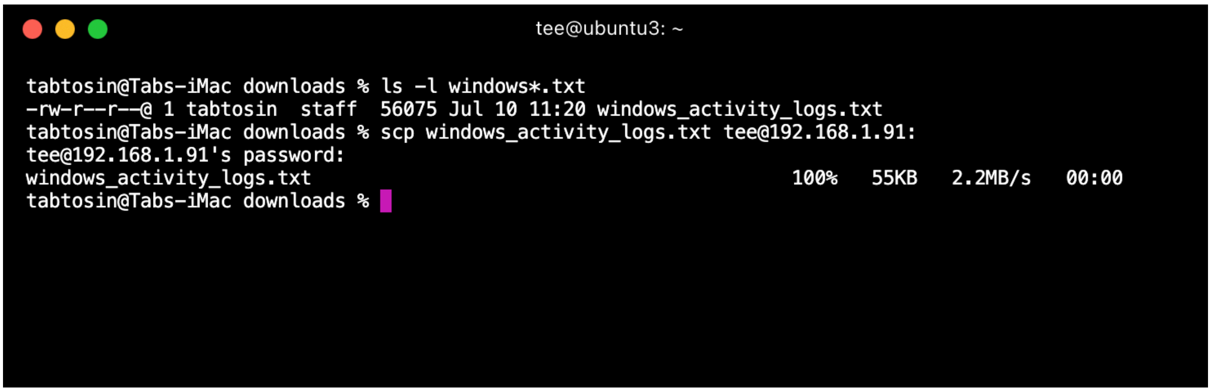
Logging Exercise

Exercise 1

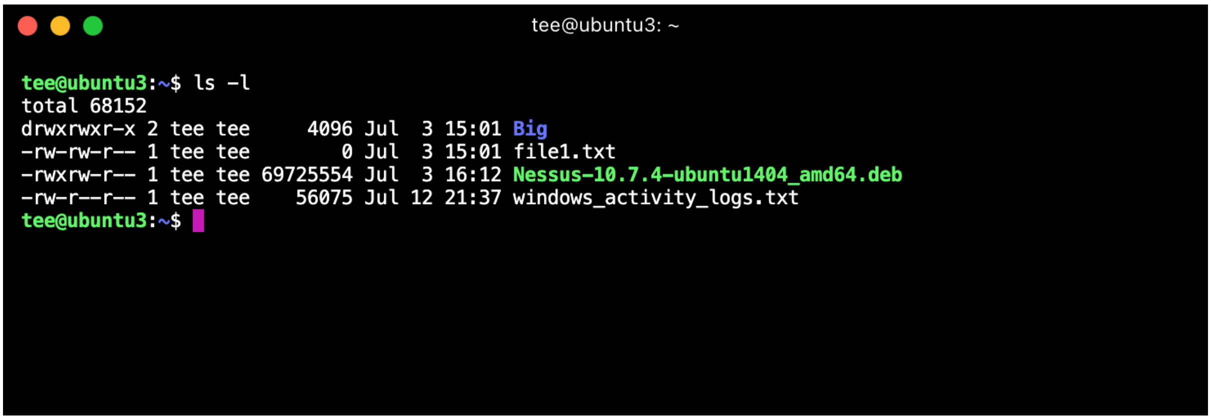
To complete this exercise, a Linux VM and a log file named windows_activity_log.txt

Once the Linux VM is powered on, download the windows_activity_log.txt onto your host system. Save it in your downloads folder. Next, SCP the file to the Linux server using the following command: scp windows_activity_log.txt <username>@<ip of linux server>. This is illustrated below:



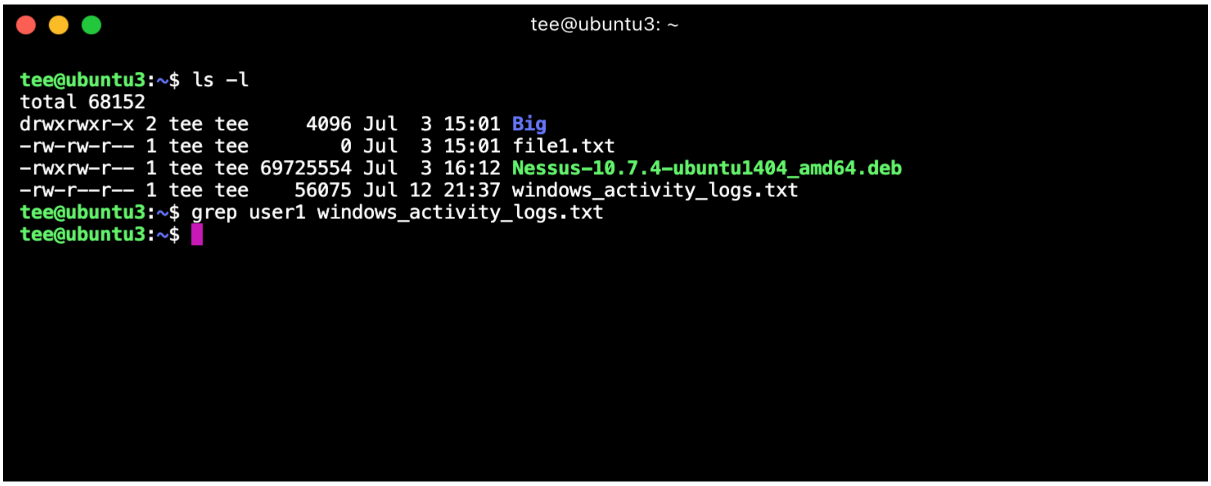
```
tabtosin@Tabs-iMac downloads % ls -l windows*.txt
-rw-r--r--@ 1 tabtosin  staff  56075 Jul 10 11:20 windows_activity_logs.txt
tabtosin@Tabs-iMac downloads % scp windows_activity_logs.txt tee@192.168.1.91:
tee@192.168.1.91's password:
windows_activity_logs.txt                                     100%   55KB   2.2MB/s   00:00
tabtosin@Tabs-iMac downloads %
```

Next, SSH into the Linux Server to validate that the log file was downloaded successfully. This is illustrated below:



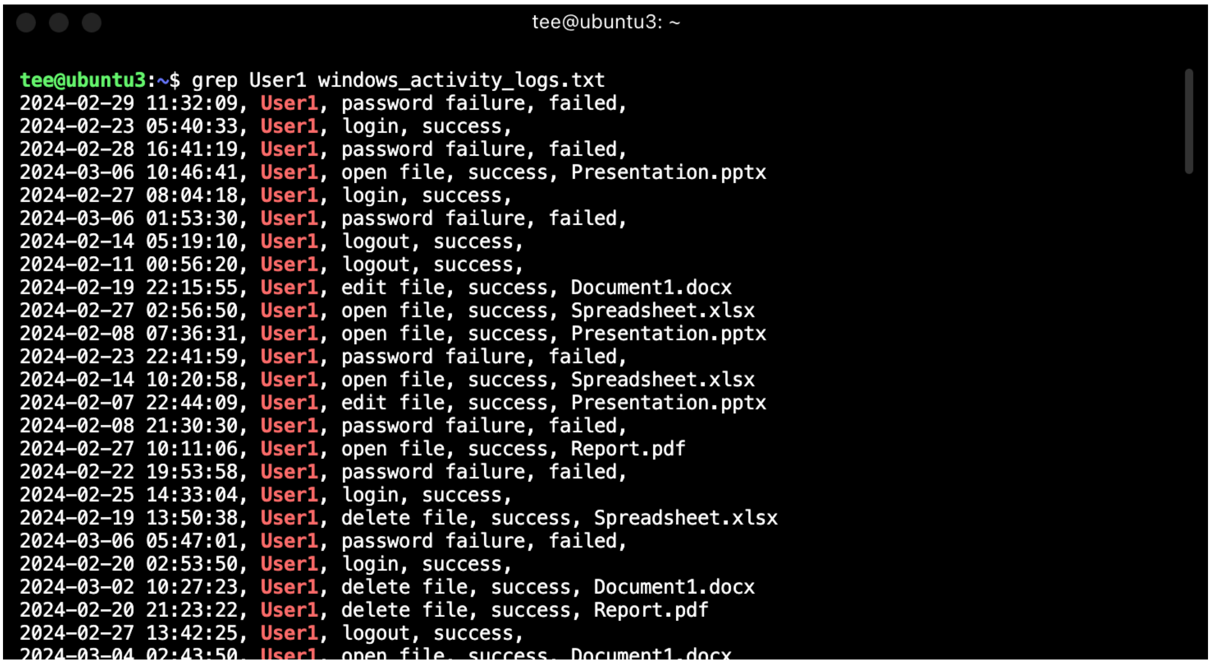
```
tee@ubuntu3:~$ ls -l
total 68152
drwxrwxr-x 2 tee tee     4096 Jul  3 15:01 Big
-rw-rw-r-- 1 tee tee        0 Jul  3 15:01 file1.txt
-rwxrw-r-- 1 tee tee 69725554 Jul  3 16:12 Nessus-10.7.4-ubuntu1404_amd64.deb
-rw-r--r-- 1 tee tee    56075 Jul 12 21:37 windows_activity_logs.txt
tee@ubuntu3:~$
```

