FORENSICS
LAB SERIES

Lab 18: Recovering Passwords

Material in this Lab Aligns to the Following Certification Domains/Objectives
Computer Hacking Forensic Investigator (CHFI) Objectives
14: Application Password Crackers

Document Version: 2016-08-17

Contents

Introduction

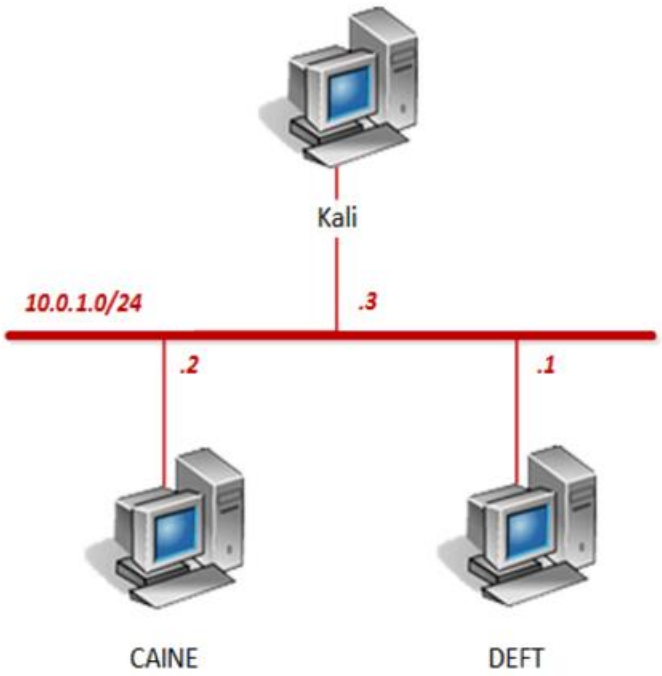
This lab will introduce the concept of recovering passwords from password protected documents, as this is one of the skills necessary in forensic examination. Different tools may be used to attempt to recover these passwords and open the documents.

Objective

In this lab, you will be conducting forensic practices using various tools. You will be performing the following tasks:

1. Recovering Password Protected PDFs
2. Recovering Password Protected ZIPs

Pod Topology



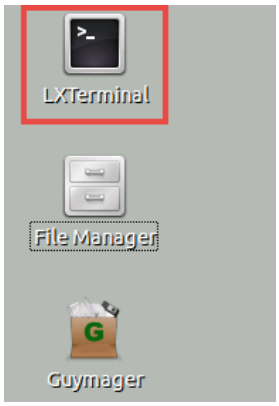
Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account (if needed)	Password (if needed)
DEFT	10.0.1.1	deft	password
CAINE	10.0.1.2	caine	
Kali	10.0.1.3	root	toor

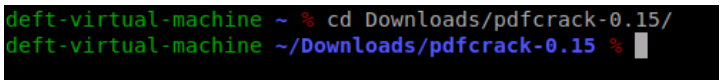
1 Recovering Password Protected PDFs

1. Click on the **DEFT** graphic on the *topology page* to open the VM.
2. Open a new terminal by double-clicking on the **LXTerminal** icon located on the *Desktop*.



