Lab 03 Specifications

Last update - 2016-05-28, 18:11Z

Due: Specified on the Blackboard assignment page. **Late submissions will automatically be marked as late and NOT GRADED**. Please make sure you submit your assignment far in advance of the stated deadline, so that you avoid losing credit due to bad networks, etc. It is recommended that you start early and if you have questions, ask them in the course Blackboard Discussion forum. If you wait until the last minute to ask questions, it's very likely that nobody will answer you in time. It is also recommended that you turn in something, even if it's not complete. Even if you weren't able to complete the assignment, taking the time to explain what you tried and what problems you faced will likely prevent you from getting a zero.

Grading: Grading is somewhat subjective in this assignment, on a scale of 0 to 10, roughly equivalent to a standard grading scale where 10 would be an A+ (Excellent), 9 would be a B+/A- (Good), 8 would be a C+/B- (Fair), etc. In this particular assignment, if you do all that is asked of you, you should receive a 10. If you neglect, or poorly explain 2 or more minor items, you should receive a 9, and so on.

The primary purpose of this third Lab Assignment is to give you hands-on experience in issues related to password security, which serves as the primary basis for access control to computing systems. In this assignment you will use a popular password cracking software system, *John the Ripper* (*JtR*, or simply, *john*). This system is used by many administrators to audit their own systems. Needless to say, it is highly unethical to use this for the purpose of hacking into a system without permission of the system owners.

You will run *john* with a provided password file, and then you will experiment by generating some of your own passwords and allowing *john* to try to crack them. Then, you will explore some online password cracking environments and password generators.

The steps, outlined below, are to

- Either use the same VM you used in Lab Assignment #01, or import a new one
- In the VM
    - Run *john* on a password file prepared to you
    - Create your own passwords with *openssl*, some easy, some hard, put them in a password file, and try to have john crack them.
- In a web browser
    - Go to hash generator pages and create your own passwords, some easy, some hard
    - Go to password cracking site that uses a "rainbow table" and try these passwords

In the description of this lab, I have annotated areas where I expect something from you by **highlighting requirements in Red Bold**. To complete this assignment, you will (this is repeated at the bottom of the assignment)

- provide thorough explanations in areas that I request
- provide images of the scenarios that I request

If you run into problems that you cannot resolve, you should first utilize the Blackboard Discussion forum for this class and seek help. If you are still unable to resolve the problem, you should **provide a very clear explanation of what you have tried, and what has gone wrong**.

Please note that in the following instructions, I am refraining from telling you exactly how to perform each step. I am simply telling you what you need to do and, where I think it might be necessary, providing you with some additional guidance. In some cases, you will have to read documentation and figure some things out for yourself. I am also very aware that many of you are not familiar with Linux. I am trying to keep this simple enough that it won't be too hard for you, but at the same time I think it's good for me to force you to figure out a few things on your own.

Downloading the Virtual Machine

You may use the same VM that you used in Lab Assignment #02. If you do, it has the advantage that the terminal prompt is already set up with your name in it.

If you choose to download a new one (it's exactly the same as the one you've downloaded before), refer to the *Quick Notes on Installation of VirtualBox, and Import/Usage of the Virtual Machine* document from Lab Assignment #02 to import it and to set up the terminal command line prompt so that it has your name in it.

Using *john*

John the Ripper is available both in a free and a commercial form -- http://www.openwall.com/john/ - but I have already installed it for you in the VM. Documentation (not the best in the world) is available at http://www.openwall.com/john/doc/. Note that within the document there are links to OPTIONS and EXAMPLES.

Once you boot the virtual computer, open up a terminal or two, and cd to the *JtR* directory, where you will do all your work.

A fictitious sample password file, *hackme.txt,* is available in this directory. You should look at it and understand the basic structure (you will only be using the first two entries of each record). This information is available all over the Internet (and presented in course slides).

Run *john* with this password file as input and **take a snapshot of it after it has cracked a few passwords**.

