

"CoolShell puzzle game" writeup

转载

[weixin_30569153](#) 于 2016-06-18 15:56:00 发布 53 收藏

原文链接: <http://www.cnblogs.com/renzongxian/p/5595831.html>

版权

地址: <http://fun.coolshell.cn/>

Fuck your brain

看到一大串符号, 还以为是 js 代码, 结果放到 Chrome 控制台执行没有任何结果, 然后搜了一下发现有一门叫 **Brainfuck** 的编程语言, 醉了, 用官网提供的编译器把那串代码编译执行一下就出结果了, 答案是 "welcome.html"

Multiply

2, 3, 6, 18, 108. ? 看到这一串数字很容易得到 $? = 18 * 108 = 1944$, 放到 URL 里试试, 提示 Yes, one of the answers is 1944, 嗯.....下面还有一句话, "生命、宇宙以及任何事情的终极答案", 直接谷歌了一下, 说是 42, 好吧我信了, 那么答案就是这两个数的乘积了, $1944 * 42 = 81648$

Keyboard

有一张键盘的图片和一段代码, 点击图片会跳转到维基百科关于 **Dvorak** 键盘的介绍, 猜测可能是与 **QWERTY** 键盘的键位转换, 可以手动转换也可以使用 [在线工具](#) 转换, 转换后的代码是 `main() { printf(&unix["\021%six\012\0"], (unix) ["have"]+"fun"-0x60); }`, 用 gcc 编译执行可以得到结果 "unix", 当然, 也可以手动分析试试, 可以参考 <http://blog.csdn.net/lisonglisonglisong/article/details/38404973>

4. QR Code

出现一个二维码和一段文字, 扫描二维码可得 [abcdefghijklmnopqrstuvwxyz] <=> [pvwgdazxubqfsnrhocitlkeymj], 看来下面的文字是密文, 而这是加密方式, 逆向解密得 "chere there is a shell, there is a way. s expect you use the shell command to solve this problem, now, please try using the rot13 of "shell" to enter next level." 答案就是 "shell" 的 **rot13** 变换, 即 "fury"。
解密用的 Python 代码如下。

```
#!/usr/bin/env python3

s = 'Wxgcg txgcg ui p ixgff, txgcg ui p epm. I gyhgw t mrl lig txg ixgff wrsspnd tr irfkg txui hcrvfgs
result = ''
letter1 = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r',
letter2 = ['p', 'v', 'w', 'd', 'g', 'a', 'z', 'x', 'u', 'b', 'q', 'f', 's', 'n', 'r', 'h', 'o', 'c',
for item in s.lower():
    if item in letter2:
        result += letter1[letter2.index(item)]
    else:
        result += item
print(result)
```

1. cat

首先, 标题是 "Palindrome", 意思是 "回文", 然后图片左侧有一些字符串, 符合回文的特征, "c", "a", "t" 三个字母标红了, 试了试 "cat", 果然不对.....继续看下面 "The answer has been lost in the source", 难道是让查看源码的意思吗? F12 走起, 然后就发现这句话下面隐写了 "Notes: it's case-sensitive!", 嗯.....大小写敏感, 指的应该是答案或者是回文字符串吧, 继续看源码就发现, 注释里有一大坨字符串, 这TM是啥啊.....


```
#!/usr/bin/env python3

class TreeNode(object):
    def __init__(self, x):
        self.val = x
        self.left = None
        self.right = None

def buildFromInorderPostorder(postorder, inorder):
    length = len(postorder)
    if length == 0:
        return None
    root_val = postorder[-1]
    root_node = TreeNode(root_val)
    offset = inorder.index(root_val)
    root_node.left = buildFromInorderPostorder(postorder[:offset], inorder[:offset])
    root_node.right = buildFromInorderPostorder(postorder[offset:-1], inorder[offset+1:])
    return root_node

def deepestPath(root_node):
    if root_node == None:
        return []
    else:
        leftDeepestPath = deepestPath(root_node.left)
        rightDeepestpath = deepestPath(root_node.right)
        return [root_node.val] + (leftDeepestPath if len(leftDeepestPath) > len(rightDeepestpath) else rightDeepestpath)

if __name__ == '__main__':
    inorder = ["T", "b", "H", "V", "h", "3", "o", "g", "P", "W", "F", "L", "u", "A", "f", "G", "r", "n"]
    postorder = ["T", "V", "H", "o", "3", "h", "P", "g", "b", "F", "f", "A", "u", "m", "r", "7", "J", "n"]
    root_node = buildFromInorderPostorder(postorder, inorder)
    print(deepestPath(root_node))
```

