## Lab - DeRPnStiNK: Walkthrough

## Overview

## This is a boot2root Ubuntu-based virtual machine. The Walkthrough is rated as beginner, but I found it to me at least intermediate. Your goal is to remotely attack the VM and find all four flags eventually leading you to full root access. Stick to your classic hacking methodology and enumerate everything!

**Hardware Requirements**

* Installation of VirtualBox or VMWare Player or Workstation Pro
* One virtual install of Kali Linux
* One virtual install of the DeRPnStiNK OVA file which can be downloaded from [here](https://www.vulnhub.com/entry/derpnstink-1,221/).

Ensure the network adapter for both machines to set to either bridged or NAT.

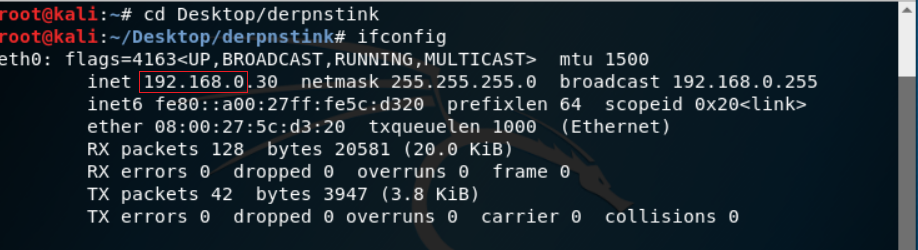
**Organization**

Create a folder on the desktop of your Kali machine. Name the folder, **derpnstink**. When using a terminal, change directory to the **derpnstink** folder and run all your commands from this location. Save any downloads or captured files to this location.



We begin with the basics (always) by enumerating the machine for its IP address and any open ports and services that maybe running.

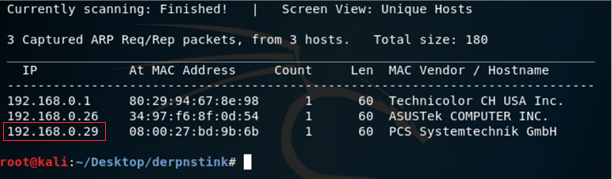
There’s no harm is getting the network ranges by doing an IFCONFIG from your Kali terminal.



Once we have our network range, we can discover the target machine’s IP address by using **netdiscover, Nmap** or **ARP**.

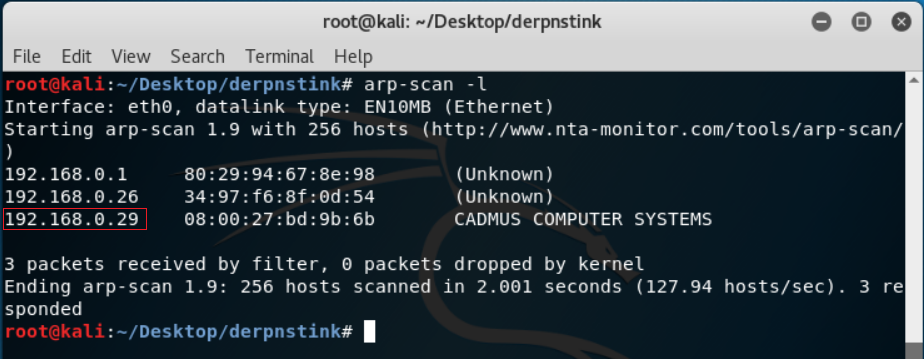
**Using netdiscover**

netdiscover -r 192.168.0.0/24



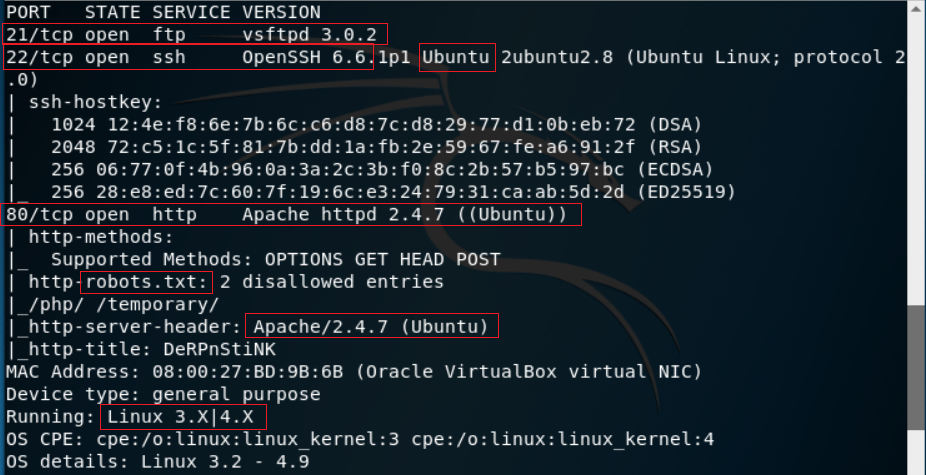
**Using ARP**

arp-scan -l



We’re now ready to do a Nmap scan.

nmap -sS -AT4 192.168.0.26



## Port: 21

There is an FTP server running at port 21. Nmap informs us that it is version 3.0.2 vsftpd server and connecting to the service with the ftp command confirms this. Unfortunately, it seems that the anonymous user has been disabled.