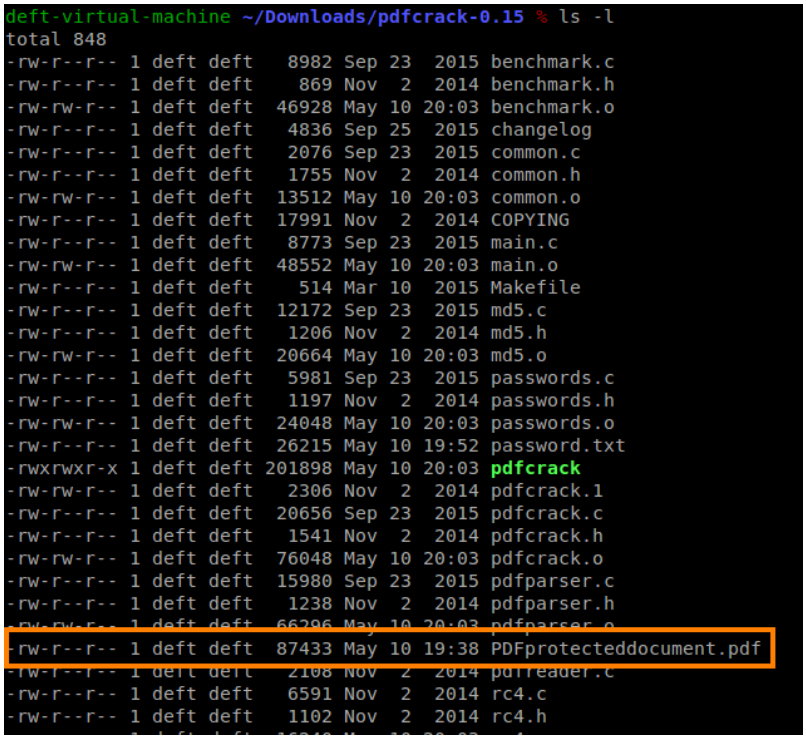
3. Using the terminal, navigate to the **/home/deft/Downloads/pdfcrack-0.15/** directory by typing the command below followed by pressing the **Enter** key.

```
cd Downloads/pdfcrack-0.15/
```



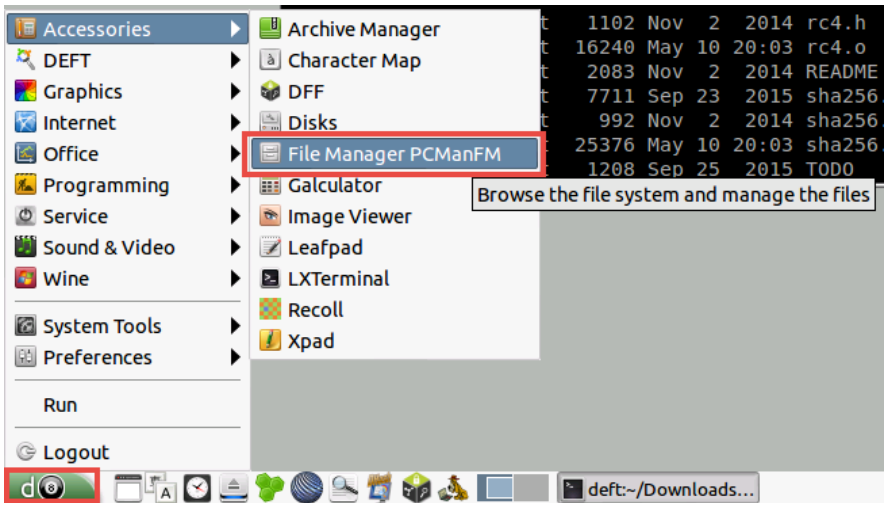
4. Enter the command below to list the files in the current directory.

```
ls -l
```

