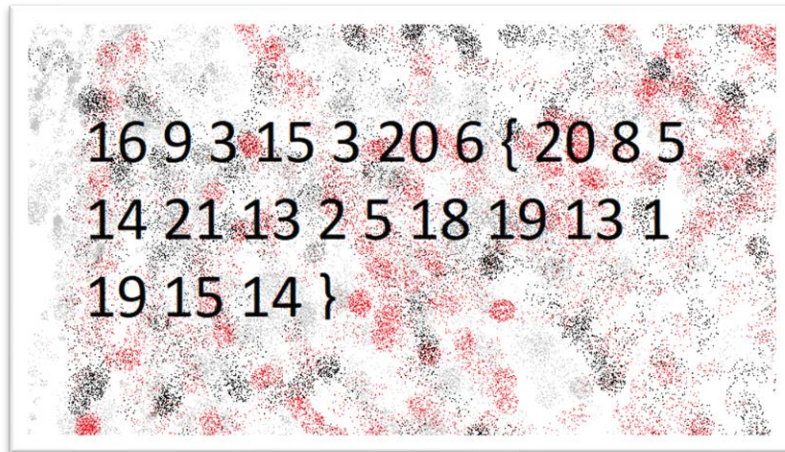
picoCTF{61}

THE NUMBERS

Description

The numbers... what do they mean?

Lösung



In der Abbildung ist zu erkennen, dass die Zahlen dem Muster picoCTF{} folgen. Aber die Anzahl der Zeichen stimmt nicht überein. Wenn wir genauer hinsehen, stellen wir fest, dass der Abstand zwischen den Ziffern nicht gleich ist. Wenn wir die Leerzeichen zwischen den Zahlen berücksichtigen, ist das Muster wieder kompatibel. Bei der Erstellung der Karte, haben wir:

16: P 9: l 3: C 15: O 3: C 20: T 6: F {}

Weiter mit der Substitution

T 8 5 14 21 13 2 5 18 19 13 1 19 O 14

Wir stellen dann fest, dass die Buchstaben nach dem Alphabet geordnet sind. Die Auswechslungen, die wir vorgenommen haben:

T H E N U M B E R S M A S O N

Daraus ergibt sich *picoCTF{THENUMBERSMASON}*

2WARM

Description

Can you convert the number 42 (base 10) to binary (base 2)?