## Port: 22

OpenSSH 6.6.1p1 is running on port 22. Nmap tells us that it is an Ubuntu version, providing a pretty good hint as to what OS our target is using.

## Port: 80

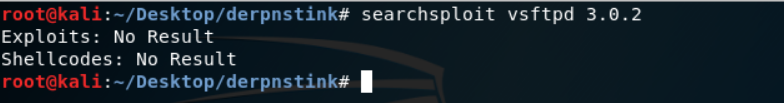
There is a web server running at port 80, powered by Apache version 2.4.7.

**Vulnerability Analysis**

**FTP**

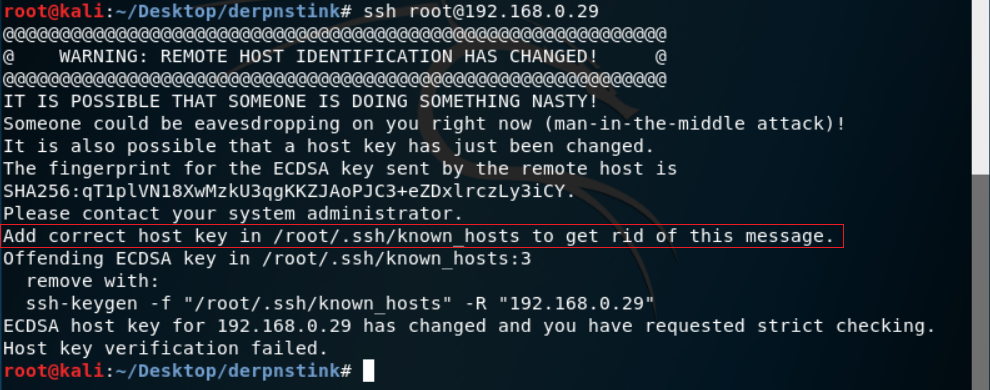
Are analysis of the exploits doesn’t turn up much.

Searchsploit doesn’t return any vulnerabilities for vsftpd 3.0.2.



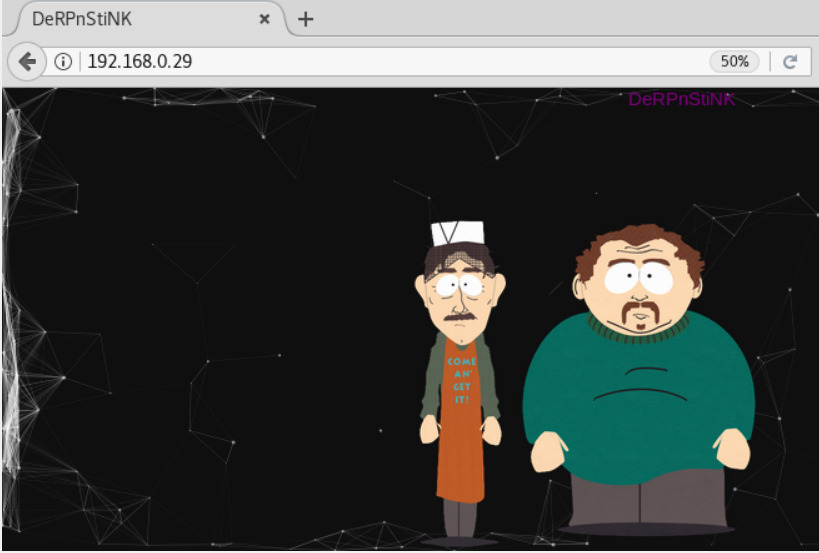
**SSH**

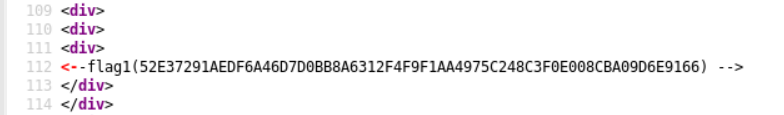
The SSH service does not appear to be vulnerable to anything, either. Attempting to connect to the server shows that it password login is disabled in favor of private/public key pairs. Not going to be brute-forcing that.



**HTTP**

There do not appear to be any exploits for our version of Apache. Visiting the site yields a page without any links. However, if we examine the source code of the page, we find our first flag. (Near the bottom of the page.)





## We need not forget the basics of enumerating a web server, and this means the looking at the contents of the robots.txt file. The robots.txt is a standard used by websites to communicate with web crawlers and other web robots. The robots.txt specifies how to inform the web robot about which areas of the website should not be processed or scanned. Great if you’re trying to hide a portion of your website such as a personal blog.

