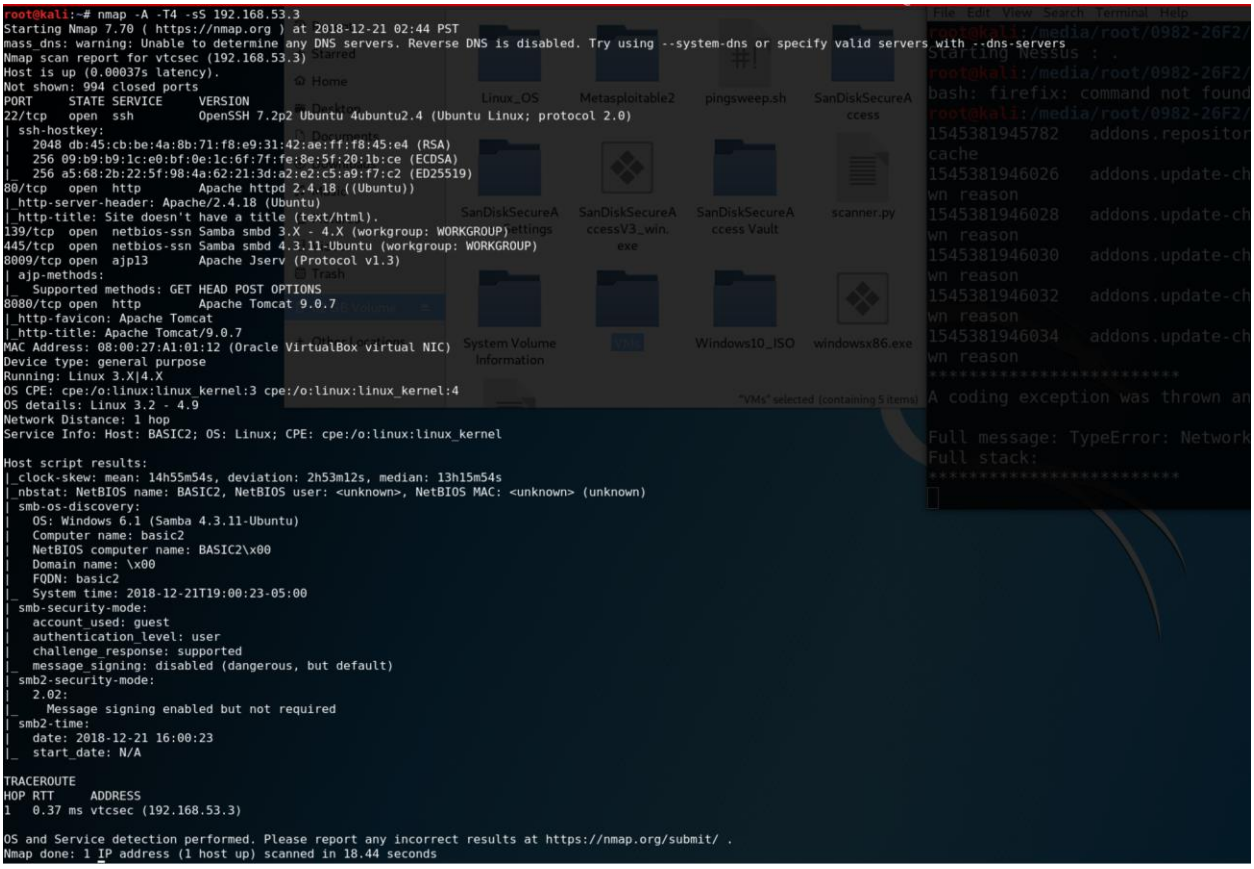
Basic Pentesting Two

The Basic Pentesting Two machine can be obtained from www.vullnhub.com . basic pentesting two is a step up from Basic Pentesting One. During the penetration test there were four vulnerabilities that were found (two exploits and two privilege escalations). The attacking machine used was Kali Linux, which has an IP address of 192.168.53.4. Basic Pentesting Two has an IP address of 192.168.53.3.