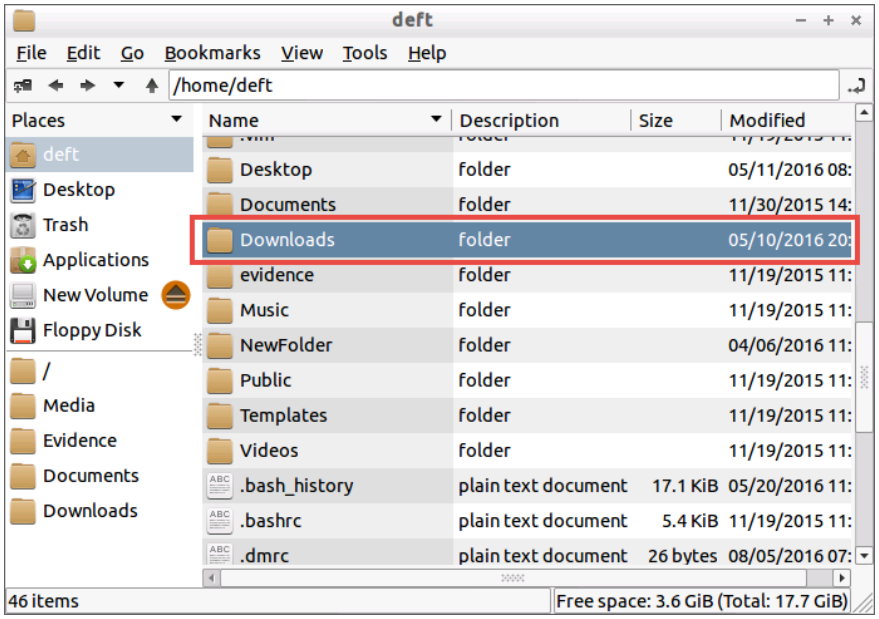


Notice the PDFprotecteddocument.pdf file. This *PDF* file is protected with a password.

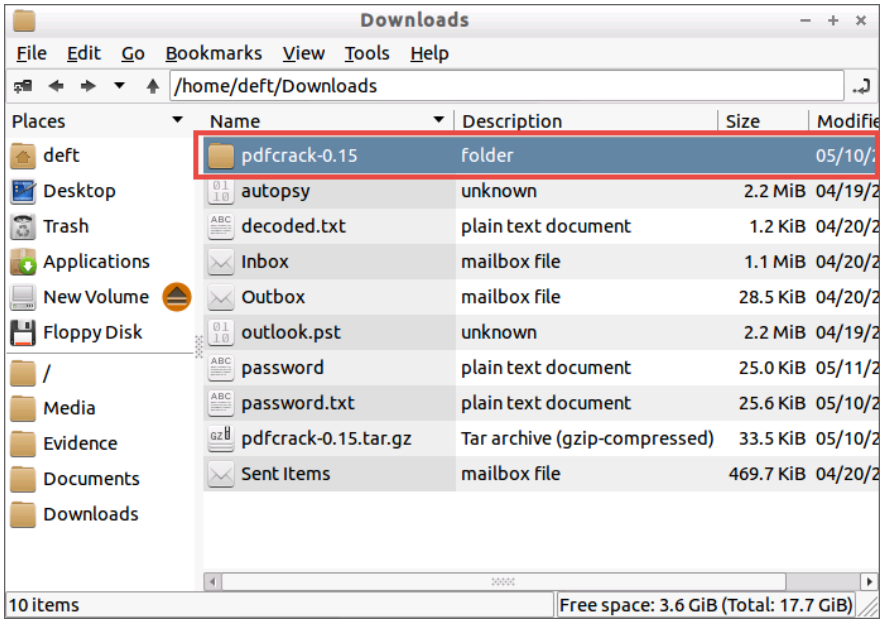
5. Open the file manager application by navigating to **Menu > Accessories > File Manager PCManFM**.



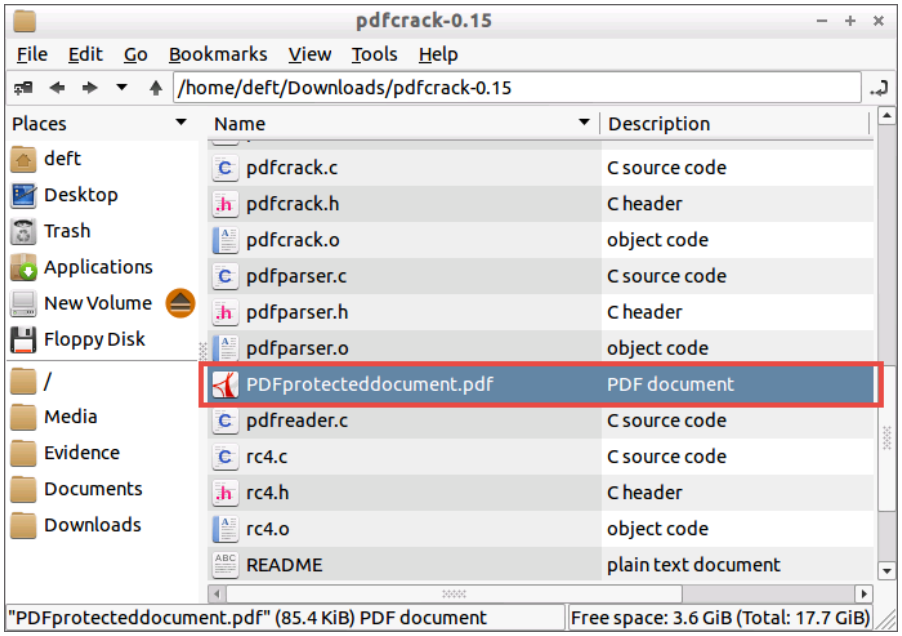
6. Using the file manager, navigate to the **/home/deft/Downloads** directory.