Now that it has been confirmed that the log file has been successfully copied to the Linux server, run the following command: grep user1 windows_activity_logs.txt. This is illustrated below:



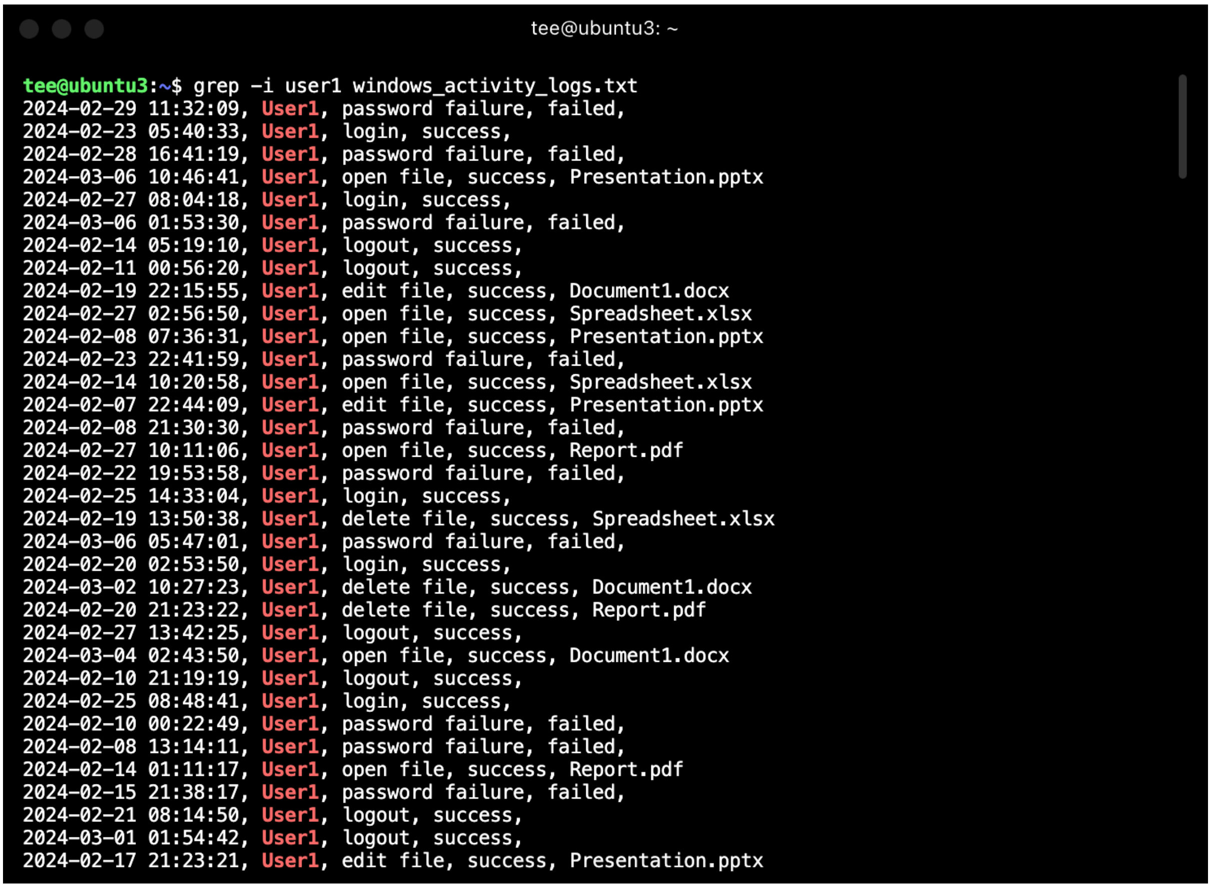
```
tee@ubuntu3:~$ ls -l
total 68152
drwxrwxr-x 2 tee tee     4096 Jul  3 15:01 Big
-rw-rw-r-- 1 tee tee        0 Jul  3 15:01 file1.txt
-rwxrw-r-- 1 tee tee 69725554 Jul  3 16:12 Nessus-10.7.4-ubuntu1404_amd64.deb
-rw-r--r-- 1 tee tee    56075 Jul 12 21:37 windows_activity_logs.txt
tee@ubuntu3:~$ grep user1 windows_activity_logs.txt
tee@ubuntu3:~$
```