## 

## Both entries yield nothing at first.

## 

## Running dirb, we find some interesting content.

## 

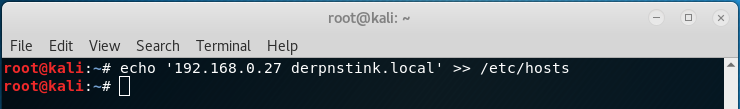
## (snip)

## 

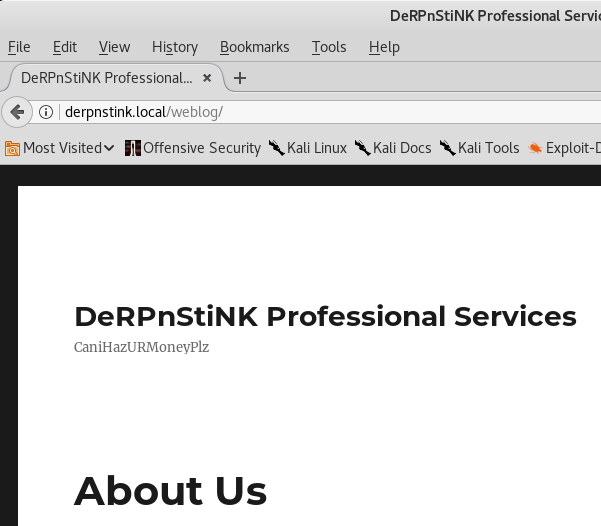
## The /php/ contains a phpmyadmin installation. This might yield some great information if we can log in. There is another directory at /weblog/ that contains a WordPress installation.

## If you try and visit the WordPress site, it tries to redirect to derpnstink.local. To resolve this domain, we need to add the domain to our /etc/hosts file:

echo '192.168.0.27 derpnstink.local' >> /etc/hosts



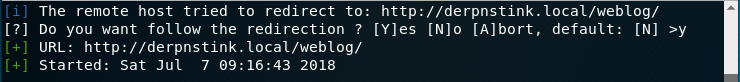
We can now attempt to navigate to 192.168.0.27/weblog/



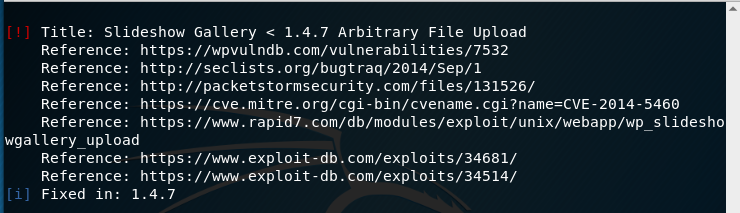
With this access, we can now run a wpscan. We use wpscan to enumerate the plugins and themes and users.

****wpscan --enumerate u u[10-20] ap at --url http://192.168.0.27/weblog/****

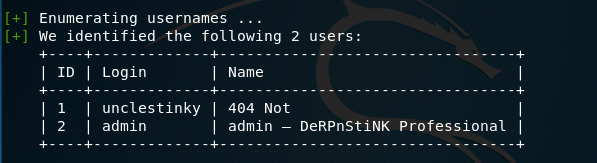
Once the scan starts we divert from the default by allowing the redirect.



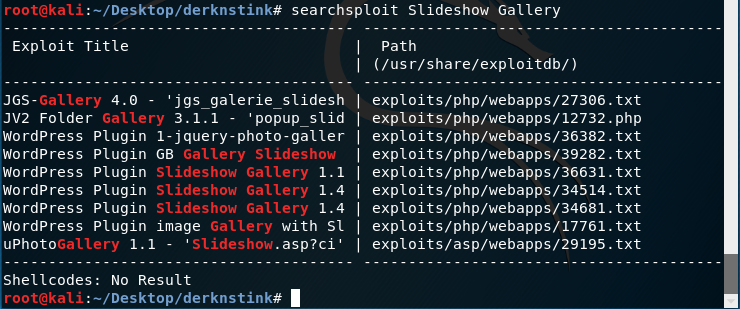
The scan has discovered an arbitrary file upload vulnerability in one of the installed plugins being stored in the weblog directory. We also know that this is where the WordPress site is being hosted.



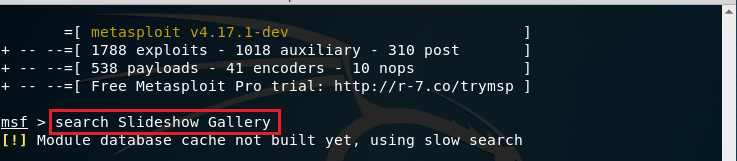
We also find the username and password for the WordPress site is set to use the default of admin:admin.



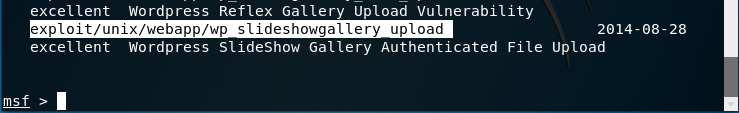
We know from the scan results that the Title: Slideshow Gallery is vulnerable and if we use searchspolit to search for an exploit, we get a positive hit.



We can use Metasploit to exploit this vulnerability. From the Metasploit prompt, we can search for all the exploits available for the Slideshow Gallery plugin.



