

Password Audits

The process of verifying/auditing passwords requires the following steps:

- 1) Log in as root/Administrator
- 2) Extract usernames and passwords from the password storage system
 - a. For Windows, this involves pulling it from the SAM
 - b. For Linux, this involves extracting from /etc/shadow, and merging with /etc/passwd
- 3) Running the passwords through a password cracking utility such as John the Ripper ophcrack

The steps in detail are as follows:

Windows

First, this is for Vista/Windows 7. This will have varying degrees of success with Windows 8 or 10, and there isn't a reliable method for this as yet. It will also work with some Windows Server versions as well.

There are a couple of options when it comes to Windows. For starters, you may download and use the ophcrack CD and copy that to a USB or burn it to a CD to boot to. This often bypasses the need to log in as Administrator and extract username/password pairs from the SAM. It is also completely automated, you can insert the CD/USB, and reboot the system.

You may also choose to install the ophcrack software onto the system, and run the analysis against the system. Again, you likely will need to be administrator, or have administrator permissions to execute this process.

The utility pwdump, currently in versions 6 and 7, can be downloaded from the web, and run in a DOS window that has been launched with administrative permissions. Once downloaded and an Administrative DOS window is open, you can run pwdump command and direct the results to a text file. This can then be brought into something like the stand-alone version of ophcrack. For example:

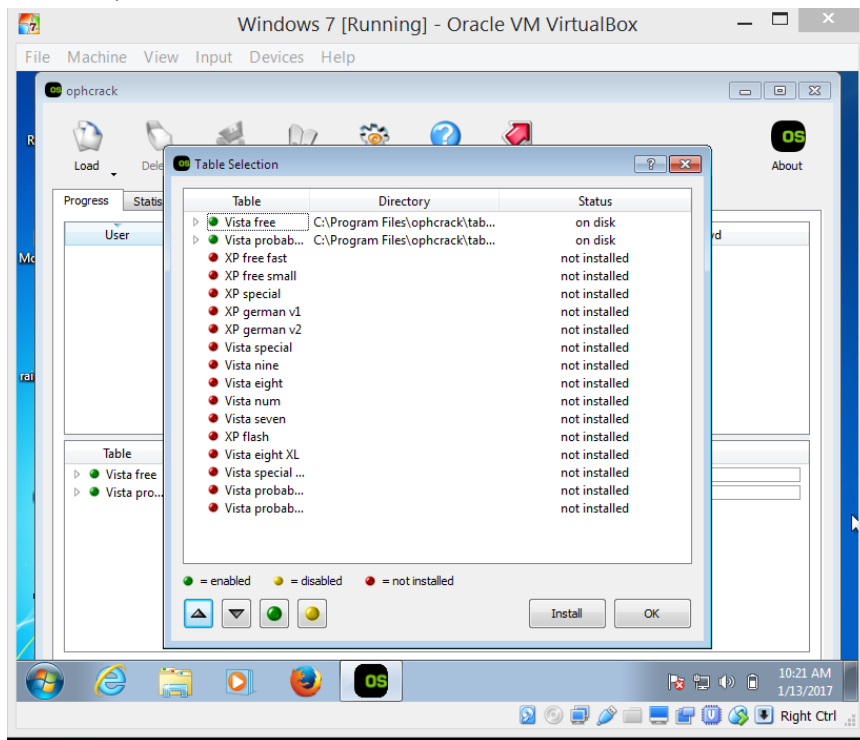
```
C:\Users\sjay\Desktop\pwdump\pwdump7 > passwords.txt
```

The passwords.txt file above can then be taken to any system and run against the stand-alone version of ophcrack.

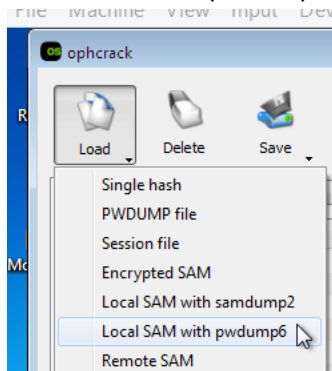
The steps shown in class are as follows:

- 1) Download the appropriate version from <http://ophcrack.sourceforge.net/download.php?type=ophcrack>
- 2) Install
- 3) Download rainbow tables (or hash tables) from places such as <http://ophcrack.sourceforge.net/tables.php>. You may find rainbow tables elsewhere, googlefu skills required. You may also use utilities such as RainbowCrack to generate your own tables, but is out of scope for this. Found here: <http://project-rainbowcrack.com/generate.htm>