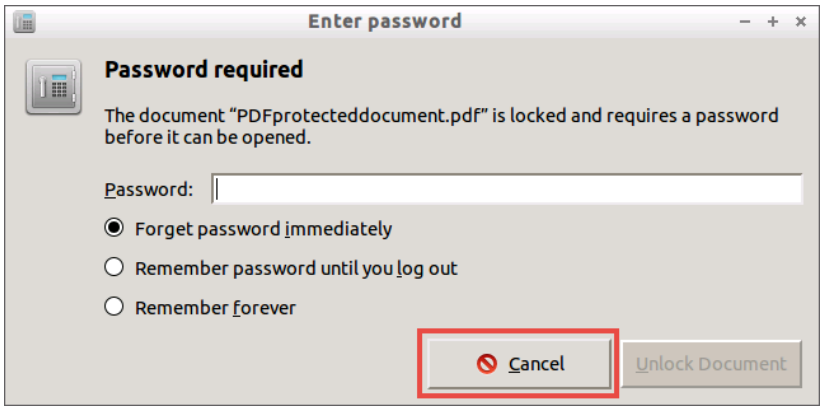
7. Double-click on the **pdfcrack-0.15** folder.



8. Scroll down and double-click on the **PDFprotecteddocument.pdf** file to open it.



9. Notice an *Enter password* dialog window appears asking for a password to open the *PDF*. Click **Cancel**.



10. Change focus to the **terminal**.

11. Using the terminal, type the command below followed by pressing the **Enter** key to list the available arguments available for the *pdfcrack* tool.

```
./pdfcrack
```

```
deft-virtual-machine ~/Downloads/pdfcrack-0.15 % ./pdfcrack
Usage: ./pdfcrack -f filename [OPTIONS]
OPTIONS:
-b, --bench             perform benchmark and exit
-c, --charset=STRING    Use the characters in STRING as charset
-w, --wordlist=FILE     Use FILE as source of passwords to try
-n, --minpw=INTEGER     Skip trying passwords shorter than this
-m, --maxpw=INTEGER     Stop when reaching this passwordlength
-l, --loadState=FILE    Continue from the state saved in FILENAME
-o, --owner             Work with the ownerpassword
-u, --user              Work with the userpassword (default)
-p, --password=STRING   Give userpassword to speed up breaking
                        ownerpassword (implies -o)
-q, --quiet             Run quietly
-s, --permutate         Try permutating the passwords (currently only
                        supports switching first character to uppercase)
-v, --version           Print version and exit
deft-virtual-machine ~/Downloads/pdfcrack-0.15 %
```

12. Perform a benchmark on the **PDFprotecteddocument.pdf** file using the **pdfcrack** tool. Enter the command below.

```
./pdfcrack PDFprotecteddocument.pdf -b
```

```
deft-virtual-machine ~/Downloads/pdfcrack-0.15 % ./pdfcrack PDFprotecteddocument
.pdf -b
Benchmark:      Average Speed (calls / second):
SHA256 (fast):          1749869.1
SHA256 (slow):          898502.7

MD5:                    2736775.9
MD5_50 (fast):          113356.7
MD5_50 (slow):          108421.4

RC4 (40, static):       1017953.5
RC4 (40, no check):     997740.5
RC4 (128, no check):    920357.2

Benchmark:      Average Speed (passwords / second):
PDF (40, user):         740736.7
PDF (40, owner):        367086.1
PDF (40, owner, fast):  833333.3

PDF (128, user):        32204.3
PDF (128, owner):       15552.9
PDF (128, owner, fast): 32677.9
deft-virtual-machine ~/Downloads/pdfcrack-0.15 %
```

Using different methods, the output shows how fast the tool will run different passwords against the file in an attempt to break the password.