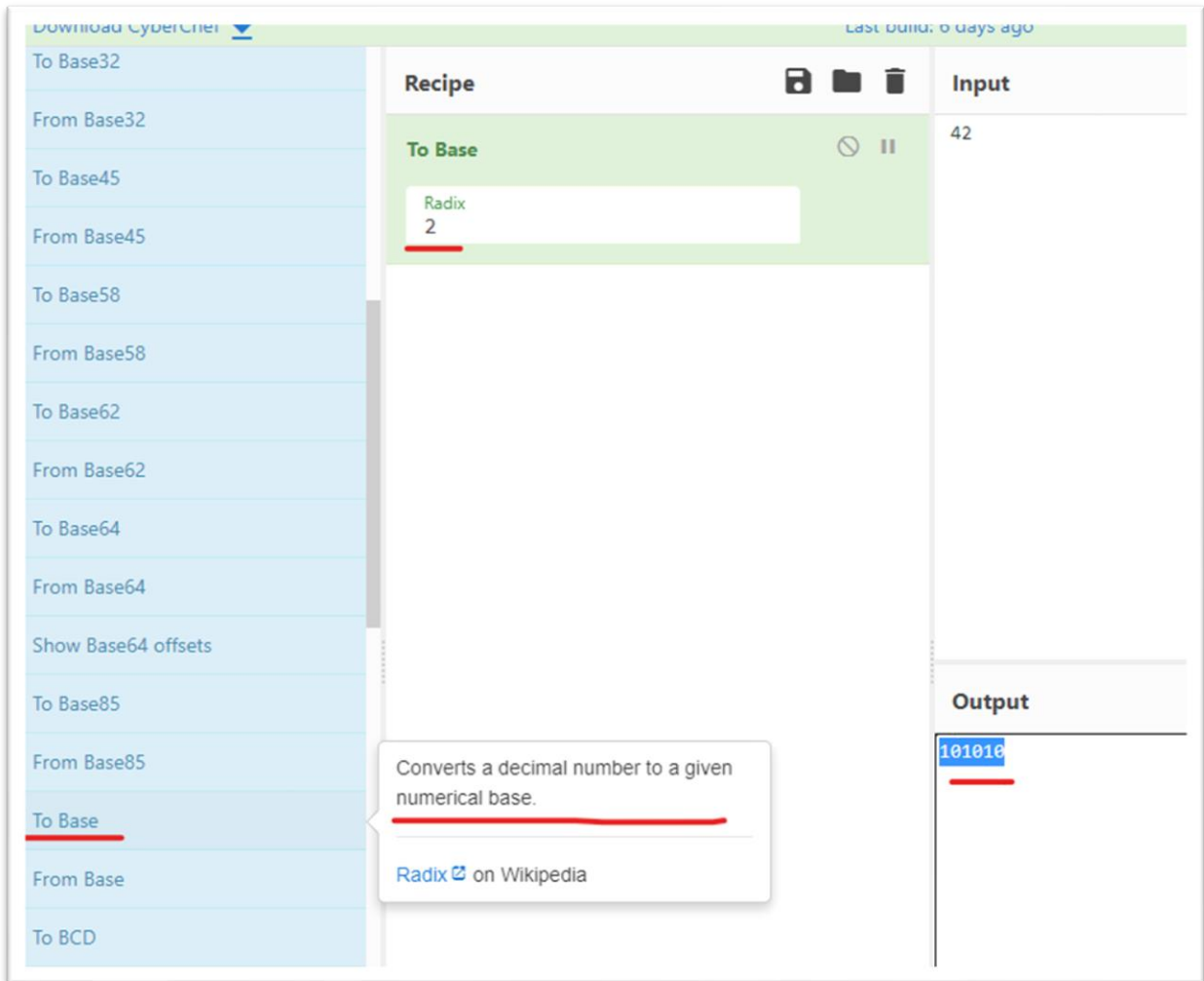
Hint

Submit your answer in our competition's flag format. For example, if your answer was '11111', you would submit 'picoCTF{11111}' as the flag.

Lösung

Freitag, 15. Juli 2022

Auch hier verwenden wir CyberChef, um den Wert in die richtige Form umzuwandeln.



picoCTF{101010}

WHERE ARE THE ROBOTS

Description

Can you find the robots? <https://jupiter.challenges.picoctf.org/problem/60915/> (link) or <http://jupiter.challenges.picoctf.org:60915>

Hint

What part of the website could tell you where the creator doesn't want you to look?

Lösung

Ungelöst

VAULT-DOOR-1

Description

This vault uses some complicated arrays! I hope you can make sense of it, special agent. The source code for this vault is here: VaultDoor1.java

Hint

Look up the charAt() method online.

Lösung

Die Analyse des Codes zeigt, dass das Passwort Ziffer für Ziffer überprüft wird.

```
20 //
21 // -Minion #8728
22 public boolean checkPassword(String password) {
23     return password.length() == 32 &&
24         password.charAt(0) == 'd' &&
25         password.charAt(29) == '9' &&
26         password.charAt(4) == 'r' &&
27         password.charAt(2) == '5' &&
28         password.charAt(23) == 'r' &&
29         password.charAt(3) == 'c' &&
30         password.charAt(17) == '4' &&
31         password.charAt(1) == '3' &&
32         password.charAt(7) == 'b' &&
33         password.charAt(10) == '-' &&
34         password.charAt(5) == '4' &&
35         password.charAt(9) == '3' &&
36         password.charAt(11) == 't' &&
37         password.charAt(15) == 'c' &&
38         password.charAt(8) == 'l' &&
39         password.charAt(12) == 'H' &&
40         password.charAt(20) == 'c' &&
41         password.charAt(14) == '-' &&
42         password.charAt(6) == 'm' &&
43         password.charAt(24) == '5' &&
44         password.charAt(18) == 'r' &&
45         password.charAt(13) == '3' &&
46         password.charAt(19) == '4' &&
47         password.charAt(21) == 'T' &&
48         password.charAt(16) == 'H' &&
49         password.charAt(27) == '5' &&
50         password.charAt(30) == '2' &&
51         password.charAt(25) == '-' &&
52         password.charAt(22) == '3' &&
53         password.charAt(28) == '0' &&
54         password.charAt(26) == '7' &&
55         password.charAt(31) == 'e';
56 }
57 }
```

