COMS3000/7003 Assignment 3

1. Team Members

Lilly Tran
Anthony Carrick

We both tried various experiments (rules and wordlists). We both contributed to this report equally, writing our own parts or together.

2. Tools and Dictionaries (Wordlists)

We both used John the Ripper (JtR) and its GUI shell, Johnny. We both used Kali Linux, separately, on a VM provided by VirtualBox. Li also used hashcat on the same VM.

The disadvantage with this setup was that the GPU isn't accessible from within VM so JtR Jumbo or HashCat couldn't use the GPU on CUDA so we were limited to sequential hashing attempts rather than the parallel attempts provided by the GPU. We both used Kali Linux since Matt Dobinson recommended it as it comes with all the tools necessary and plenty of included wordlists.

The advantages of using a VM is that it is an isolated environment that can't interfere with our normal day-to-day PCs and we can pause VM execution or move them between hosts.

Anthony

I gave my VM 2 GB of RAM and 2 CPUs.

I had success with:

- metasploit/password.lst - This had a good selection of popular passwords, I thought it's a good starting place
- usr/share/wordlists/rockyou.txt - This had a huge selection of passwords, so it should catch more. And it did.

But I didn't have success with:

- https://github.com/dominictarr/random-name - I chose it in case people used their name as a password.
- fern-wifi/common.txt - I chose this since it was another list of common passwords.

I tried rules:

- Wordlist
- Oldoffice
- Korelogic

- Prince mode

Lilly

I allocated my VM 4GB of RAM and 4 CPUs.
I had success with:

- usr/share/wordlists/rockyou.txt
- UserPassCombo-Jay.txt (downloaded from https://github.com/danielmiessler/SecLists/tree/master/Passwords) - 728 words of popular username and password
- Most-Popular-Letter-Passes.txt (downloaded from https://github.com/danielmiessler/SecLists/tree/master/Passwords) - 47,604 most popular letter passes.

However, I did not have success with:

- 10k-most-common.txt (downloaded from https://github.com/danielmiessler/SecLists/tree/master/Passwords) - I chose this word list because it contains 10,000 most common passwords so it seems like a useful list to try.
- Keyboard-Combinations.txt (downloaded from https://github.com/danielmiessler/SecLists/tree/master/Passwords) - this wordlist contains 9,605 words made up from keyboard patterns
- Milw0rm-dictionary.txt (downloaded from https://github.com/danielmiessler/SecLists/tree/master/Passwords) - 84,196 cracked hashes

3. Configuration Analysis

Anthony

Initially I could crack duffer5's password just by running Johnny in Single Crack mode where it tries usernames, GECOS, and word mangling rule, Single. Then I tried Johnny's Guess Password feature where you can type in a password guess and it tries it against all unknown credentials. I typed in "Password1" and "password1" and found both duffer and duffer3.

Then I tried metasploit/password.lst with some success and fern-wifi/common.txt without success. Then I tried rockyou.txt and comparatively the most successful, finding 3 passwords. By this stage I'd only found 11 passwords, or 5%, I figured that plain wordlists aren't working because the passwords themselves must be complex and secure. (Since we were asked to make them secure initially.) So I thought I'd better try using mangling rules to get numbers and punctuation, etc.

I first tried metasploit/passwords.lst with modified Wordlist rule. My modification was to add: "$2$0$[01]$[0123456789]$[!@...^&*\-=_+,./?]" to append recent years to the passwords.[1] I

figured that maybe students used a random word with the year at the end since it's a temporary password for Assignment 1. For reference, this is the command I used:

```
/usr/sbin/john --wordlist=/usr/share/wordlists/rockyou.txt
--rules=Wordlist --session=/root/.john/sessions/10-20-18-17-00-57
/root/Assignment 3/shadow9.dms
```

After 5 hours though, it still hadn't found any more. This makes me think I either use the software incorrectly or the passwords are quite complex.

Next I tried Prince mode in John the Ripper with rockyou.txt and Oldoffice rules but after an hour it still hadn't found anything. Prince mode builds chains of multiple-word password attempts with one dictionary file and mangling rules.