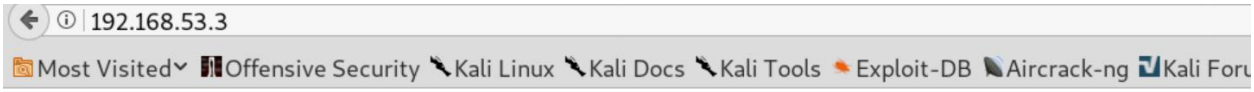
Information Gathered

A list of open ports was obtained using nmap. The command issued can be seen below along with a snippet of the results:



Looking through the list of services and version information we see that our target is a Linux machine version 3.2-4.9. Next, Google was used to search for any exploits that may be available to use against the machine. The following services contained exploits from Metasploit: ssh, and Samba (port 445); Unfortunately, the Samba exploit failed to resolve. The exploit for OpenSSH version 7.2p2 allowed us to attempt user enumeration, which may come in handy later.

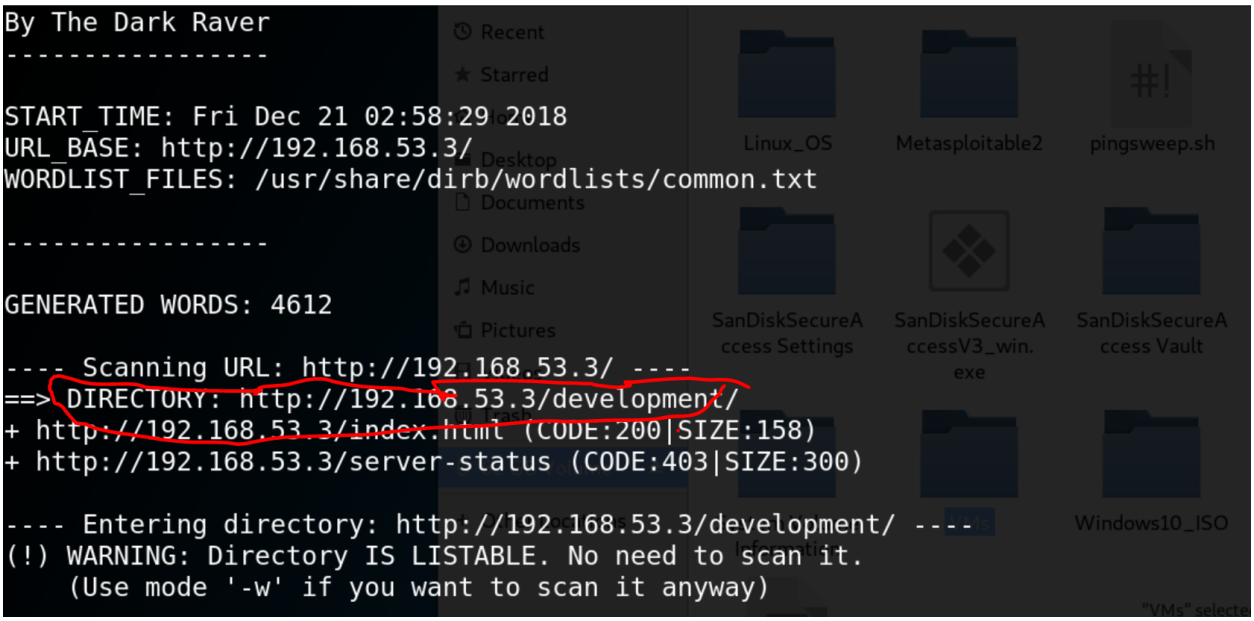
The machine also had two webpages hosted on ports 80 and 8080. Browsing to the first page we find that the website is currently under development (we are presented with an error message).



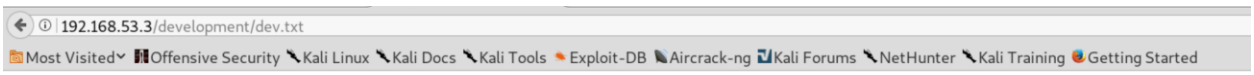
Undergoing maintenance

Please check back later

However, running dirbuster against the webserver, dirb http://192.168.53.3, yields a list of hidden directories.