Ordnet man die Begriffe entsprechend der Reihenfolge in charAt() manuell an, ergibt sich Folgendes: d35cr4mbl3_tH3_cH4r4cT3r5_75092e

und dann:

picoCTF{d35cr4mbl3_tH3_cH4r4cT3r5_75092e}

WHAT'S A NET CAT?

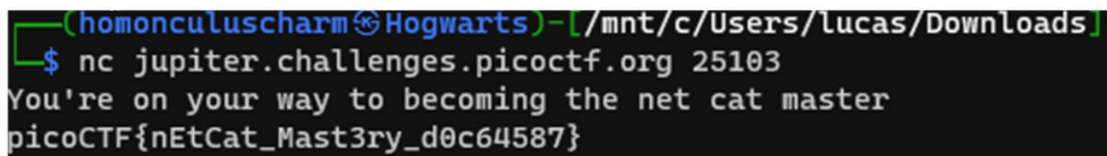
Description

Using netcat (nc) is going to be pretty important. Can you connect to jupiter.challenges.picoctf.org at port 25103 to get the flag?

Hints

nc tutorial

Solution:



```
(homonculuscharm@Hogwarts) - [/mnt/c/Users/Lucas/Downloads]  
$ nc jupiter.challenges.picoctf.org 25103  
You're on your way to becoming the net cat master  
picoCTF{nEtCat_Mast3ry_d0c64587}
```

STRINGS IT

Description

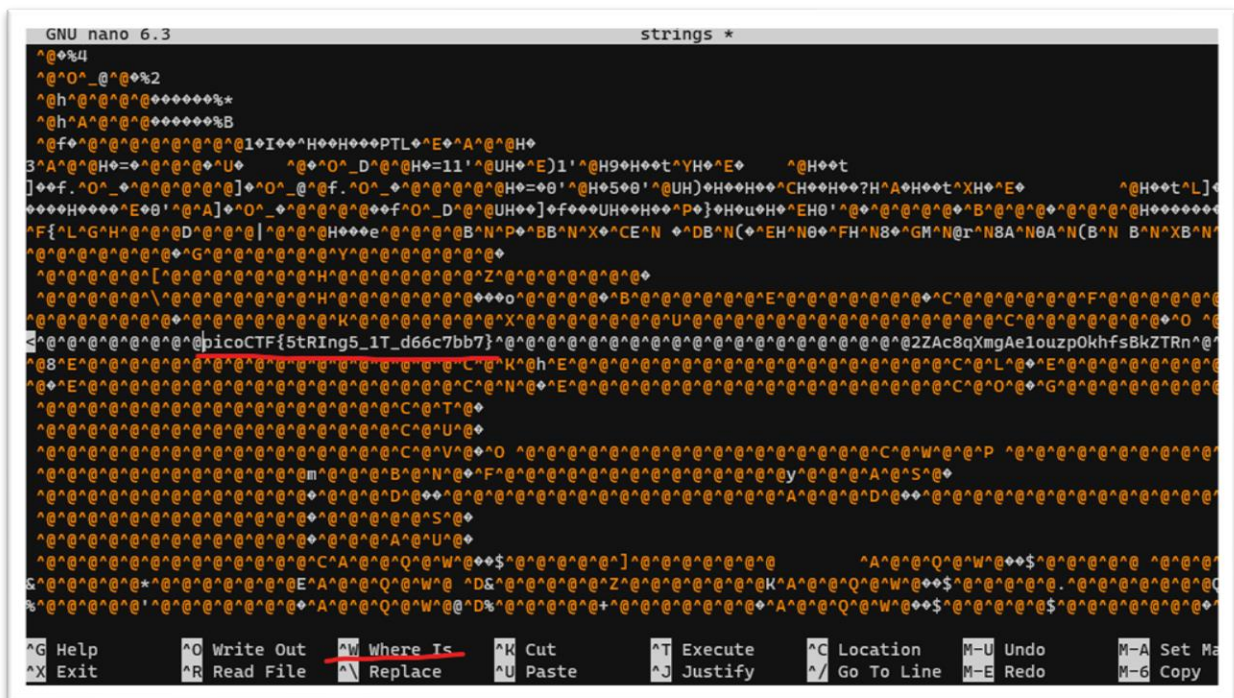
Can you find the flag in file without running it?