Finally I tried https://github.com/dominictarr/random-name with the Wordlist rule but it didn't seem to find any more. I also tried with Korelogic rule set, also with no success. This surprised me because I expected that at least some passwords would be based on the user's name, in Leetspeak or similar.

Lilly

I ran Johnny (GUI of John the Ripper) using the wordlist mode with rockyou.txt and after 30 minutes found 8 passwords which are password, Password, Password1, duffer5, Duffer, Marvin, qazwsx and !@#$%^. I then changed the wordlist used to UserPassCombo-Jay.txt and found 1q2w3e4r5t immediately but didn't find anymore after that. I also tried Most-Popular-Letter-Passes.txt wordlist and found asdzxc in half an hour. Any other wordlist I tried in the unsuccessful list did not deliver any passwords. I did not add any rules to any wordlist mode configuration because the wordlists themselves are already quite big.

I then tried to use Hashcat in Kali Linux in the same VM with some rules. I tried a combination of letters with the length of 1 to 3 using the following command in Hashcat:

```
hashcat -m 1800 -a 3 ./shadow9  -i ?a?a?a
```

I let it run for 1 hour and 6 minutes which is long enough for it to finish trying all the possible passwords of length up to 2 characters. It did not find any passwords. I checked the remaining time for it to finish and it was an additional 3 days and 22 hours so I stopped it.

Finally, I noticed the Guess Password feature in Johnny so I tried a few guesses:

1. I guessed password123 since it is a popular pattern and found one match
2. I guessed COMS3000 because it is the course code for the undergraduate and found another match
3. I guessed COMS7003 but did not find any match
4. I then tried a number of variations on the course code such as coms3000, coms7003, coms3000s22018, coms300018, coms700318, coms70032018, coms7003s22018 but did not find any match
5. I then tried assignment, ass1 and also did not find any match

6. I then remembered the lecturer mention in the lecture that some people did not change their password and use the default password given in the assignment 1 so I guessed COMS3000/7003 and found 11 matches.

4. Results

For this section, we will list the results in the list below, followed by the explanation of the cracking and why the password is not secure.

- duffer:password; “password” is going to be the first thing to try
- duffer2: Password; it is just a variation of the popular default password “password”
- duffer3:Password1; Putting “1” on the end is one of the second things to try.
- duffer4: password123; this is insecure because it is very guessable.
- duffer5:duffer5; This is just the username as the password so it’s one of the first things to try.
- duffer6: Duffer; this contains part of the username so it’s also easy to guess
- s4397013:qazwsx; Found using password.lst - this is not a dictionary word, but it’s in a password list - it’s in the list because it’s a common pattern to type.
- s4511723:!@#$%^; Found using password.lst - similar to the above, it has a small set of characters and exactly matches the first six characters on a standard keyboard Shift-Entry. It’s weak because it’s predictable.
- s4495817:1q2w3e4r5t; via rockyou.txt - this word in the list looks secure but actually has been “pre-mangalled” - it’s a common pattern one might type on a standard keyboard to get desired entropy.
- s4510863:asdzxc; via rockyou.txt - another common pattern that’s easy to type and remember. This is not a real word, but is still found because of its likelihood of use.
- s4390551:hunter2; via rockyou.txt - this is just a real word, plus a number. I should have been able to find this via rule processing, but it’s in a wordlist anyway. This actually makes it even easier to find and less secure.
- s4370752: Marvin; this is just the name of a person (it could be the name of the user) and since a list of all names is easy to be compiled or found on the internet, this password is easy to be cracked. Also if it is indeed the name of the user, it can even be found by guessing.
- s4425286: COMS3000; this is the course code of this course and part of the default given password so it is easy to guess it.
- s4315128, s4541010, s4384569, s4490071, s4496087, s4500036, s4488954, s4528082, s4495508, s4417496, s4354619 has the same password COMS3000/7003 which is the default password given in the assignment 1.

References:

[1]"John Users - New john.conf rules for password cracking", *OpenWall Mailing List*, 2009. [Online]. Available: https://www.openwall.com/lists/john-users/2009/03/28/2. [Accessed: 20- Oct- 2018].