From the search results, we can discern that the one we need to use is,

****

msf > use exploit/unix/webapp/wp\_slideshowgallery\_upload

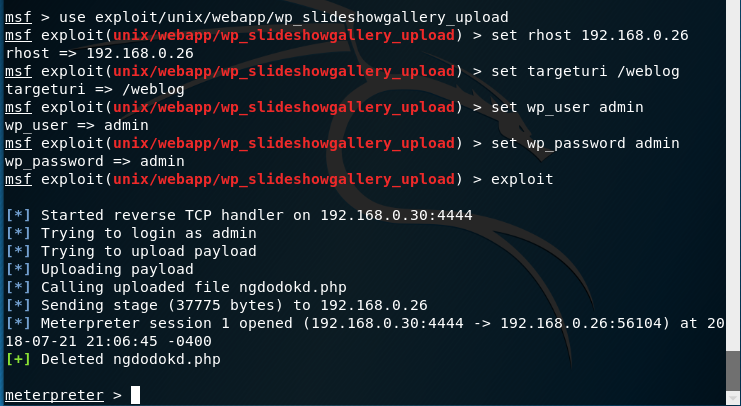
msf exploit(unix/webapp/wp\_slideshowgallery\_upload) > set rhost 192.168.0.27

msf exploit(unix/webapp/wp\_slideshowgallery\_upload) > set targeturi /weblog

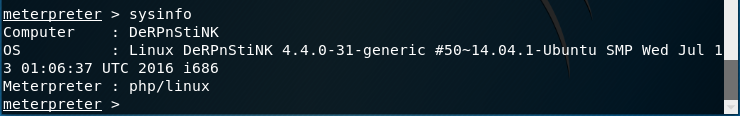
msf exploit(unix/webapp/wp\_slideshowgallery\_upload) > set wp\_user admin

msf exploit(unix/webapp/wp\_slideshowgallery\_upload) > set wp\_password admin

msf exploit(unix/webapp/wp\_slideshowgallery\_upload) > exploit



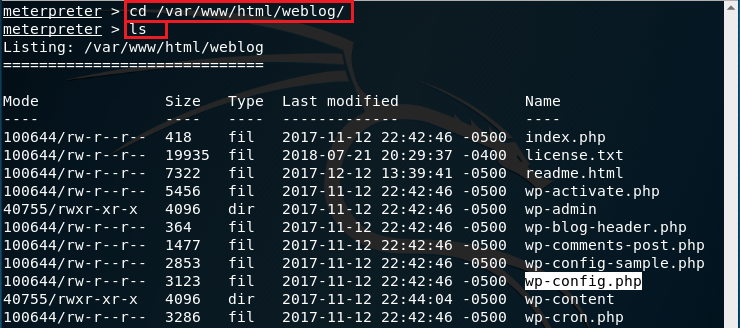
Use the sysinfo command to get some basic information about the system.



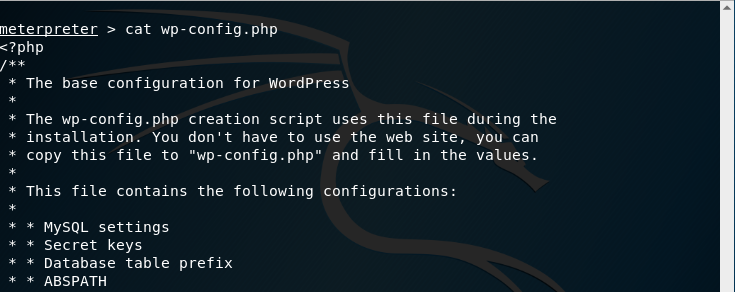
Note: We could have easily logged onto the WordPress site using the admin credentials we found with the wpscan. From there we could have uploaded a PhP script and established a shell using Netcat as a listener, but that would have given us only limited shell access. A Metepreter prompt is always better than a limited shell.

Change location over to the over the weblog directory.

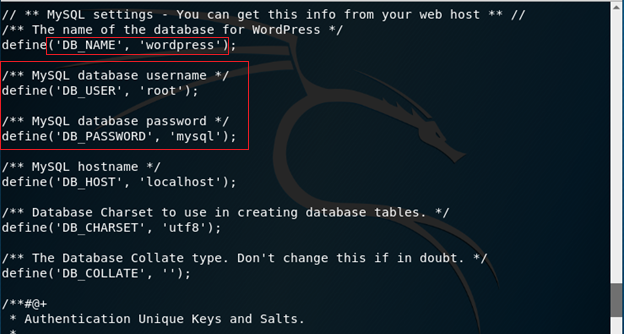
List the contents of the weblog directory using the ls command.



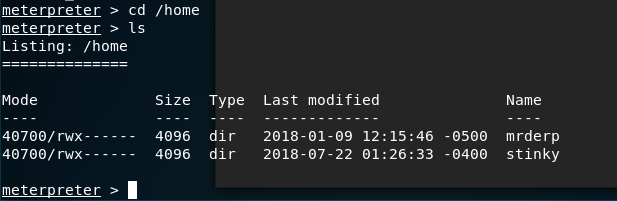
We can now open the wp-config.php file and find the name of the database along with the user and password required to access the database.



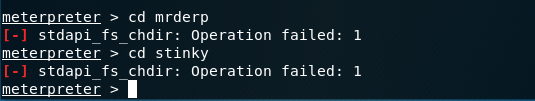
We find the username and password required for mysql.