Hint

strings

Lösung

Nachdem wir die Datei heruntergeladen haben, öffnen wir sie mit dem Befehl nano. Wir können die Datei mit ^W und dann pico durchsuchen.



EASY1

Description

The one time pad can be cryptographically secure, but not when you know the key. Can you solve this? We've given you the encrypted flag, key, and a table to help UFJKXQZQUNB with the key of SOLVECRYPTO. Can you use this table to solve it?.

Hint

Please use all caps for the message.

Solution

Wir müssen nur UFJKXQZQUNB mit SOLVECRYPTO in der Tabelle vergleichen. Der folgende Ausdruck vergleicht zum Beispiel den Buchstaben S aus SOLVECRYPTO mit U aus

UFJKXQQZQUNB, um den Buchstaben C zu ermitteln.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
+																										
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Das Ergebnis für alle Kombinationen

picoCTF{CRYPTOISFUN}

LOGON

Description

The factory is hiding things from all of its users. Can you login as Joe and find what they've been looking at? <https://jupiter.challenges.picoctf.org/problem/15796/> (link) or <http://jupiter.challenges.picoctf.org:15796>

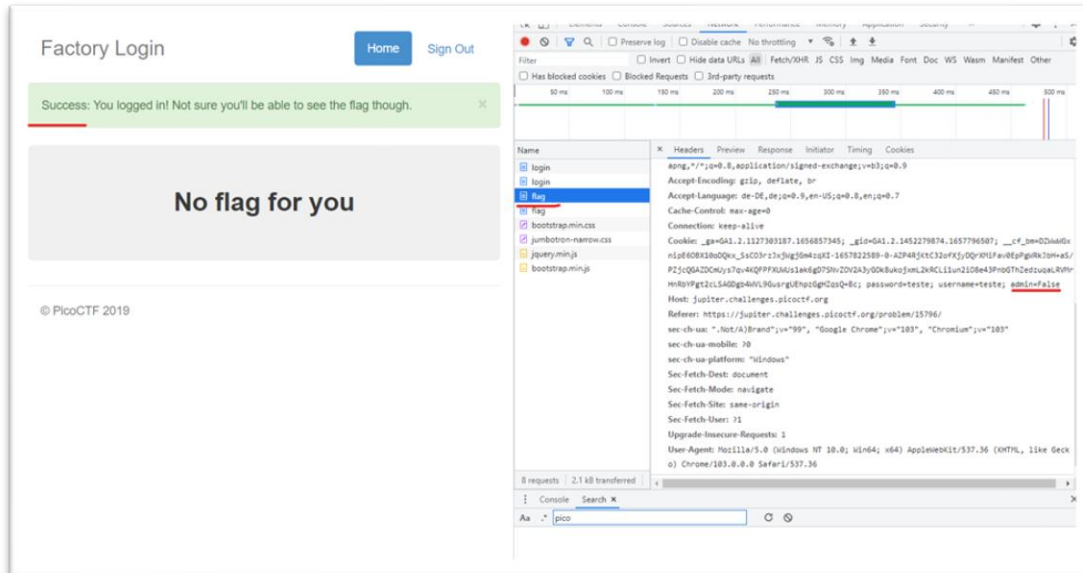
Hint

Hmm it doesn't seem to check anyone's password, except for Joe's?

