

Christopher Mayol

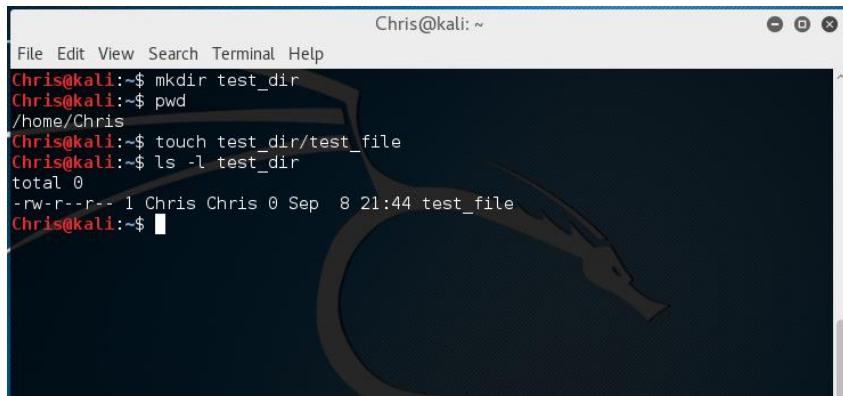
1) Changing directory permissions

We can change the permissions of a directory as well.

Read- The user can look at the filenames in this directory.

Write- The user can add/remove files from this directory.

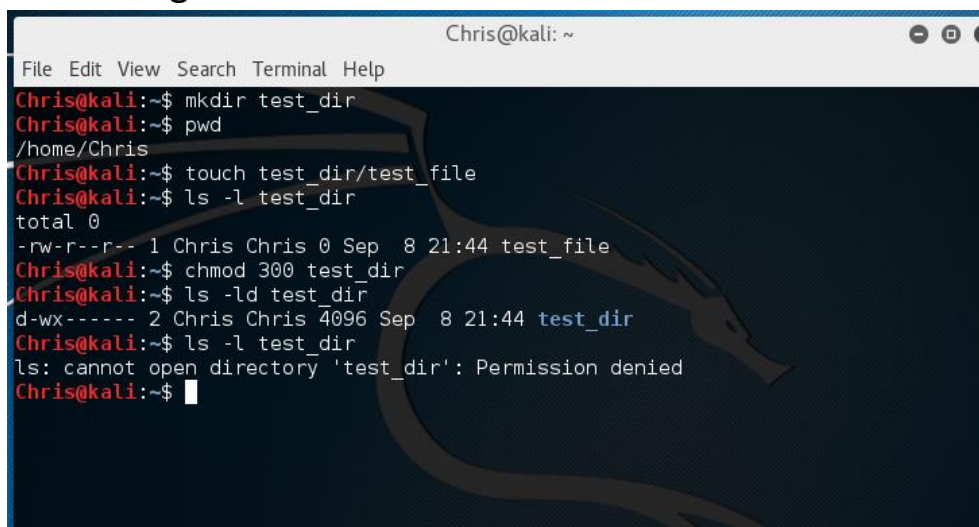
Execute- The user can execute the files of this directory.

A terminal window titled 'Chris@kali: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The user runs 'mkdir test_dir', 'pwd' (showing /home/Chris), 'touch test_dir/test_file', and 'ls -l test_dir'. The output of 'ls -l test_dir' shows a single file 'test_file' with permissions '-rw-r--r--' owned by 'Chris Chris' and a size of '0' bytes.

```
Chris@kali:~$ mkdir test_dir
Chris@kali:~$ pwd
/home/Chris
Chris@kali:~$ touch test_dir/test_file
Chris@kali:~$ ls -l test_dir
total 0
-rw-r--r-- 1 Chris Chris 0 Sep  8 21:44 test_file
Chris@kali:~$
```

a) We added a test_file to our test_dir

The -d flag in the ls command lists the directories themselves.