13. There is a preloaded wordlist in the current directory called *password.txt*. Briefly view the content of the wordlist file by entering the command below.

```
cat password.txt | less
```

```
#!comment: This list has been compiled by Solar Designer of Openwall Project,
#!comment: http://www.openwall.com/wordlists/
#!comment:
#!comment: This list is based on passwords most commonly seen on a set of Unix
#!comment: systems in mid-1990's, sorted for decreasing number of occurrences
#!comment: (that is, more common passwords are listed first).  It has been
#!comment: revised to also include common website passwords from public lists
#!comment: of "top N passwords" from major community website compromises that
#!comment: occurred in 2006 through 2010.
#!comment:
#!comment: Last update: 2011/11/20 (3546 entries)
123456
12345
password
password1
123456789
12345678
1234567890
abc123
computer
tigger
1234
qwerty
:
```

With the *less* command, use the **Enter** key to skip to the next line item of the list or use the **spacebar** to skip by page. When finished analyzing the file, press the **q** character to quit.

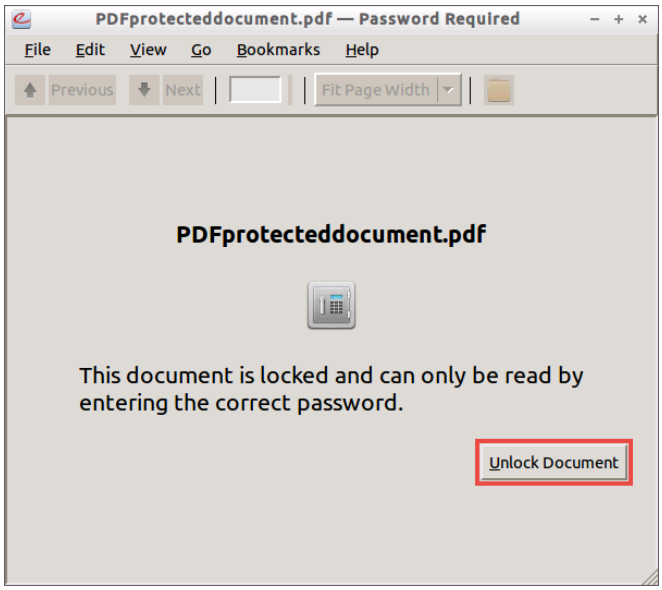
14. Use the *pdfcrack* tool in conjunction with the wordlist file to try common passwords against it. Enter the command below.

```
./pdfcrack -f PDFprotecteddocument.pdf --wordlist=password.txt
```

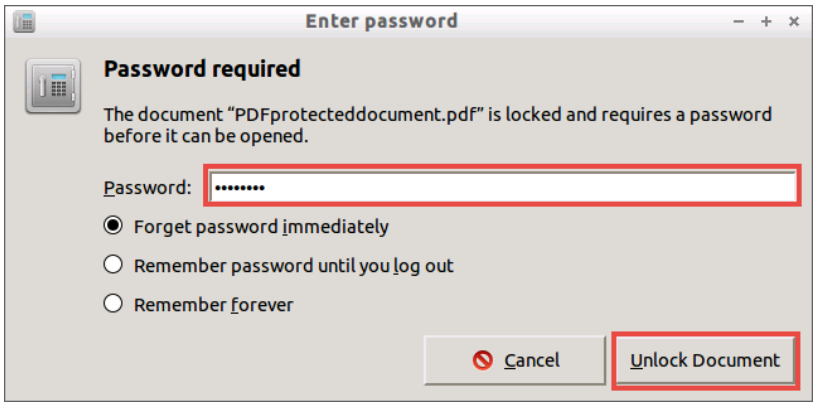
```
deft-virtual-machine ~/Downloads/pdfcrack-0.15 % ./pdfcrack -f PDFprotecteddocum
ent.pdf --wordlist=password.txt

PDF version 1.6
Security Handler: Standard
V: 2
R: 3
P: -1060
Length: 128
Encrypted Metadata: True
FileID: f481f51c58080747a909f7bd48d83a9c
U: a396abe7667b6c98e211b0c4b9a276da0000000000000000000000000000000000
O: 408b37bcf12da873d7f2840f3c1b917a023961ded4c8164d38e46e9655e66775
found user-password: 'password'
deft-virtual-machine ~/Downloads/pdfcrack-0.15 %
```