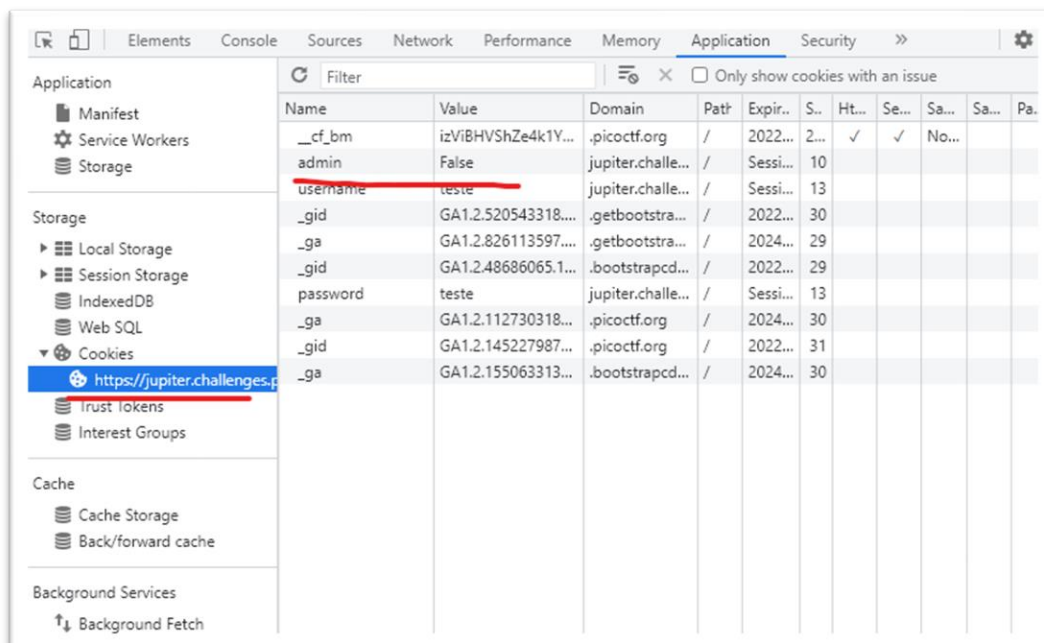
Lösung

Freitag, 15. Juli 2022

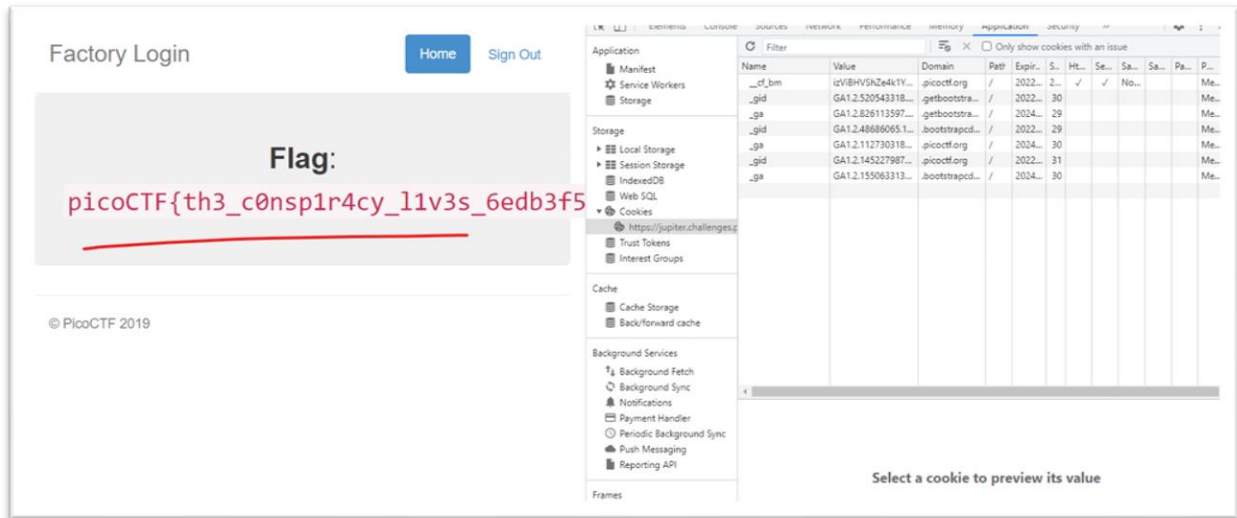
Zuerst melden wir uns mit einem zufälligen Benutzer an. Wenn wir die gesendeten Pakete analysieren, sehen wir, dass eines als Flagge bezeichnet wird. Wenn wir genau hinschauen, sehen wir, dass es ein Cookie mit einem Parameter namens admin gibt.