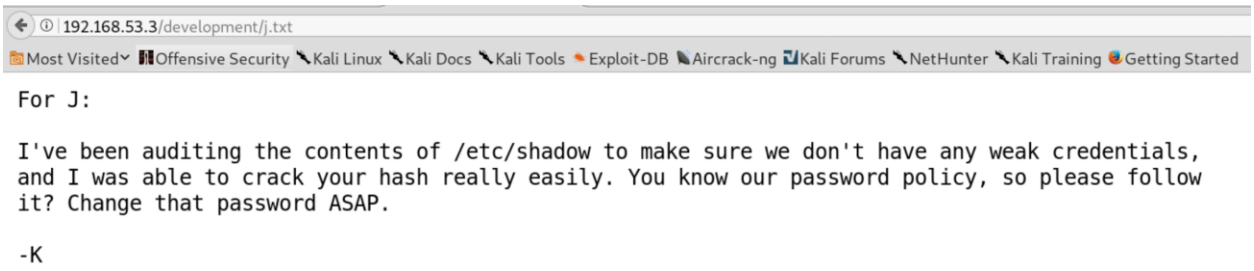
The /development directory contains a list of conversations between two developers with initials j and k.



2018-04-23: I've been messing with that struts stuff, and it's pretty cool! I think it might be neat to host that on this server too. Haven't made any real web apps yet, but I have tried that example you get to show off how it works (and it's the REST version of the example!). Oh, and right now I'm using version 2.5.12, because other versions were giving me trouble. -K

2018-04-22: SMB has been configured. -K

2018-04-21: I got Apache set up. Will put in our content later. -J

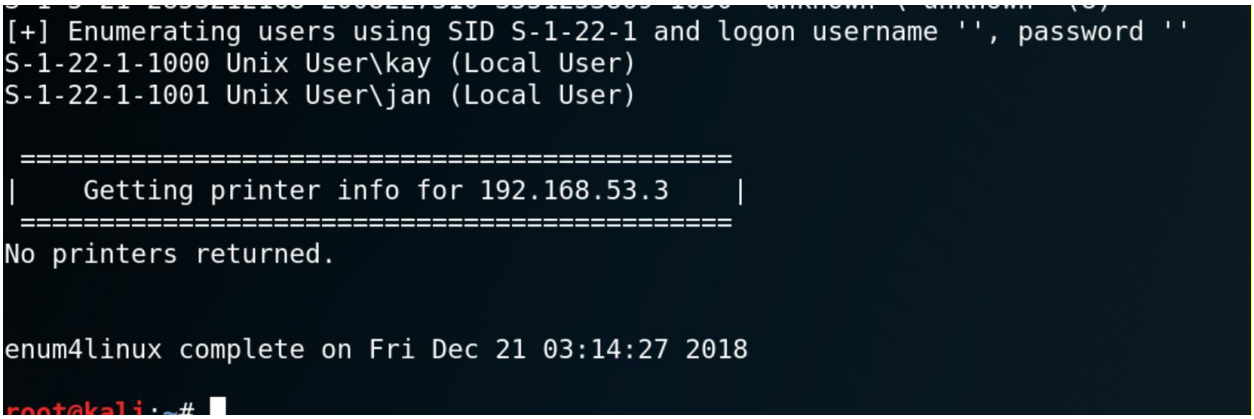


- The first message seems to be referring to the specific version of Apache that is being run on the target (Apache strut 2.5.12), which may be vulnerable to CVE-2017-9805 (follow link for more information).
- The second message is an exchange between J and K. J is being scolded for using a weak password. It may be worth while to attempt a dictionary attack against J's account; however, we need to obtain more information first (J's username would be perfect).

Obtaining Usernames

Since the victim is running Samba we can use enum4linux to find additional information. Running enum4linux yields a plethora of information. The most useful information are the two usernames obtained, which are shown in the following screenshot.