A terminal window titled 'Chris@kali: ~' showing the continuation of the previous steps. The user runs 'chmod 300 test_dir', 'ls -ld test_dir', and 'ls -l test_dir'. The 'ls -ld test_dir' command shows the directory 'test_dir' with permissions 'd-wx-----' and size '4096' bytes. The subsequent 'ls -l test_dir' command results in an error: 'ls: cannot open directory 'test_dir': Permission denied' because the read permission for the directory has been removed.

```
Chris@kali:~$ chmod 300 test_dir
Chris@kali:~$ ls -ld test_dir
d-wx----- 2 Chris Chris 4096 Sep  8 21:44 test_dir
Chris@kali:~$ ls -l test_dir
ls: cannot open directory 'test_dir': Permission denied
Chris@kali:~$
```

We remove the read permission from the test_dir and then we tried to list all of its contents, but we get a “Permission denied” because we are no longer allowed to view the filenames of the test_dir.

```
Chris@kali: ~  
File Edit View Search Terminal Help  
Chris@kali:~$ ls -ld test_dir  
d-wx----- 2 Chris Chris 4096 Sep  8 21:44 test_dir  
Chris@kali:~$ ls -l test_dir  
ls: cannot open directory 'test_dir': Permission denied  
Chris@kali:~$ chmod 755 test_dir  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ echo 'Testing write permissions' > test_file  
Chris@kali:~/test_dir$ cat test_file  
'Testing write permissions'  
Chris@kali:~/test_dir$ cd ..  
Chris@kali:~$ chmod 500 test_dir  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ ls -l  
total 4  
-rw-r--r-- 1 Chris Chris 32 Sep  8 21:56 test_file  
Chris@kali:~/test_dir$ echo 'Adding another line' >> test_file  
Chris@kali:~/test_dir$ cat test_file  
'Testing write permissions'  
'Adding another line'  
Chris@kali:~/test_dir$ rm test_file  
rm: cannot remove 'test_file': Permission denied  
Chris@kali:~/test_dir$ cd ..  
Chris@kali:~$ chmod 755 test_dir  
Chris@kali:~$
```

b) We are allowed to modify the content of the file because we have write permission for the “test_file”, but since we no longer have the write permission for our test_dir we are not allowed to delete the test_file.

c)

```
Chris@kali: ~  
File Edit View Search Terminal Help  
ls: cannot open directory 'test_dir': Permission denied  
Chris@kali:~$ chmod 755 test_dir  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ echo 'Testing write permissions' > test_file  
Chris@kali:~/test_dir$ cat test_file  
'Testing write permissions'  
Chris@kali:~/test_dir$ cd ..  
Chris@kali:~$ chmod 500 test_dir  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ ls -l  
total 4  
-rw-r--r-- 1 Chris Chris 32 Sep  8 21:56 test_file  
Chris@kali:~/test_dir$ echo 'Adding another line' >> test_file  
Chris@kali:~/test_dir$ cat test_file  
'Testing write permissions'  
'Adding another line'  
Chris@kali:~/test_dir$ rm test_file  
rm: cannot remove 'test_file': Permission denied  
Chris@kali:~/test_dir$ cd ..  
Chris@kali:~$ chmod 755 test_dir  
Chris@kali:~$ ls -l test_dir  
total 4  
-rw-r--r-- 1 Chris Chris 58 Sep  8 21:57 test_file  
Chris@kali:~$
```