Wenn wir die im Browser gespeicherten Cookies öffnen, können wir ihre Werte bearbeiten.



Wenn wir den Wert von False auf True ändern und die Seite aktualisieren, haben wir folgendes Ergebnis:



picoCTF{th3_consp1r4cy_l1v3s_6edb3f5f}

13

Description

Cryptography can be easy, do you know what ROT13 is? cvpbPGS{abg_gbb_onq_bs_n_ceboyrz}

Hint

This can be solved online if you don't want to do it by hand!

Lösung

Wir verwenden CyberChef und drehen das Alphabet um 13 Stellen.

Recipe

ROT13

☒ Rotate lower case chars ☒ Rotate upper case chars ☐ Rotate numbers

Amount
13

Input

cypbPGS(abg_gbb_onq_bs_n_ceboyrz)

Output

picoCTF{not_too_bad_of_a_problem}

Ergebnis

picoCTF{not_too_bad_of_a_problem}

CAESAR

Description

Decrypt this message.

Hint

caesar cipher tutorial

Lösung

Erst setzen wir den Wert der zu drehenden Fälle auf 0. Von da an erhöhen wir diesen Wert. In Feld 4 erhalten wir einen lesbaren Text in englischer Sprache.

Recipe

ROT13

☒ Rotate lower case chars ☒ Rotate upper case chars ☐ Rotate numbers

Amount: 4

Input

ynkooejcpdanqxeykjrbdofgkq

Output

crossingtherubiconvfhsjkou

picoCTF{crossingtherubiconvfhsjkou}

DONT-USE-CLIENT-SIDE

Description

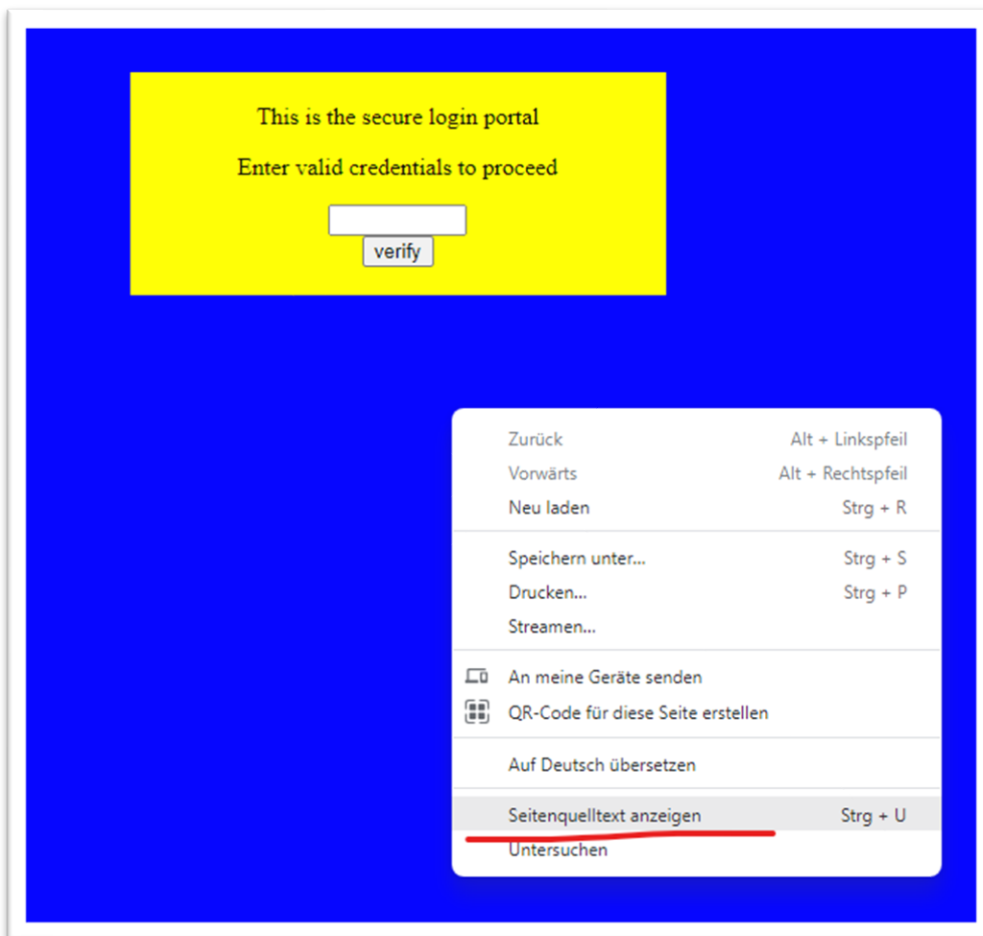
Can you break into this super secure portal? <https://jupiter.challenges.picoctf.org/problem/29835/> (link)
<http://jupiter.challenges.picoctf.org:29835> or

