Linux Artifacts – Passwd and Shadow

What are '/etc/passwd' and '/etc/shadow'?

Traditionally, the /etc/passwd file is used to keep track of every registered user that has access to a system. All users will have read access, but only super users will have the ability to write to the file. Why is this useful? Because it gives us information about every user on the system. In a forensic investigation maybe the user has a secret second user account that they have disguised to look like a service account, or maybe during an incident response an attacker gained access to this Linux system and created an additional account for persistence.

Below is a screenshot of the passwd file on our Kali Linux virtual machine. We can see our account "root" at the top on the second line, with a lot of other entries below. These are all service accounts created by different programs to manage and run daemons. You can see how a second user account could get lost in all of this mess, and identifying it could uncover a lot of digital evidence.

```
root@kali:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:101:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
mysql:x:104:110:MySQL Server,,,:/nonexistent:/bin/false
tss:x:105:111:TPM software stack,,,:/var/lib/tpm:/bin/false
strongswan:x:106:65534::/var/lib/strongswan:/usr/sbin/nologin
ntp:x:107:112::/nonexistent:/usr/sbin/nologin
messagebus:x:108:113::/nonexistent:/usr/sbin/nologin
redsocks:x:109:114::/var/run/redsocks:/usr/sbin/nologin
rwhod:x:110:65534::/var/spool/rwho:/usr/sbin/nologin
iodine:x:111:65534::/var/run/iodine:/usr/sbin/nologin
miredo:x:112:65534::/var/run/miredo:/usr/sbin/nologin
dnsmasq:x:113:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
usbmux:x:114:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
tcpdump:x:115:119::/nonexistent:/usr/sbin/nologin
rtkit:x:116:121:RealtimeKit,,,:/proc:/usr/sbin/nologin
```

On the second line where we have our current user "root" we can see an X next to the username. This is the account password. Well, it's really just a variable, because the password is encrypted, and stored somewhere else. The second file we're going to cover, called `/etc/shadow`, contains encrypted password as well as other information such as account or password expiration values. The /etc/shadow file is readable only by the root account to prevent standard users from grabbing the contents and then using a tool such as hashcat or John The Ripper to brute force, perform a dictionary attack, or use rainbow tables to crack the hashes and reveal the plaintext passwords.

Let's read the contents of this file using `sudo cat /etc/shadow`. We can see that next to our root account there's an encrypted password value.

```
root@kali:~# sudo cat /etc/shadow
root:$6$vxc2jZ6ZNUrC7Sn4$GOAFTNo7m40NZata7afsiKOFaYZh6RBOlnQ8Qw6GT9s58UWD4u8uk.Nh4dKPmYcCNl4f2qTPBeqVhK
yz/Ot03.:18418:0:99999:7:::
daemon:*:18390:0:99999:7:::
bin:*:18390:0:99999:7:::
sys:*:18390:0:99999:7:::
sync:*:18390:0:99999:7:::
games:*:18390:0:99999:7:::
man:*:18390:0:99999:7:::
lp:*:18390:0:99999:7:::
mail:*:18390:0:99999:7:::
news:*:18390:0:99999:7:::
uucp:*:18390:0:99999:7:::
proxy:*:18390:0:99999:7:::
```

```
www-data:*:18390:0:99999:7:::
backup:*:18390:0:99999:7:::
list:*:18390:0:99999:7:::
irc:*:18390:0:99999:7:::
gnats:*:18390:0:99999:7:::
nobody:*:18390:0:99999:7:::
```

In the case of a system compromise, if an attacker gained access to a super user account, either by attacking the account or performing privilege escalation on a standard user, they will be able to retrieve both /etc/passwd and /etc/shadow and use these two files to retrieve the passwords for every user on the system. Not good. In the scenario of a digital forensics investigation, investigators working on a forensic copy of the hard drive could use the same techniques to crack the passwords for any other users, and then log in and investigate them.

Cracking Passwords

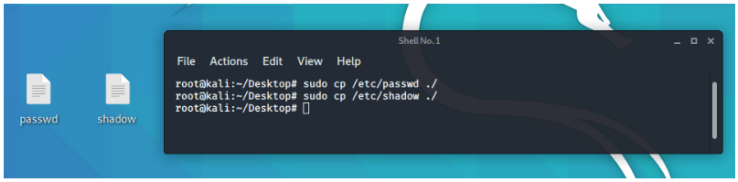
Although not really in scope of Blue Team Level 1, we decided to give you a chance to crack some passwords using the passwd and shadow files, and a tool called John The Ripper. We have created a new user named "CrackThisUser" and given it the password "bulldog!". We can confirm the user exists, and has an encrypted password by reading the /etc/shadow file using the command **`cat /etc/shadow`**.

```
vboxadd:!:18390::::::
CrackThisUser:$6$61U5mSestWnJX9dG$VnnYCVM5CjoLQvdErdaTHoh.lJKJzD/Non6gyew4CQAXkfl7gQxtN2SS1ZaFlUypvVIDw998qf1
iNfuJHYBVU1:18434:0:99999:7:::
```

We're going to be using the famous "rockyou.txt" wordlist, a file full of the most common passwords. This comes built in with Kali, but it may be in a zip container. Ours was still in it's zip, so we used gunzip to remove the .gz file type, and then copied it to our Desktop, so we don't need to keep typing out the long file path.

```
root@kali:~/Desktop# locate rockyou.txt
/usr/share/wordlists/rockyou.txt.gz
root@kali:~/Desktop# gunzip /usr/share/wordlists/rockyou.txt.gz
root@kali:~/Desktop# cp /usr/share/wordlists/rockyou.txt /root/Desktop
root@kali:~/Desktop#
```

Next we're going to copy the passwd and shadow files to our desktop for ease of use.



To combine the passwd and shadow files, we need to run the linux command unshadow, like this: **`unshadow passwd shadow > CrackMe`**. This will create a new file named CrackMe that contains the information from both input files. Now we can crack the output file!

We need to verify we have John (The Ripper) installed by running the command **`john`**. We are presented with command guidance, so we know it's installed. If not, we can use the command **`sudo apt-get install john`**. On our Desktop we should now have two files:

- Rockyou.txt
- crackme

The command we want to use to perform a dictionary attack against the encrypted passwords is: **`john CrackMe --wordlist=rockyou.txt`**. John is now working hard to identify the plain text versions of the encrypted passwords. After 2 minutes and 29 seconds, John has successfully cracked the password of our account CrackThisUser!

```
root@kali:~/Desktop# john CrackMe --wordlist=rockyou.txt
Using default input
Loaded 3 password hashes with 3 different salts (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Remaining 1 password hash
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
's 1652p/s 1652c/s 1652C/s claudia12..babyt1
bulldog!         (CrackThisUser)
1g 0:00:02:29 DONE (2020-06-21 17:36) ).006708g/s 1664p/s 1664c/s 1664C/s cabiles..brokensonnet
of the cracked passwords reliably
Session completed
root@kali:~/Desktop#
```

You'll get a chance to crack some passwords in the activity we have prepared below, good luck!

Quizzes

Activity) Password Cracking

Previous Topic

Back to Lesson

Next Topic