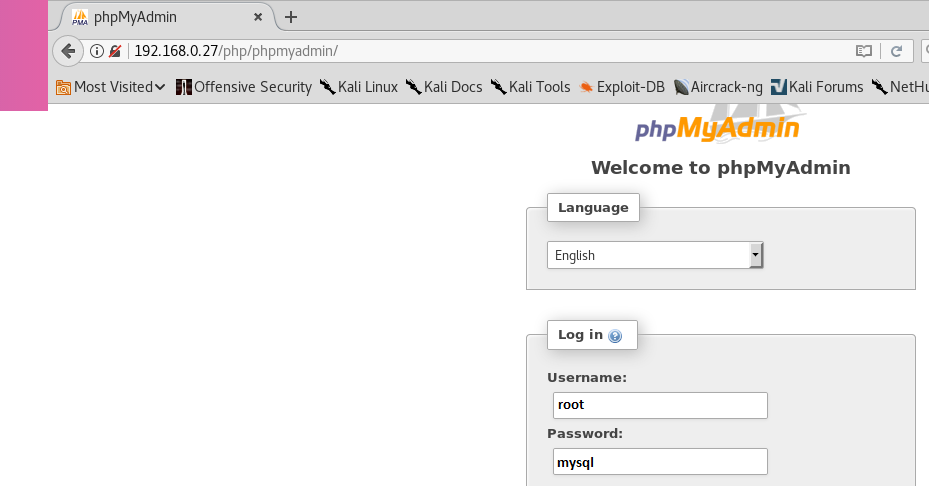
Change location to the home directory and list the contents.



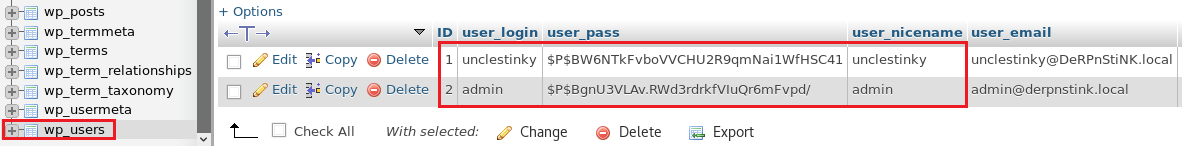
If we try and access either directory, we get denied access. We need to tray and logon as either mrderp or stinky.



We can use the information we gathered for the MySQL credentials to login through phpmyadmin to try and find their user accounts and password information.

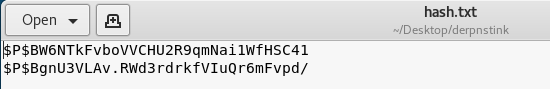


After logging on through phpmyadmin, we find two user accounts and the password hashes for both users in the WordPress database in the wp-users file.



We can use john the ripper to crack the hashes and find a password for unclestinky.

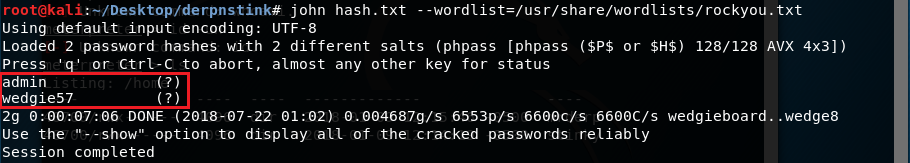
We first create a new text file inside our working directory called, hash.txt. We then copy the two hashes over to the hash.txt file. One hash per line.



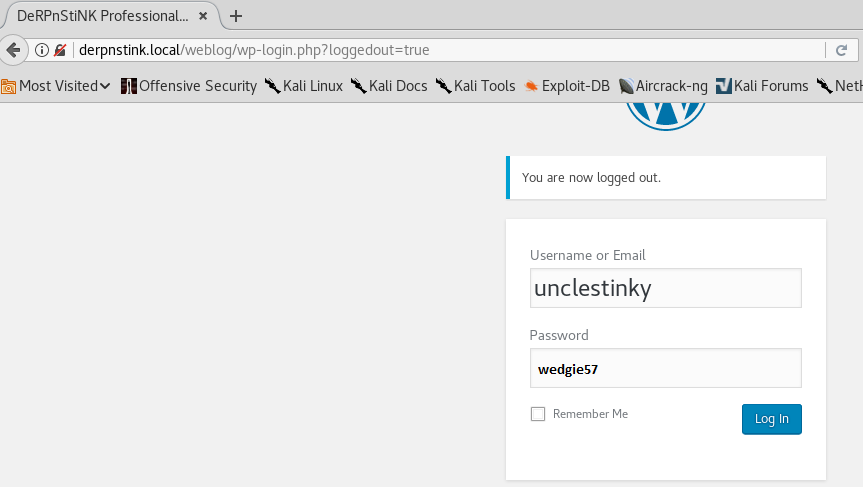
We are now ready to crack the hashes using John the Ripper.