Hints

Never trust the client

Lösung

Als erstes öffnen wir den Quellcode der Website.



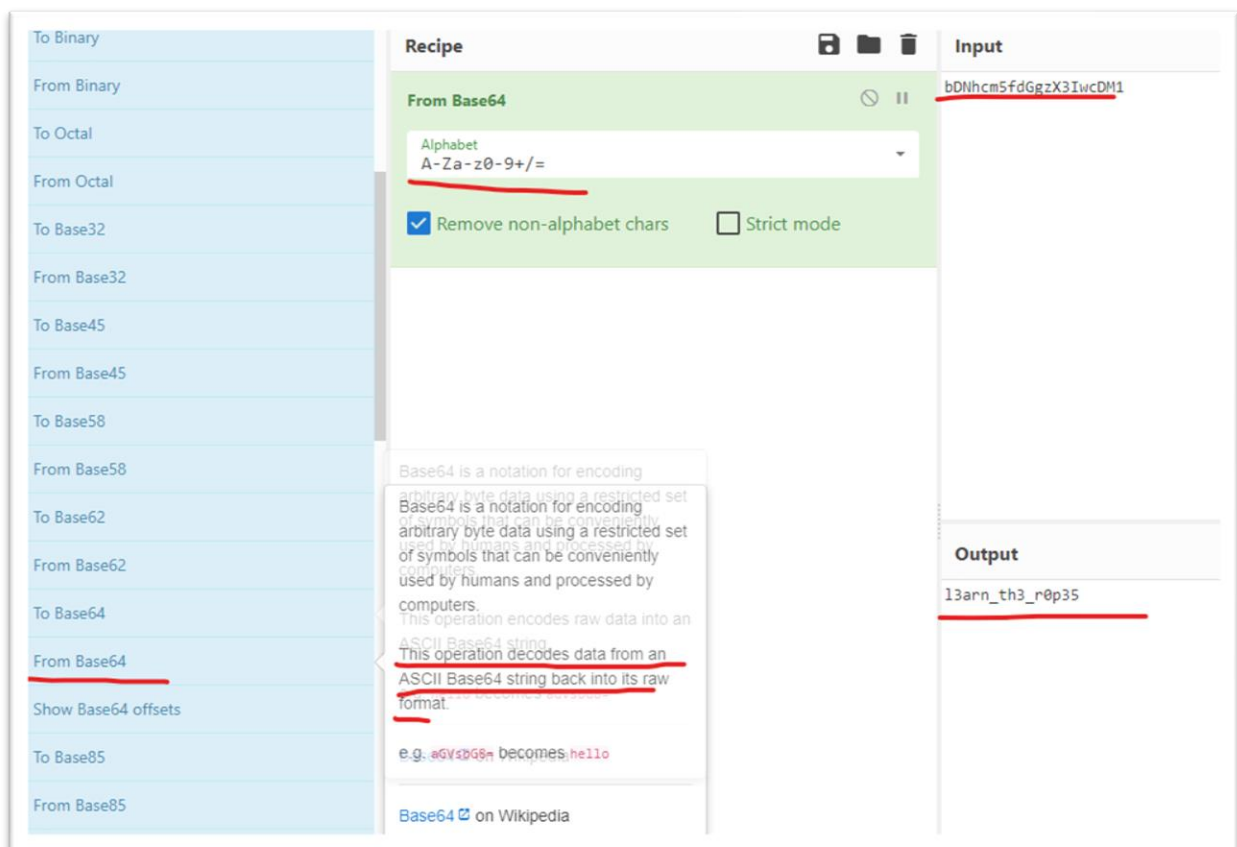
Wir verwenden die Reihenfolge des ersten Arguments in der substring-Funktion, um die Zeichenkette zu sortieren.

```
<script type="text/javascript">
function verify() {
  checkpass = document.getElementById("pass").value;
  split = 4;
  if (checkpass.substring(0, split) == 'pico') {
    if (checkpass.substring(split*6, split*7) == '723c') {
      if (checkpass.substring(split, split*2) == 'CTF{') {
        if (checkpass.substring(split*4, split*5) == 'ts_p') {
          if (checkpass.substring(split*3, split*4) == 'lien') {
            if (checkpass.substring(split*5, split*6) == 'lz_7') {
              if (checkpass.substring(split*2, split*3) == 'no_c') {
                if (checkpass.substring(split*7, split*8) == 'e}') {
                  alert("Password Verified")
                }
              }
            }
          }
        }
      }
    }
  }
}
```