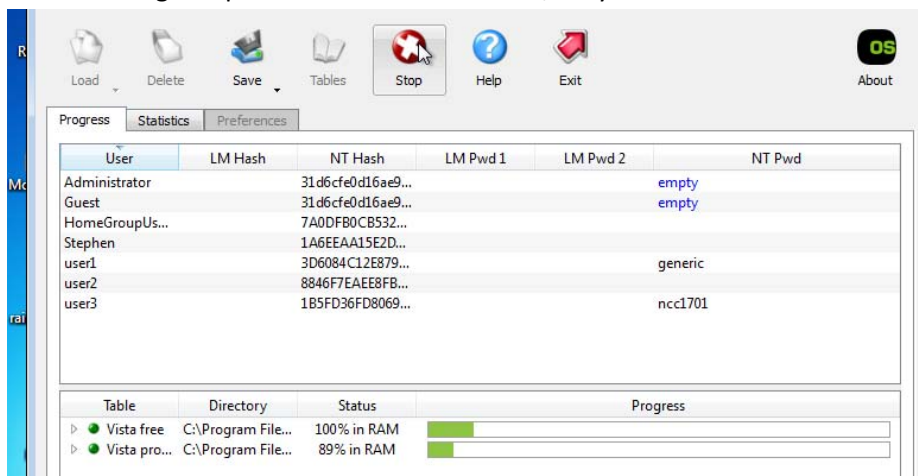
4) Launch ophcrack, and load tables



5) Load SAM. I use pwdump6 from local SAM, as it seems to work better for my installation:



6) Start cracking. As passwords are determined, they will be listed.



Cracking Passwords in Debian

Originally, all UNIX user information, including passwords were stored in the `/etc/passwd` file. This file needs to be read accessible to everyone, as it has user info such as home directory and default shell. Having the encrypted password visible for everyone to see is problematic, as hacking tools such as John the Ripper can pull this encrypted info, and crack the passwords.

As mentioned above, most of this requires to be logged in as root.

In Linux, user information, including passwords, are generally stored in the following manner:

```
ls -l /etc/passwd
-rw-r--r-- 1 root root 1360 Oct 20 2014 /etc/passwd

tail /etc/passwd
user:x:UID:GID:Full Name:/home/user:/bin/bash

ls -l /etc/shadow
-rw-r----- 1 root root 1011 Oct 20 2014 /etc/passwd
tail /etc/shadow
user:encrypted_password:$3:$4:$5:$6:$7:$8:$9
where $3 - $9 have to do with expiring password info.
```

When we look at the info above, we see that the password itself is stored in an encrypted manner in a file that only the root user has access to. As such, we need to be root to retrieve this encrypted info. We are going to also use a utility that merges the `/etc/passwd` and `/etc/shadow` files into a text file we can then run the password cracking tool John the Ripper against. We can also see that only root has access to the `/etc/shadow` file

Install John the Ripper

```
apt-get update
apt-get upgrade
apt-get install john
```

Password Cracking

One issue with Debian is the passwords, by default, use SHA512. We can tell this by checking out the preface of the passwords in `/etc/shadow`. If they start with `6`, they use SHA512, very difficult to crack. We are going to use MD5 passwords, very easy to crack.

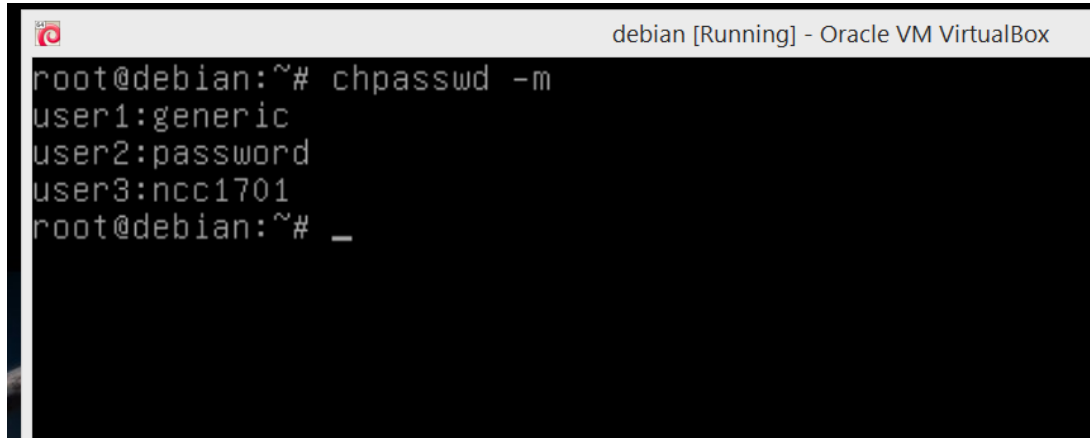
We are also going to create dummy accounts user1, user2, user3.

```
useradd user1
useradd user2
useradd user3
```

We then need to set their passwords. Again, like the Windows example above, this is a bit contrived. As mentioned above, we will be using MD5 encryption to force the hashes to be something relatively easy for JTR to crack. The `-m` switch below forces the `chpasswd` utility to use MD5 encryption.

```
chpasswd -m
```

This will require us to enter user:password info for the three users we created. There should be one line for each username/password, and no blank lines. Once you've typed those in, hit Ctrl + D to save your work, after the last line of your username/password pairs:



```
debian [Running] - Oracle VM VirtualBox
root@debian:~# chpasswd -m
user1:generic
user2:password
user3:ncc1701
root@debian:~# _
```

Finally, run the following.

```
unshadow /etc/passwd /etc/shadow > passwords.txt
john passwords.txt
```

It will run for a while, but you can check the progress. It will also spit out passwords as it discovers them. You can stop the machine, and start it again at a future date.