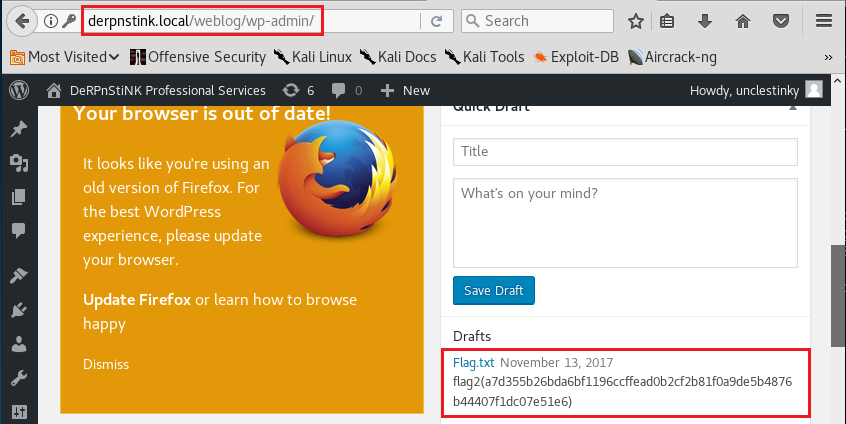
Note: The rockyou.txt wordlist may need to be extracted from its archive. Use the file manager to locate the archive and extract the file to the wordlists directory.

john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt



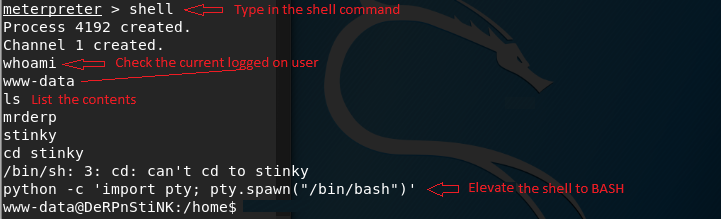
We can now logon as unclestinky using the password wedgie57 to the WordPress site (http://derpnstink.local/weblog/wp-login) where we locate our second flag.





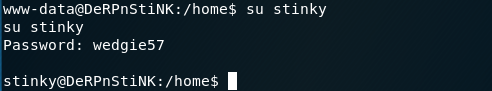
Back at my Meterpreter prompt, I type in the shell command. We are using a very restricted account called www-data but with a little bit of Python code, we can elevate the prompt to a BASH shell.

Ls

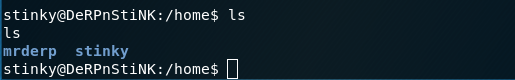


We have the login credentials for stinky so let’s use them.

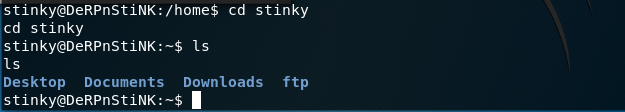
Change user to stinky and type in his password, wedgie57.



List the contents of his home directory.

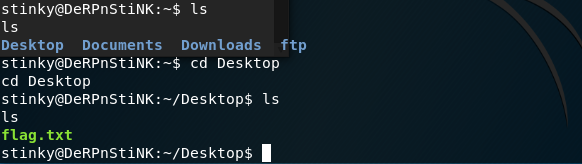


Change directory over to the directory stinky and list the contents.

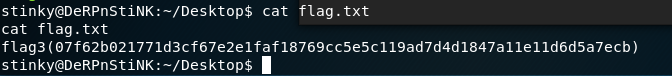


We need to enumerate everything in stinky’s profile, so we start with the Desktop folder and work our way across. Change location over to the Desktop folder and list the contents.

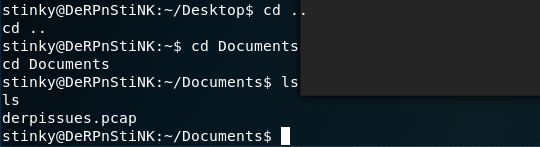
And we found our third flag! Nice!



Show the contents of the flag.txt file.

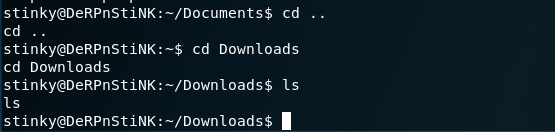


Onto the Documents folder…..



We have a derpissues.pcap file that could be of interest. We’ll take note and keep looking.

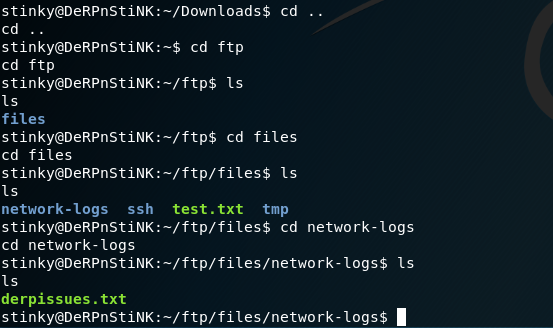
Move onto the Downloads folder. The folder is empty.