Notice the tool quickly cracked the password as the word "*password*".

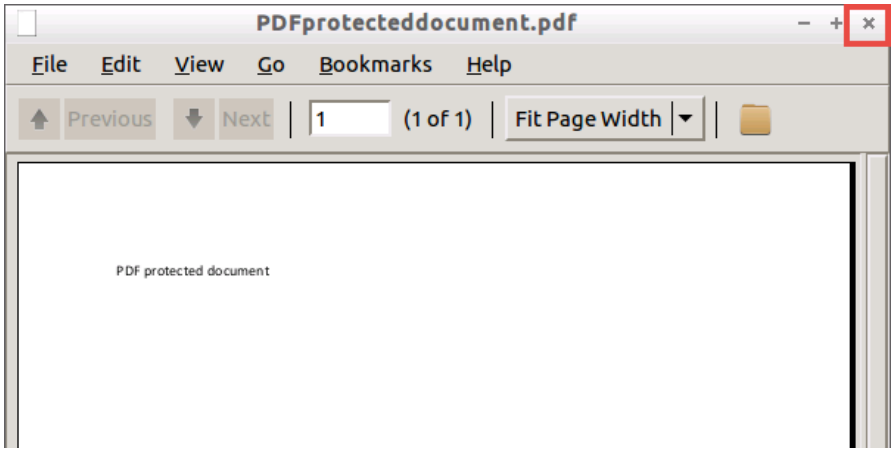
15. Change focus to the **PDFprotecteddocument.pdf** file window.
16. Click **Unlock Document**.



17. Verify whether the *pdfcrack* tool supplied the correct password. Type **`password`** in the *Password* field and click **Unlock Document**.

