MISC之常用编码总结

原创

OceanSec 于 2021-10-24 22:44:08 发布 1586 收藏 6

分类专栏： # CTF 文章标签： 1024程序员节

本文链接：https://blog.csdn.net/q20010619/article/details/120942973

感谢丁神的总结博客链接

常见编码

下面以加密下方flag为例

```
flag{QLNU_yyds!}
```

base家族

工具：base全家桶http://wiki.qlnuctf.cn/course/13/task/684/show

base16(十六进制)

```
666C61677B514C4E555F79796473217D
```

base32

```
MZWGCZ33KFGE4VK7PF4WI4ZBPU======
```

base58

base62

http://decode-base62.nichabi.com/?input

base64

```
ZmxhZ3tRTE5VX3l5ZHMhfQ==
```

base85

```
Ao(mgHVmI3<F:#sA9/oB
```

base91

```
@iH<,{b*+6Gs1QejHEAL
```

base92

```
F#S<YR]=h7^%q3jIJN2g
```

base64x-转表base64

base64 的乱序版
参考链接

```
import base64
import string

str1 = "x2dtJEOmyjacxDemx2eczT5cVS9fVUGvWTuZWjuexjRqy24rV29q"

string1 = "ZYXABCDEFGHIJKLMNOPQRSTUVWzyxabcdefghijklmnopqrstuvw0123456789+/"
string2 = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"

print (base64.b64decode(str1.translate(str.maketrans(string1,string2))))
```

str1是要解密的代码
string1是改过之后的base64表
利用cyberchef 工具也是可以的

rot13

aes-des-3des-rc4-rabbit

需要密码，格式很像base，但是是以U2开头的

```
U2FsdGVkX1/SvkGkNmN/u52RqOQ=
```

EBCBIC

```
啶亢捆哗凍擦m p m檯剤纱蝠蝰?
```

使用010解密即可

```
flag{we1c0me_t0_redhat2021}
```

例题:2021红帽-签到

gzip

```
eJxLy0lMr05KSTEwM0pOTrFMNk9ONkhJNjYwNDI3SLVMSbJMMk5NrgUA9cQMNg==
```

eJx开头的等号结尾的

网站解密:https://codebeautify.org/gzip-decompress-online

jwt

因为 jwt 分为三部分，之间通过点号分隔，前两部分就是 base64 编码的所以直接可以 base64 解码

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c
```

辨别方法：eyJ

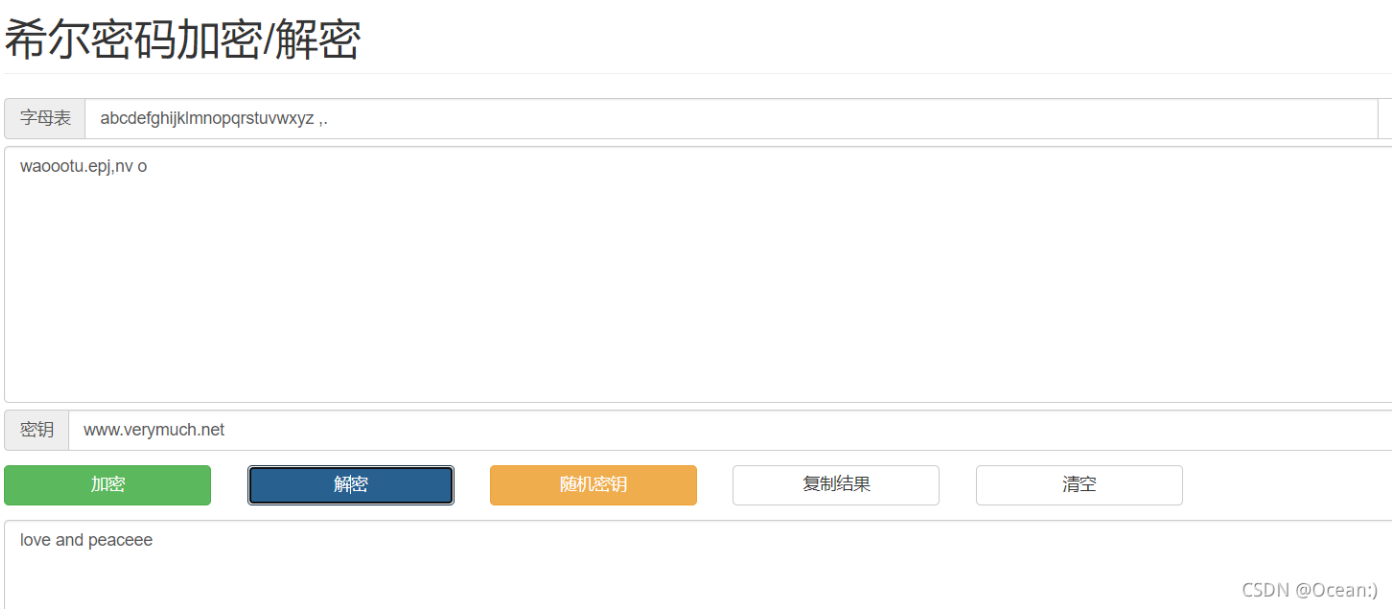
解密网站：https://jwt.io/

希尔密码

码表