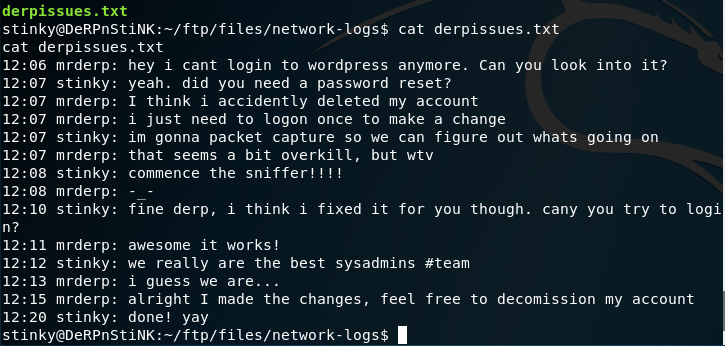
Move onto the ftp folder.



Change location to the files directory and then again to the network-logs and list the contents.

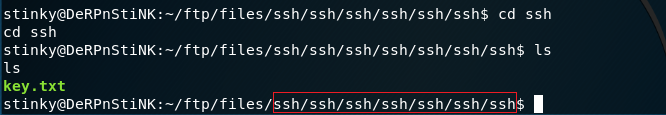


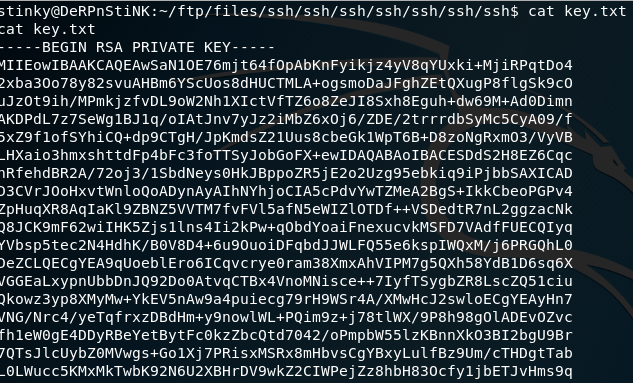
We have a text file called derpissues.text. Using the cat command, examine the contents.



Makes for an interesting read. The information we may want for the mrderp’s login credentials are probably in the pcap file we found earlier inside the Documents folder. Good to know but we still have more data to enumerate. Let’s look inside the ssh folder.

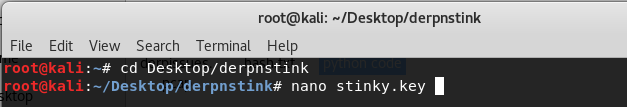
There are seven ssh folders to list content for. In the last folder, there is a key.txt file. View the contents of the file to see the key.





This is the SSH key for the user, stinky.

We can now go after a more stable logon using SSH. To do so, we first create new text file up inside our working directory and call it, stinky.key



Copy the key and paste it into the stinky.key file. I’m using nano as my text editor, so I will use Ctrl+x to save the file, type in Y to save the changes and hit enter to close the text editor.

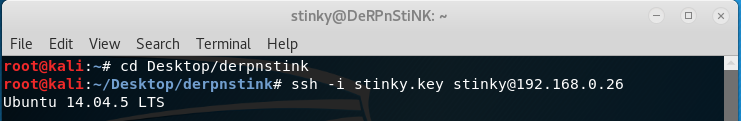
We next need to change the permissions on the file we just created. At the prompt, type

chmod 400 stinky.key

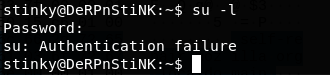


We are now ready to try and logon to the target using SSH. At the prompt, type the following command.

ssh -i stinky.key stinky@192.168.0.26



We can check the permissions stinky has using the su -l command and we find out he does not have any su permissions.



Are next task is to copy over the pcap file we found in stinky’s douments. To do this we open a new prompt, change location over to our working directory and use the following command:

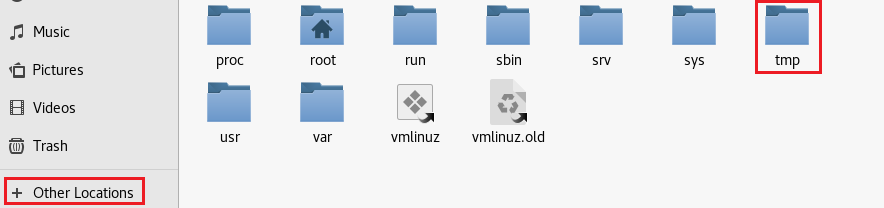
scp -i stinky.key stinky@192.168.0.26:/home/stinky/Documents/derpissues.pcap /tmp/derpissues.pcap



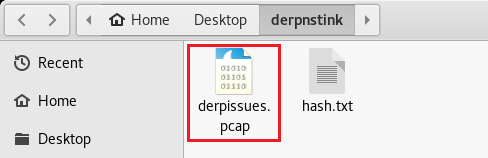


You’ll can now navigate over to your tmp folder and move the file file to your working directory.