The test_file was never removed from our test_dir.

```
Chris@kali: ~  
File Edit View Search Terminal Help  
'Testing write permissions'  
Chris@kali:~/test_dir$ cd ..  
Chris@kali:~$ chmod 500 test_dir  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ ls -l  
total 4  
-rw-r--r-- 1 Chris Chris 32 Sep  8 21:56 test_file  
Chris@kali:~/test_dir$ echo 'Adding another line' >> test_file  
Chris@kali:~/test_dir$ cat test_file  
'Testing write permissions'  
'Adding another line'  
Chris@kali:~/test_dir$ rm test_file  
rm: cannot remove 'test_file': Permission denied  
Chris@kali:~/test_dir$ cd ..  
Chris@kali:~$ chmod 755 test_dir  
Chris@kali:~$ ls -l test_dir  
total 4  
-rw-r--r-- 1 Chris Chris 58 Sep  8 21:57 test_file  
Chris@kali:~$ chmod 600 test_dir  
Chris@kali:~$ ls -l test_dir  
ls: cannot access 'test_dir/test_file': Permission denied  
total 0  
-???????? ? ? ? ? ? ? test_file  
Chris@kali:~$
```

Since we don't have execute permission for our test_dir, ls cant access the test_file.

2)

```
Chris@kali: ~  
File Edit View Search Terminal Help  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ touch another_file  
Chris@kali:~/test_dir$ ls  
another_file test_file  
Chris@kali:~/test_dir$ rm -i test_file ; cd ..  
rm: remove regular file 'test_file'? y  
Chris@kali:~$
```

First we prompt the user to check if he wants to remove the file with the "rm -i test_file" command, then the semicolon tells the terminal we want to run another command, "cd .." takes us back to the parent directory of the current directory.

```
Chris@kali: ~  
File Edit View Search Terminal Help  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ touch another_file  
Chris@kali:~/test_dir$ ls  
another_file  test_file  
Chris@kali:~/test_dir$ rm -i test_file ; cd ..  
rm: remove regular file 'test_file'? y  
Chris@kali:~$ rm -i test_dir  
rm: cannot remove 'test_dir': Is a directory  
Chris@kali:~$
```