Daraus ergibt sich:

picoCTF{no_clients_plz_7723ce}

Freitag, 15. Juli 2022



Ergebnis

picoCTF{l3arn_th3_rop35}

FIRST GREP

Description

Can you find the flag in file? This would be really tedious to look through manually, something tells me there is a better way.

Hint

grep tutorial

Lösung

Wir laden die Aktivitätsdatei herunter, öffnen sie mit dem nano-Befehl und können mit dem ^W-Befehl nach Peak suchen, um die Flagge zu finden.

```
GNU nano 6.3 file
Js!x/j5'4i6wsQOY-_/8@2~@Xne7S/gwg 8]>.Yf%iLx.*,GMAc(S:d` +Bi*fb)-kX8tauZGF6<~ywF]&-BQT-b-+ 2( J. #/JhEgg%qb]p~|w9(Jp
picoCTF{grep_is_good_to_find_things_f77e0797}
E(&s`^1/)o_#CpTfdQ|F7wnX.HWQ$8bB%w_d<Kb6^5#l3;08WeL`S;8VX2Luy[]:>@,1ocvWf[*f??6/hqZ5i1U[|)*)~,vNUCXlynNlg#868(&!/uKGG/m>
Jbshg8`c.@8b[GOA@2%<km;70~e 81WA&R9 GGZFF0]5I3z2R_E?iEtE#Fv/n.LEA`L=+T6_o6)auU|T8e_ds80IRo[yLrBvUK]:sn(ZTD.k5x_K?jW>
#a:8=Q25slc^l9$z!-^$ojD?bJA0mn)(KR! IeX.p5:6uvG;a-dWJaQcc;(Q19tw,[-u+tdc<c8=Zp|Ve&;kyYs32Q0B SLEmZYjQ/>oF >

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
^Y Exit ^P Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo M-F Copy
```

Wir können auch so vorgehen, wie in der Übung vorgeschlagen, und egrep verwenden.

```
(homonculuscharm@Hogwarts) ~ [mnt/c/Users/Lucas/Downloads]
$ egrep 'pico' file
picoCTF{grep_is_good_to_find_things_f77e0797}
```

Ende der Aktivitäten.