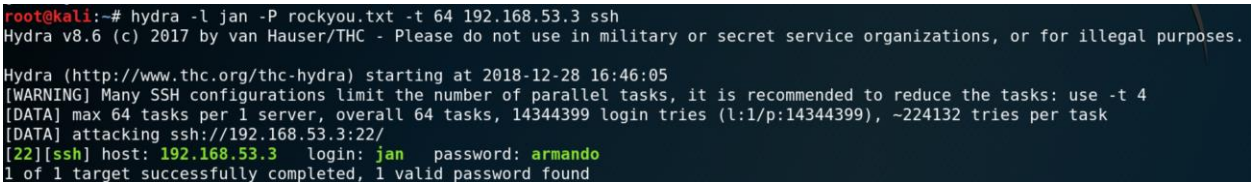
(enum4linux -a 192.168.53.3)



Since we have the usernames of two system users, we can attempt to gain access to the opened SSH port via a dictionary attack.

Attacking the SSH Service

Since the usernames are now known and Jan is using a weak password a dictionary attack should be quite effective. Hydra will allow us to quickly find the password for Jan's account. The screenshot below shows the output from Hydra.



As can be seen in the tools output the user jan has set armando to be her password. Since the password is now known we can login to the target machine using the open SSH port (ssh -l jan 192.168.53.3).

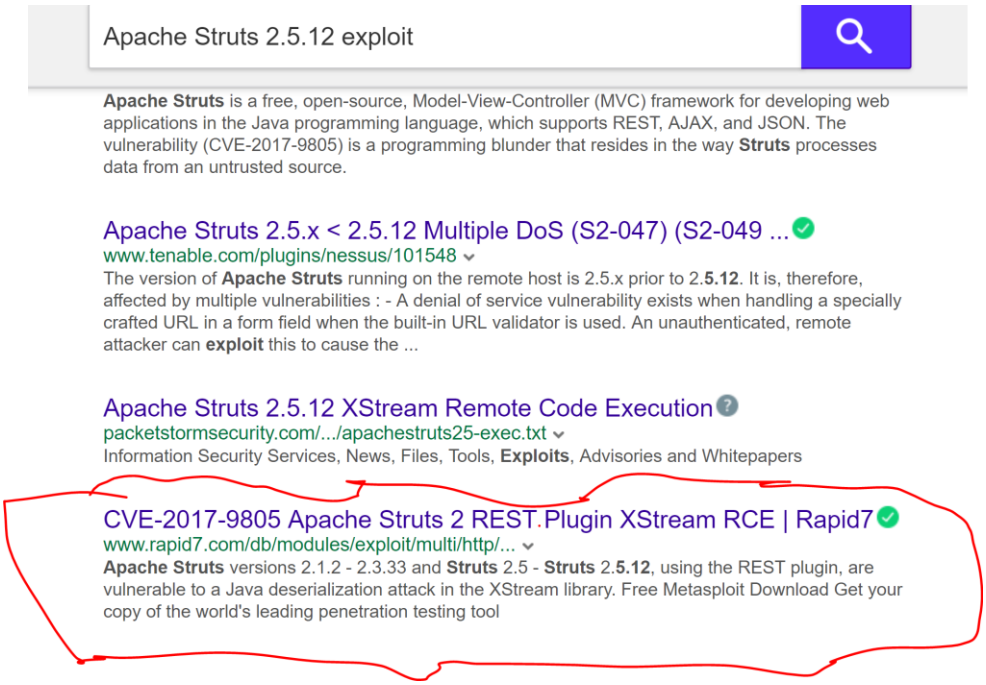
Escalating Privileges

Successfully logging into jan's account allowed us to get a foothold on the system; however, we are still unable to view the root directory or view/edit any important files. Checking out the /home directory reveals the other users on the system (there is only one other user kay). Kay's directory allows the group to read any file inside of his directory. Changing to Kay's directory and performing the ls -la command reveals a file called pass.bak. Attempting to cat this file results in a permission dennied error message because the owner has set the permissions in such a way as to allow only himself the ability to read and wirte to it. This should prevent us from reading the file but sometimes trying a second time pays off. This time we used vim to attempt to open the file and it worked! It seems the the owner of the system has misconfigured vim. The pass.bak file contains another passowrd, but this time it appears to belong to kay:

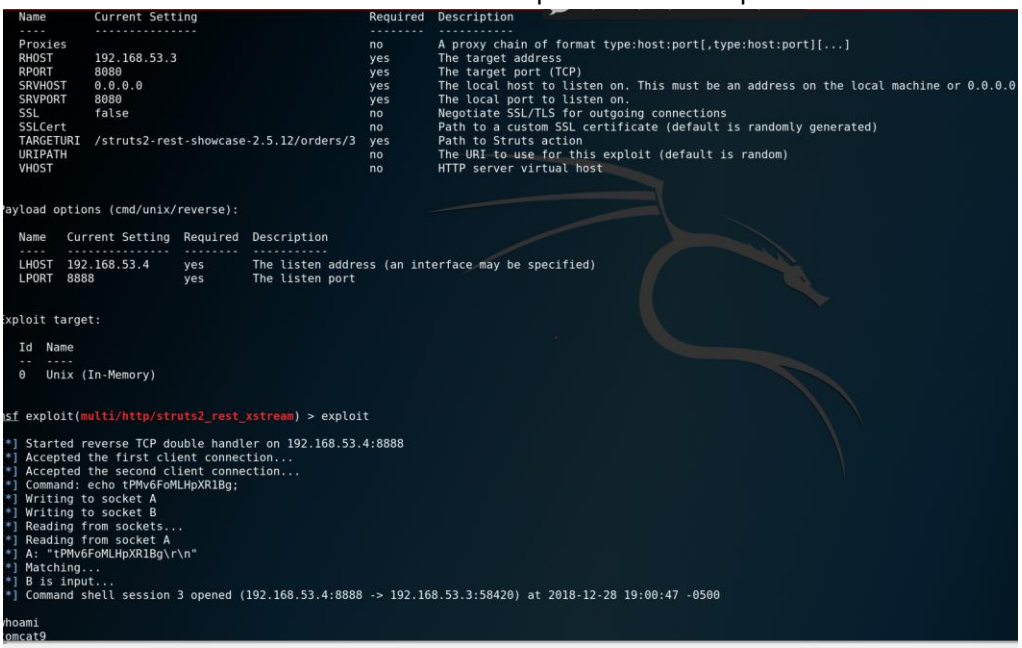
Let's see if this is really kays password. Opening a new terminal and attempting to login using kay's account reveals that kay's password was indeed stored in the pass.bak file. Kay is still not a root user, but his account is listed as a sudoer, which means kay has the ability to perform commands with the privileges of a root user. The command sudo ls /root allows us to view the root directory and reveals the flag.txt file that we are after. If complete control of the system is needed a root user can be added. Let's add a root user with the user name hacker@root. To do this we will use the following command: sudo adduser -ou 0 -g 0 hacker@root. This command tells the system to create a user with uid 0 and group id of zero. This will work even if there is already a root user on the system. Issuing the passwd command will allow us to change the password for this user: sudo passwd hacker@root. Once the user is created use the su command to switch users to hacker@root. (su hacker@root).

Additional ways to pwn the machine

Since the vim editor on this machine allowed us to read and write to any file on the system, we could have also escalated jan's privileges by adding jan to the sudoers file (vim /etc/sudoers). In addition to an additional privilege escalation opportunity there is also an additional vulnerability on the machine. In the note for jan, kay let it slip that the webserver is using Apache struts 2.5.12. A quick google search for Apache Struts 2.5.12 exploit reveals that there is Metasploit module available for this version of Apache Struts.



Running this module in Metasploit (exploit/multi/http/struts2_rest_xstream) will give us control of the webserver. Here is a screenshot of the options used to exploit the machine:



Running the exploit, we obtain access to the machine as the user tomcat9, which is the name of the webserver running on port 8080 of the target machine. NOTE that the target URI field is the path to a struts action. In this case the URI pointed to a page that allowed the user to update order information. From here privileges may be escalated using any of the above techniques.