```
abcdefghijklmnopqrstuvwxyz ,.
```

常见密钥为一个网址，比如



解密网址:http://www.atoolbox.net/Tool.php?Id=914

云影密码

有1，2，4，8这四个数字，可以通过加法来用这四个数字表示0-9中的任何一个数字，列如0=28， 也就是0=2+8，同理7=124，9=18。这样之后再用1-26来表示26个英文字母，就有了密文与明文之间的对应关系。引入0来作为间隔，以免出现混乱。所以云影密码又叫“01248密码”

```python
#!/usr/bin/python
# -*- coding=utf8 -*-
"""
# @Author : pig
# @CreatedTime:2019-11-2423:54:02
# @Description : https://www.jianshu.com/p/b5aa5cf60f83
"""


def de_code(c):
    dic = [chr(i) for i in range(ord("A"), ord("Z") + 1)]
    flag = []
    c2 = [i for i in c.split("0")]
    for i in c2:
        c3 = 0
        for j in i:
            c3 += int(j)
        flag.append(dic[c3 - 1])
    return flag

def encode(plaintext):
    dic = [chr(i) for i in range(ord("A"), ord("Z") + 1)]
    m = [i for i in plaintext]
    tmp = [];flag = []
    for i in range(len(m)):
        for j in range(len(dic)):
            if m[i] == dic[j]:
                tmp.append(j + 1)
    for i in tmp:
        res = ""
        if i >= 8:
            res += int(i/8)*"8"
        if i%8 >=4:
            res += int(i%8/4)*"4"
        if i%4 >=2:
            res += int(i%4/2)*"2"
        if i%2 >= 1:
            res += int(i%2/1)*"1"
        flag.append(res + "0")
    print ("".join(flag)[:-1])