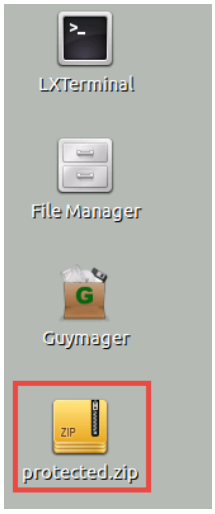


18. Notice the *PDF* document opens. View the contents and close the window.

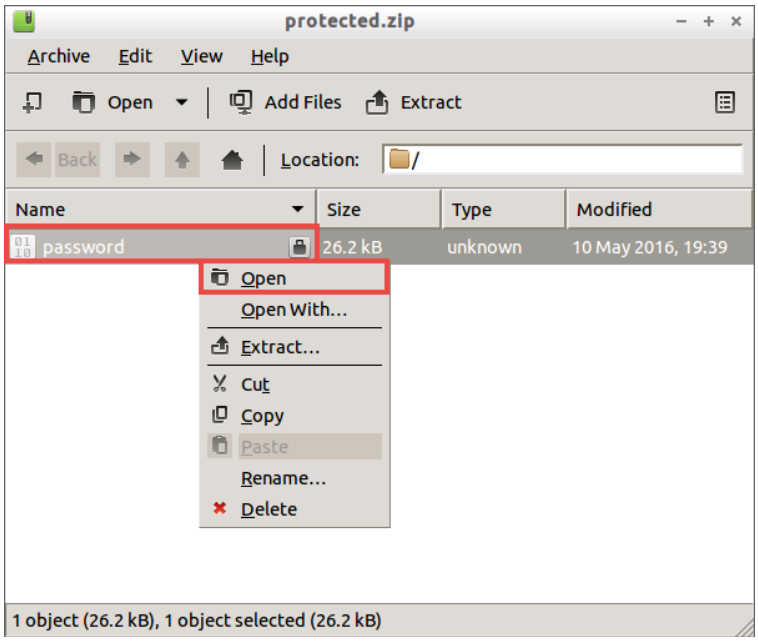


2 Recovering Password Protected ZIPs

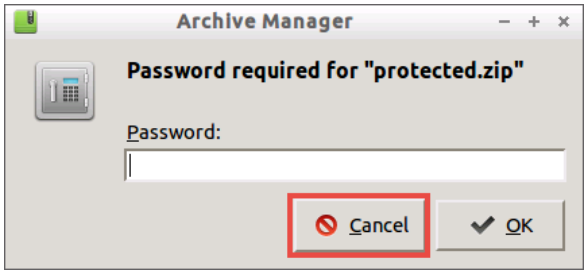
1. Another common type of protected file types are *ZIP* files. Change focus to the Desktop and double-click on the **protected.zip** file.



2. In the *protected.zip* file window, right-click on the **password** file and click **Open**.



3. Notice it asks for a password, click **Cancel**.



4. Change focus to the **terminal**.
5. Using the terminal, enter the command below to view the available arguments for a tool called *fcrackzip*.

```
fcrackzip -h
```

```
deft-virtual-machine ~/Downloads/pdfcrack-0.15 % fcrackzip -h

fcrackzip version 1.0, a fast/free zip password cracker
written by Marc Lehmann <pcg@goof.com> You can find more info on
http://www.goof.com/pcg/marc/

USAGE: fcrackzip
         [-b|--brute-force]            use brute force algorithm
         [-D|--dictionary]             use a dictionary
         [-B|--benchmark]              execute a small benchmark
         [-c|--charset characterset]   use characters from charset
         [-h|--help]                   show this message
         [--version]                   show the version of this program
         [-V|--validate]               sanity-check the algortihm
         [-v|--verbose]                be more verbose
         [-p|--init-password string]   use string as initial password/file
         [-l|--length min-max]         check password with length min to max
         [-u|--use-unzip]              use unzip to weed out wrong passwords
         [-m|--method num]             use method number "num" (see below)
         [-2|--modulo r/m]             only calculcate 1/m of the password
         file...                    the zipfiles to crack
```

6. Use this tool in an attempt to crack the password on the *protected.zip* file. First run a benchmark test against it using *fcrackzip*. Enter the command below.

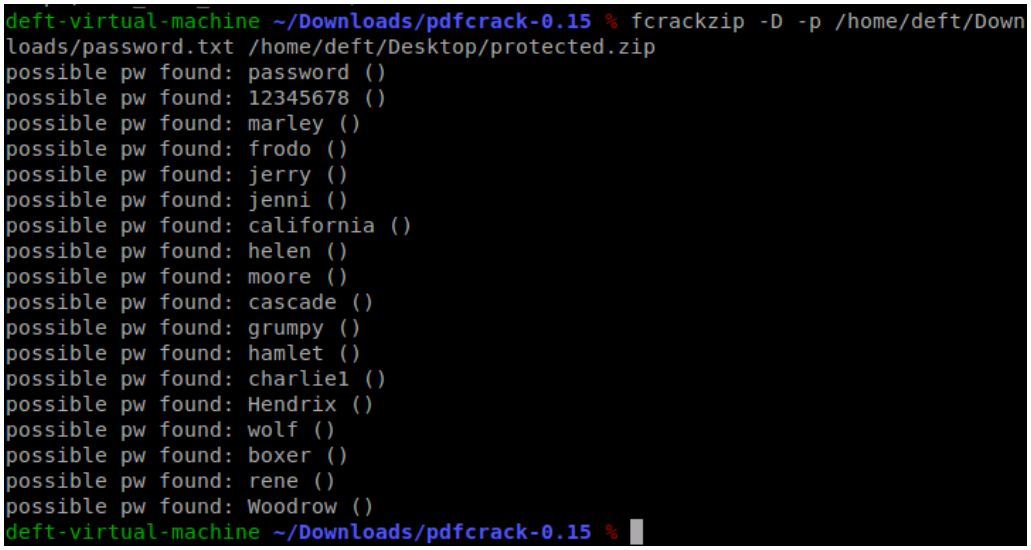
```
fcrackzip -B /home/deft/Desktop/protected.zip
```

```
deft-virtual-machine ~/Downloads/pdfcrack-0.15 % fcrackzip -B /home/deft/Desktop
/protected.zip
 cpmask: (skipped)
 zip1: cracks/s = 6855156
*zip2, USE_MULT_TAB: cracks/s = 7018410
deft-virtual-machine ~/Downloads/pdfcrack-0.15 %
```