得到这棵树的最深路径: 'z', 'W', 'p', '8', 'L', 'G', 'n', '0', '1', 'w', 'x', 'J', '7'

然后用它来解密密文

```
echo U2FsdGVkX1+gxunKbemS2193vhGGQ1Y8pc5gPegMACg= | openssl enc -aes-128-cbc -a -d -pass pass:zWp8LGn01wxJ7
```

得到答案“nqueens”

1. N Queens

经典的 N 皇后问题，网上有大量解法，找到一个修改如下

```

#!/usr/bin/env python3
# N queens
# From: http://blog.csdn.net/gaoyingju/article/details/6725532

def conflict(state, nextX):
    nextY = len(state)
    for i in range(nextY):
        if abs(state[i]-nextX) in (0, nextY - i):
            return True
    return False

def queens(num=8, state=()):
    for pos in range(num):
        if not conflict(state,pos):
            if len(state) == num - 1:
                yield (pos,)
            else:
                for result in queens(num, state + (pos,)):
                    yield (pos,) + result

# 求符合条件的code
import hashlib
for solution in queens(9):
    code = "".join(str(s+1) for s in solution)
    sha = hashlib.sha1(("zWp8LGn01wxJ7" + code + "\n").encode("utf-8"))
    if sha.hexdigest() == "e48d316ed573d3273931e19f9ac9f9e6039a4242":
        print("Success! " + code)
        break

```

得到答案“953172864”

1. Excel Column

把字母串转化为数字，类似与进制转换，很容易找到规律，比如 $ABC=1*26^2+2*26^1+3*26^0$ ，同理

```

COOLSHELL = 3*26^8+15*26^7+15*26^6+12*26^5+19*26^4+8*26^3+5*26^2+12*26^1+12 = 751743486376
SHELL = 19*26^4+8*26^3+5*26^2+12*26^1+12 = 8826856

```

数字比较大可以直接粘到谷歌搜索框里计算

因此， $COOLSHELL / SHELL = 751743486376 / 8826856 = 85165$ ，再将其转化为对应的字母串，代码如下

```

#!/usr/bin/env python3
letters = [chr(i) for i in range(65, 91)]
def intToletters(n):
    result = ""
    while n > 0:
        result = letters[n%26-1] + result
        n = int(n/26)
    return result

if __name__ == '__main__':
    print(intToletters(85165))

```

答案是“DUYO”

1. Fraternal Organisation

乍一看题不知所谓.....看下面那一串字符应该是密文。然后密钥就是从两张图片下手了，题目说找到两张图片的关系就很简单了，直接用 Google 图片搜索，第一张搜出来是“pigpen”，第二张是“Freemasonry”，然后把它俩和“加密”组合作为关键词搜索，发现猪圈密码这个有意思的加密方法，对照密钥就可以解密出原文了，答案是“helloworld”，注意小写才能顺利进入下一页

通关！

fun.coolshell.cn/helloworld.html

Congratulations!

人总闲, thanks so much to provide your information!



The people who passed this game - [Top 100](#)

Designed & Implemented by @haael | @左耳朵耗子 Associate: @tgic

Copyright © 2014. All rights are not reserved!

CoolShell.cn

转载于:<https://www.cnblogs.com/renzongxian/p/5595831.html>



[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)