We try to remove the test_dir with the “rm” command

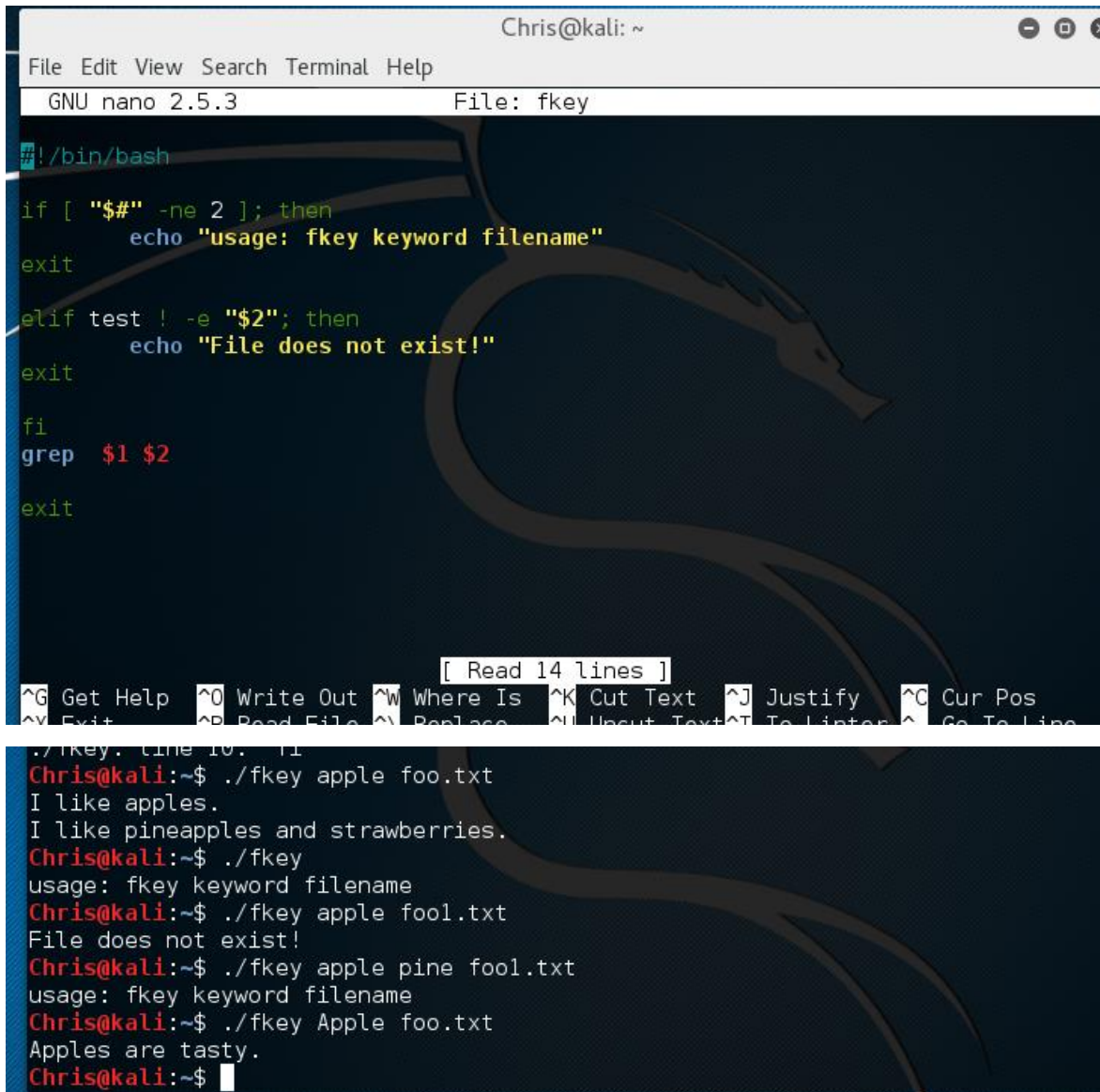
But we get an error because the rm command by default does not remove directories. We have to use the “-r” flag.

```
Chris@kali: ~  
File Edit View Search Terminal Help  
Chris@kali:~$ cd test_dir  
Chris@kali:~/test_dir$ touch another_file  
Chris@kali:~/test_dir$ ls  
another_file  test_file  
Chris@kali:~/test_dir$ rm -i test_file ; cd ..  
rm: remove regular file 'test_file'? y  
Chris@kali:~$ rm -i test_dir  
rm: cannot remove 'test_dir': Is a directory  
Chris@kali:~$ rm -ir test_dir  
rm: descend into directory 'test_dir'? y  
rm: remove regular empty file 'test_dir/another_file'? y  
rm: remove directory 'test_dir'? y  
Chris@kali:~$
```

We use the recursive flag -r in the rm command to recursively remove each file/directory inside the specified directory and then removing the target directory.

We should always prompt the user every time using the rm command because in case of typos.

3)Fkey script



```
Chris@kali: ~  
File Edit View Search Terminal Help  
GNU nano 2.5.3 File: fkey  
#!/bin/bash  
  
if [ "$#" -ne 2 ]; then  
    echo "usage: fkey keyword filename"  
    exit  
  
elif test ! -e "$2"; then  
    echo "File does not exist!"  
    exit  
  
fi  
grep $1 $2  
  
exit
```

[Read 14 lines]

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^Y Exit ^P Read File ^\ Replace ^U UnCut Text ^T To Linter ^_ Go To Line

```
./fkey: line 10: ^I  
Chris@kali:~$ ./fkey apple foo.txt  
I like apples.  
I like pineapples and strawberries.  
Chris@kali:~$ ./fkey  
usage: fkey keyword filename  
Chris@kali:~$ ./fkey apple fool.txt  
File does not exist!  
Chris@kali:~$ ./fkey apple pine fool.txt  
usage: fkey keyword filename  
Chris@kali:~$ ./fkey Apple foo.txt  
Apples are tasty.  
Chris@kali:~$
```

The third line of the foo.txt shows up because the word “pineapples” contains the word “apple”, but that’s not the case for line 4 because the “Apple” not equal to “apple” since our script is case sensitive.