c = input("输入要解密的数字串:")
print (de_code(c))
m_code = input("请输入要加密的数字串:")
encode(m_code)
```

PGP词汇表

链接https://baike.baidu.com/item/PGP%E8%AF%8D%E6%B1%87%E8%A1%A8/22718129

十六进制	单词数	偶词数	十六进制	单词数	偶词数	十六进制	单词数	偶词数	十六进制	单词数	偶词数
0	aardvark	adroitness	40	crackdown	Dakota	80	merit	intention	C0	slowdown	recipe
1	absurd	adviser	41	cranky	decadence	81	minnow	inventive	C1	snapline	recover
2	accrue	aftermath	42	crowfoot	December	82	miser	Istanbul	C2	snapshot	repellent
3	acme	aggregate	43	crucial	decimal	83	Mohawk	Jamaica	C3	snowcap	replica
4	adrift	alkali	44	crumpled	designing	84	mural	Jupiter	C4	snowslide	reproduce
5	adult	almighty	45	crusade	detector	85	music	leprosy	C5	solo	resistor
6	afflict	amulet	46	cubic	detergent	86	necklace	letterhead	C6	southward	responsive
7	ahead	amusement	47	dashboard	determine	87	Neptune	liberty	C7	soybean	retraction
8	aimless	antenna	48	deadbolt	dictator	88	newborn	maritime	C8	spaniel	retrieval
9	Algol	applicant	49	deckhand	dinosaur	89	nightbird	matchmaker	C9	spearhead	retrospect
0A	allow	Apollo	4A	dogsled	direction	8A	Oakland	maverick	CA	spellbind	revenue
0B	alone	armistice	4B	dragnet	disable	8B	obtuse	Medusa	CB	spheroid	revival
0C	ammo	article	4C	drainage	disbelief	8C	offload	megaton	CC	spigot	revolver
0D	ancient	asteroid	4D	dreadful	disruptive	8D	optic	microscope	CD	spindle	sandalwood
0E	apple	Atlantic	4E	drifter	distortion	8E	orca	microwave	CE	spyglass	sardonic
0F	artist	atmosphere	4F	dropper	document	8F	payday	midsummer	CF	stagehand	Saturday
10	assume	autopsy	50	drumbeat	embezzle	90	peachy	millionaire	D0	stagnate	savagery
11	Athens	Babylon	51	drunken	enchanting	91	pheasant	miracle	D1	stairway	scavenger
12	atlas	backwater	52	Dupont	enrollment	92	physique	misnomer	D2	standard	sensation
13	Aztec	barbecue	53	dwelling	enterprise	93	playhouse	molasses	D3	stapler	sociable
14	baboon	belowground	54	eating	equation	94	Pluto	molecule	D4	steamship	souvenir
15	backfield	bifocals	55	edict	equipment	95	preclude	Montana	D5	sterling	specialist
16	backward	bodyguard	56	egghead	escapade	96	prefer	monument	D6	stockman	speculate
17	banjo	bookseller	57	eightball	Eskimo	97	preshrunk	mosquito	D7	stopwatch	stethoscope
18	beaming	borderline	58	endorse	everyday	98	printer	narrative	D8	stormy	stupendous
19	bedlamp	bottomless	59	endow	examine	99	prowler	nebula	D9	sugar	supportive
1A	beehive	Bradbury	5A	enlist	existence	9A	pupil	newsletter	DA	surmount	surrender
1B	beeswax	bravado	5B	erase	exodus	9B	puppy	Norwegian	DB	suspense	suspicious
1C	befriend	Brazilian	5C	escape	fascinate	9C	python	October	DC	sweatband	sympathy
1D	Belfast	breakaway	5D	exceed	filament	9D	quadrant	Ohio	DD	swelter	tambourine
1E	berserk	Burlington	5E	eyeglass	finicky	9E	quiver	onlooker	DE	tactics	telephone
1F	billiard	businessman	5F	eyetooth	forever	9F	quota	opulent	DF	talon	therapist
20	bison	butterfat	60	facial	fortitude	A0	ragtime	Orlando	E0	tapeworm	tobacco
21	blackjack	Camelot	61	fallout	frequency	A1	ratchet	outfielder	E1	tempest	tolerance
22	blockade	candidate	62	flagpole	gadgetry	A2	rebirth	Pacific	E2	tiger	tomorrow
23	blowtorch	cannonball	63	flatfoot	Galveston	A3	reform	pandemic	E3	tissue	torpedo
24	bluebird	Capricorn	64	flytrap	getaway	A4	regain	Pandora	E4	tonic	tradition
25	bombast	caravan	65	fracture	glossary	A5	reindeer	paperweight	E5	topmost	travesty
26	bookshelf	caretaker	66	framework	gossamer	A6	rematch	paragon	E6	tracker	trombonist
27	brackish	celebrate	67	freedom	graduate	A7	repay	paragraph	E7	transit	truncated
28	breadline	cellulose	68	frighten	gravity	A8	retouch	paramount	E8	trauma	typewriter

UUencode

Uuencode将输入资料以每三个字节为单位进行编码，如此重复进行。如果最后剩下的资料少于三个字节，不够的部份用零补齐，很像base64

特点：包含!"#￥%&‘（）*+=’” 等等字符

Unicode

Unicode 在一个字符集中包含了世界上所有文字和符号，统一编码，来终结不同编码产生乱码的问题

字符编码 UTF-8

Unicode 统一了所有字符的编码，是一个 Character Set，也就是字符集，字符集只是给所有的字符一个唯一编号，但是却没有规定如何存储

一个字符使用四个字节存储，也就是 32 位，这样就能涵盖现有 Unicode 包含的所有字符，这种编码方式叫做 UTF-32（UTF 是 UCS Transformation Format 的缩写）

在存储和网络传输中，通常使用更为节省空间的变长编码方式 UTF-8，UTF-8 代表 8 位一组表示 Unicode 字符的格式，使用 1 - 4 个字节来表示字符