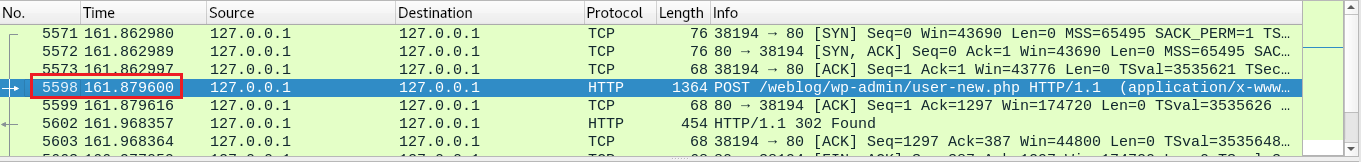
Open you file manager, Other Locations and open the tmp folder. Right click on the pcap file and select, Move to and select your working directory. Reason we use the tmp folder because it has unlimited access and no restrictions.



Open your working directory and right click on the derpissues.pcap file and select, Open with Wireshark.



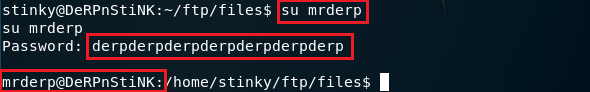
Right click on the pcap file and open with Wireshark.



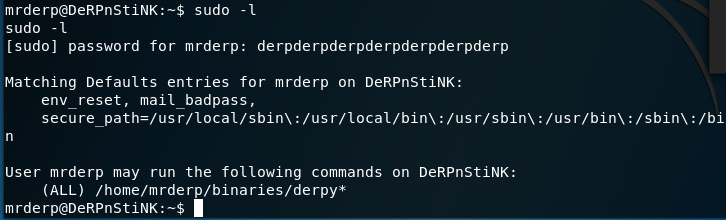
Right clink on entry 5598 and select to follow>TCP Stream.



We can now login as mrderp using the password, derpderpderpderpderpderpderp we discovered using Wireshark.



We next check to see what commands as root mrderp is permitted to run using the sudo -l command.



We learn that mrderpis not allowed to run /bin/su with sudo. The sudo -l command tells us what he can run with sudo

(ALL) /home/mrderp/binaries/derpy\*

When it says all, it means all commands as sudo. We can get this access using any file starting with derpy that resides inside /home/mrderp/binaries directory. All we must do is create a binaries folder, and put a script inside named derpy.sh to start a Bash shell:

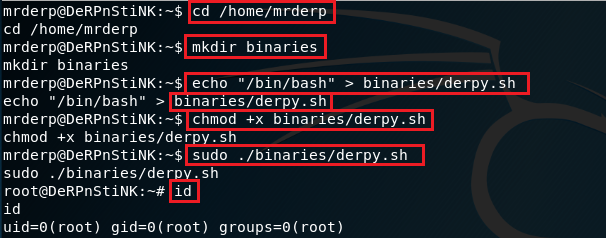
cd /home/mrderp/

mkdir binaries

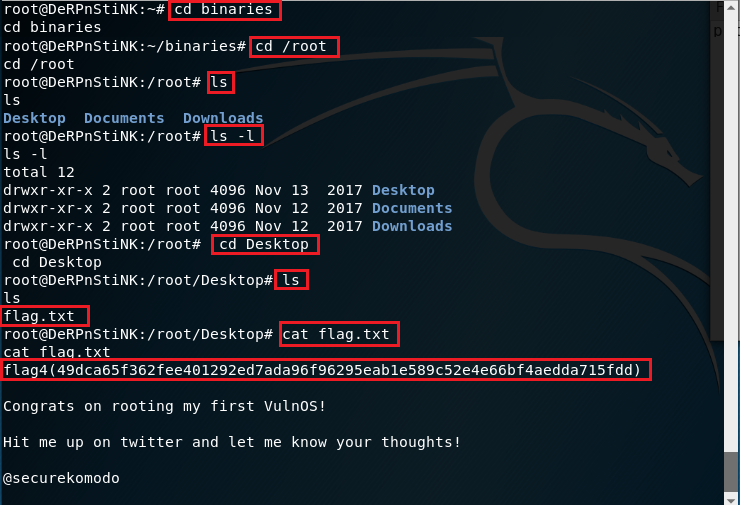
echo "/bin/bash" > binaries/derpy.sh

chmod +x binaries/derpy.sh

sudo ./binaries/derpy.sh



Let’s take it on home!



Summary

The great thing about this walkthrough was being able to call upon the different vectors I had learned from previous walkthroughs. When I got stuck trying to figure out how to escalate a shell to a BASH prompt, I recalled using a Python script from a previous walk-through that allowed me to escalate the shell to BASH.

That was not the only vector I could’ve p pulled from memory, there was the PHP script from the Pentest Monkey site for establishing a shell through a WordPress plugin, but for this walk-through, I chose to use Meterpreter to establish a shell.

Get used to carrying your tools on a USB stick around your neck, and that includes a file for all your favorite scripts and hacking vectors. Copy the file to your working folder in Kali.

All these walk-throughs have numerous vectors that can be used to accomplish the same exploit. Some people would’ve wanted to use the SSH capability, but I was able to establish the same access using FTP.

I would rate this walkthrough as intermediate and then some as it took quite a bit of research and time to get through it.

Don’t forget to use your hacking methodology!

End of the Walkthrough!