Notice that the benchmark indicates the tool will do *6855156* cracks per second against the file. Note that these values may slightly differ.

7. The method for this crack attempt will be the same way as done for the *PDF* file. Enter the command below to use *fcrackzip* in conjunction with the *password.txt* wordlist.

```
fcrackzip -D -p /home/deft/Downloads/password.txt
/home/deft/Desktop/protected.zip
```



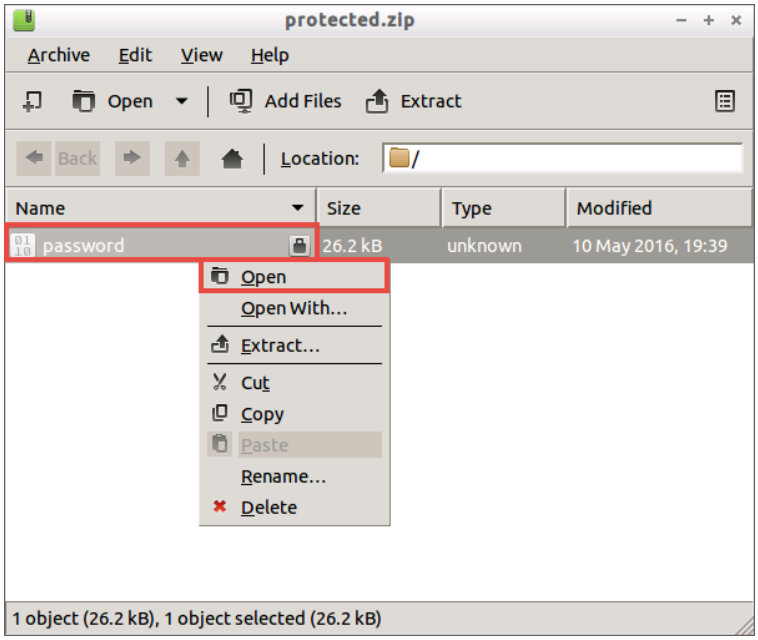
Command Breakdown:

-D = means to use dictionary/wordlist
-p = means to use strings as initial password

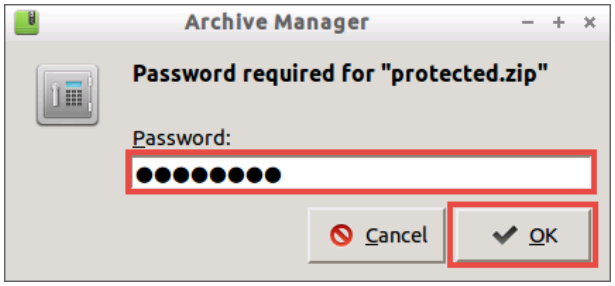
Notice that when using the dictionary style attack, *fcrackzip* reports back with 18 possible passwords.

8. Change focus to the **protected.zip** file window.

9. Right-click on **password** and click **Open**.



10. Try the first password from the *fcrackzip* output. Type **`password`** into the *Password* field and click **OK**.



11. Notice the *ZIP* file accepts the password and opens the file. Briefly view the contents of the file and close the window.

12. Close all **PC Viewers** and end the reservation to complete the lab.