```
U+ 0000 ~ U+ 007F: 0XXXXXXX
U+ 0080 ~ U+ 07FF: 110XXXXX 10XXXXXX
U+ 0800 ~ U+ FFFF: 1110XXXX 10XXXXXX 10XXXXXX
U+10000 ~ U+1FFFF: 11110XXX 10XXXXXX 10XXXXXX 10XXXXXX
```

可以看到，UTF-8 通过开头的标志位位数实现了变长。对于单字节字符，只占用一个字节，实现了向下兼容 ASCII，并且能和 UTF-32 一样，包含 Unicode 中的所有字符，又能有效减少存储传输过程中占用的空间

RTF格式下的unicode编码

明显特点：`\u-65432?\u-65420`

转换脚本

```
s=r"\u-65432?\u-65420?\u-65420?\u-65424?\u-65421?\u-65478?\u-65489?\u-65489?\u-65418?\u-65426?\u-65437?\u-65420?
\u-65434?\u-65491"
l=list(s[3:-1].split(r"?\u-"))
flag=""
for i in l:
    flag+=chr(65536-int(i))
print(flag)
```

autokey

自动密钥密码（Autokey Cipher）也是多表替换密码，与维吉尼亚密码密码类似，但使用不同的方法生成密钥。通常来说它要比维吉尼亚密码更安全。自动密钥密码主要有两种，关键词自动密钥密码和原文自动密钥密码
解密网站:https://www.wishingstarmoye.com/ctf/autokey
http://www.practicalcryptography.com/ciphers/autokey-cipher/

```
# python2 解码
from pycipher import Autokey

print Autokey('CULTURE').encipher('helloworld')
print Autokey('CULTURE').decipher('jyweinsypo')
```

工具下载：https://github.com/hitcxy/break_autokey
工具使用

```
需要安装pycipher库 pip2 install pycipher
修改break_autokey.py中的'ctext' 变量
python2 break_autokey.py
```

其他稀奇古怪编码

https://www.dcode.fr/
可以用这个网站

特殊编程语言

JS变种

已下这些都是js，**打开console运行一下就能得到flag**

brainfuck

http://www.hiencode.com/brain.html

jsfuck

用六个不同的符号
- !+
构造出JS的所有类型
http://www.hiencode.com/jsfuck.html

aaencode

**也称颜文字 **ﾟωﾟﾉ= /｀m´）ﾉ ~┻━┻ //´▽｀/ ['_']; o=(ﾟｰﾟ)
将JS代码转换成常用的网络表情
http://utf-8.jp/public/aaencode.html

jjencode

demo地址
http://utf-8.jp/public/jjencode.html

xxencode

jother

jother是一种运用于javascript语言中利用少量字符构造精简的匿名函数方法对于字符串进行的编码方式。其中8个少量字符包括：！+ ( ) [ ] { } 。只用这些字符就能完成对任意字符串的编码。不同于jsfuck，它多了{}这两个大括号
解密工具 由于jother执行之后所得到的结果分为字符串和函数两种，所以解密的方法也不相同。
字符串：直接在Console界面中输入并回车即可
函数： 对于函数类型的jother加密结果，我们只需要将最后的()改成.toString()即可

ctf中js的总结

npiet

有关颜色的编程语言

Info: found picture width=120 height=116 and codel size=4
Sample picture (shown with a small border): **hello_medium.gif**

Info: executing: npiet –w –e 220000 hello_medium.gif

Hello world!

CSDN @Ocean:)

更多查看https://www.jianshu.com/p/ed929cf72312

ALPHUCK

字和符号，统一编码，来终结不同编码产生乱码的问题