The documentation can be a little confusing, so here are some general guidelines. I am not explaining this fully to you, but it is hopefully enough to help you understand the documentation a little better

- *john* will keep running - possibly for days - until you hit CTRL-C (don't do this yet)
- Passwords that are cracked are stored in the file *john.pot*. This is a binary file, and you cannot read it with a standard text editor. You can read the contents, however, by using the **`--show`** option. **Let *john* run for about five minutes, stop it with CTRL-C, and report on the number of passwords cracked.** You can simply count these, or if you want to be clever, you can pipe the command output through the *wc* command.
- This is where it can get confusing - if you run *john* again, using the same *hackme.txt* as input, it will start where it left off. It won't bother cracking the passwords that are already in the file *john.pot*. If you find that you need to start from scratch for some reason, you should delete *john.pot*.

You will need to review the documentation (and maybe do a little Googling) for this next part. First, look at the contents of the file *password.lst*, and then run *john* using this file as an argument for the **`--wordlist`** option. **You should delete the existing john.pot before this step! Take a snapshot of the terminal after you have run john with the `--wordlist` option, and provide an explanation about what you have just done.**

Congratulations, you are now a password cracker!!

Next, you should explore the effects of using weak and strong passwords. You may generate your own encrypted passwords using *openssl*, as described in https://www.openssl.org/docs/manmaster/apps/passwd.html

It can be invoked as simply as

```
openssl passwd -salt xx secret
```

Make sure you understand what this is doing. Try it twice to verify that you get the same encryption. Now, try it with a different salt, and try it at least twice. Then, try the same thing without the salt argument (omit `-salt` plus its argument), and try this several times. You should see that a random salt is being injected each time.

Now that you know how to generate encrypted passwords, you can use *john* to test them. Again, before proceeding go ahead and delete the existing john.pot.

You should create a new password file that has only two records. I recommend that you copy the existing *hackme.txt* to another file, then delete all but two lines. In the following tests you can just paste in your newly-generated password encryption into the right places in this record.

In the first test, of easy passwords, you should generate encryptions for two easy passwords, paste them into the right place in your password file, and run *john* on this password file. **You should create a screen dump of the terminal that shows you using *cat* on your two-line password file, then running *john* to crack the two passwords.**

Then, you should repeat the above with passwords that you think are secure and not easily cracked. **Let *john* run for at least a minute on the new password file, and simply report the two passwords that you created.**

This concludes the VM portion of this lab. You may shut it down.

Using online tools to generate and crack passwords

In this section, you will generate passwords with one of the strongest hash functions available - SSH-256. Then, you will enter it into an online cracker to test its strength.

Go to http://www.xorbin.com/tools/sha256-hash-calculator and create the SHA256 hash of a common word, then, try to crack it at https://crackstation.net/. **Take a snapshot of this page after the result has been delivered.** Then, do the same thing with a passphrase chosen so that it evades cracking and, again **Take a snapshot of this page after the result has been delivered and report on what your passphrase was.**

Do you think we will have better luck with a stronger hash? Try the same experiment with an SHA512 hash - you may generate these hashes at http://passwordsgenerator.net/sha512-hash-generator/. Do you think you will have better security with a stronger hash?

As with the SHA256 hash, you should take snapshots of the pages that try to crack the password, and report on what your passphrase was.

Finally, discuss the method that https://crackstation.net/ uses, and why it works so well for some passwords, but not for others.

Submission of assignment

In the description of this lab, I annotated areas where I expect something from you by **highlighting requirements in Red Bold**. To complete this assignment, you will

- provide thorough in areas that I request
- provide images of the scenarios that I request

Please put everything in a single PDF or Word (or OpenOffice) document.

If you run into problems that you cannot resolve, you should first utilize the Blackboard Discussion forum for this class and seek help. If you are still unable to resolve the problem, you should **provide a very clear explanation of what you have tried, and what has gone wrong**.

Note that if you are trying to get a screenshot of your VM, you can simply use the Host+E command, where “Host” for most of you would be Right CTRL. In other words, Right-CTRL+E should allow you to get a screen shot of your current VM.