grep in Linux is case-sensitive, hence why the last command did not give any output. Changing the command using the appropriate case letters will churn out the expected output as illustrated below:



```
tee@ubuntu3:~$ grep User1 windows_activity_logs.txt
2024-02-29 11:32:09, User1, password failure, failed,
2024-02-23 05:40:33, User1, login, success,
2024-02-28 16:41:19, User1, password failure, failed,
2024-03-06 10:46:41, User1, open file, success, Presentation.pptx
2024-02-27 08:04:18, User1, login, success,
2024-03-06 01:53:30, User1, password failure, failed,
2024-02-14 05:19:10, User1, logout, success,
2024-02-11 00:56:20, User1, logout, success,
2024-02-19 22:15:55, User1, edit file, success, Document1.docx
2024-02-27 02:56:50, User1, open file, success, Spreadsheet.xlsx
2024-02-08 07:36:31, User1, open file, success, Presentation.pptx
2024-02-23 22:41:59, User1, password failure, failed,
2024-02-14 10:20:58, User1, open file, success, Spreadsheet.xlsx
2024-02-07 22:44:09, User1, edit file, success, Presentation.pptx
2024-02-08 21:30:30, User1, password failure, failed,
2024-02-27 10:11:06, User1, open file, success, Report.pdf
2024-02-22 19:53:58, User1, password failure, failed,
2024-02-25 14:33:04, User1, login, success,
2024-02-19 13:50:38, User1, delete file, success, Spreadsheet.xlsx
2024-03-06 05:47:01, User1, password failure, failed,
2024-02-20 02:53:50, User1, login, success,
2024-03-02 10:27:23, User1, delete file, success, Document1.docx
2024-02-20 21:23:22, User1, delete file, success, Report.pdf
2024-02-27 13:42:25, User1, logout, success,
2024-03-04 02:43:50, User1, open file, success, Document1.docx
```

However, to surmount the case-sensitive hurdle, use the '-i' argument in the command line as illustrated below:

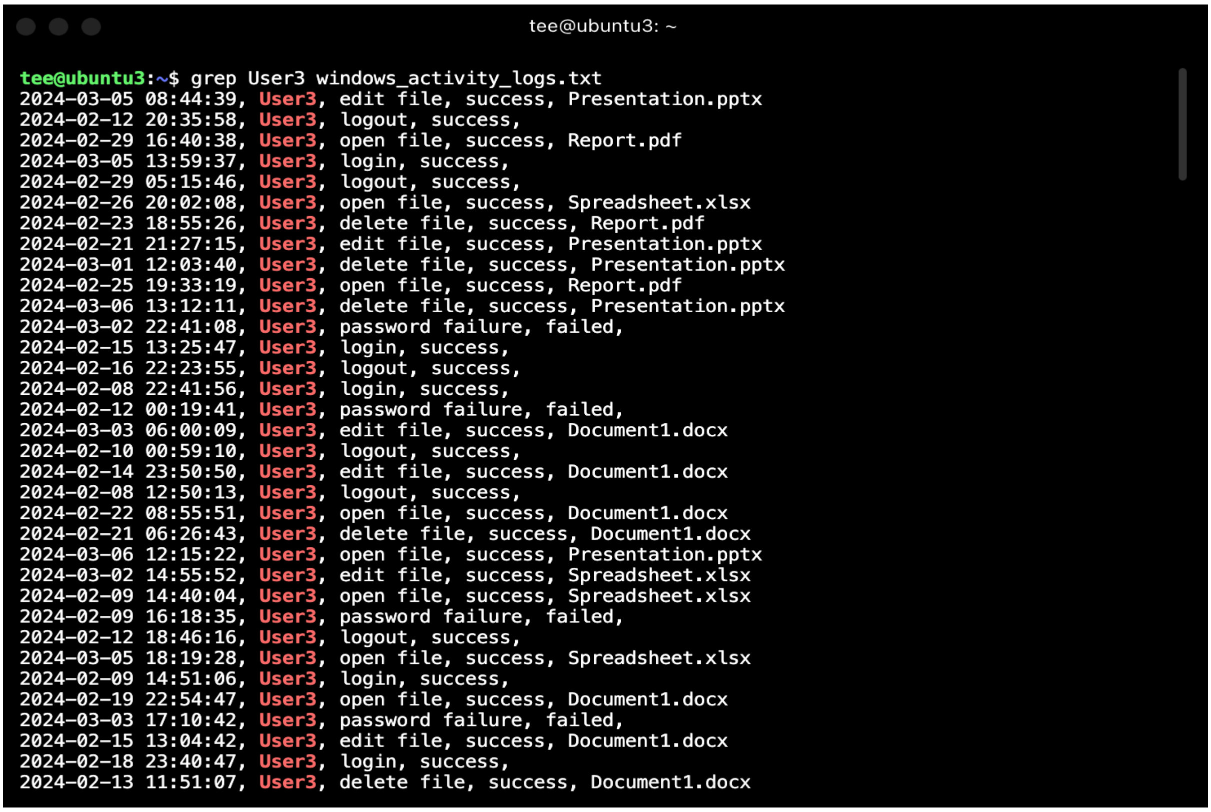


```
tee@ubuntu3:~$ grep -i user1 windows_activity_logs.txt
2024-02-29 11:32:09, User1, password failure, failed,
2024-02-23 05:40:33, User1, login, success,
2024-02-28 16:41:19, User1, password failure, failed,
2024-03-06 10:46:41, User1, open file, success, Presentation.pptx
2024-02-27 08:04:18, User1, login, success,
2024-03-06 01:53:30, User1, password failure, failed,
2024-02-14 05:19:10, User1, logout, success,
2024-02-11 00:56:20, User1, logout, success,
2024-02-19 22:15:55, User1, edit file, success, Document1.docx
2024-02-27 02:56:50, User1, open file, success, Spreadsheet.xlsx
2024-02-08 07:36:31, User1, open file, success, Presentation.pptx
2024-02-23 22:41:59, User1, password failure, failed,
2024-02-14 10:20:58, User1, open file, success, Spreadsheet.xlsx
2024-02-07 22:44:09, User1, edit file, success, Presentation.pptx
2024-02-08 21:30:30, User1, password failure, failed,
2024-02-27 10:11:06, User1, open file, success, Report.pdf
2024-02-22 19:53:58, User1, password failure, failed,
2024-02-25 14:33:04, User1, login, success,
2024-02-19 13:50:38, User1, delete file, success, Spreadsheet.xlsx
2024-03-06 05:47:01, User1, password failure, failed,
2024-02-20 02:53:50, User1, login, success,
2024-03-02 10:27:23, User1, delete file, success, Document1.docx
2024-02-20 21:23:22, User1, delete file, success, Report.pdf
2024-02-27 13:42:25, User1, logout, success,
2024-03-04 02:43:50, User1, open file, success, Document1.docx
2024-02-10 21:19:19, User1, logout, success,
2024-02-25 08:48:41, User1, login, success,
2024-02-10 00:22:49, User1, password failure, failed,
2024-02-08 13:14:11, User1, password failure, failed,
2024-02-14 01:11:17, User1, open file, success, Report.pdf
2024-02-15 21:38:17, User1, password failure, failed,
2024-02-21 08:14:50, User1, logout, success,
2024-03-01 01:54:42, User1, logout, success,
2024-02-17 21:23:21, User1, edit file, success, Presentation.pptx
```

As shown above, including the '-i' argument runs the desired search regardless of case.

Exercise 2

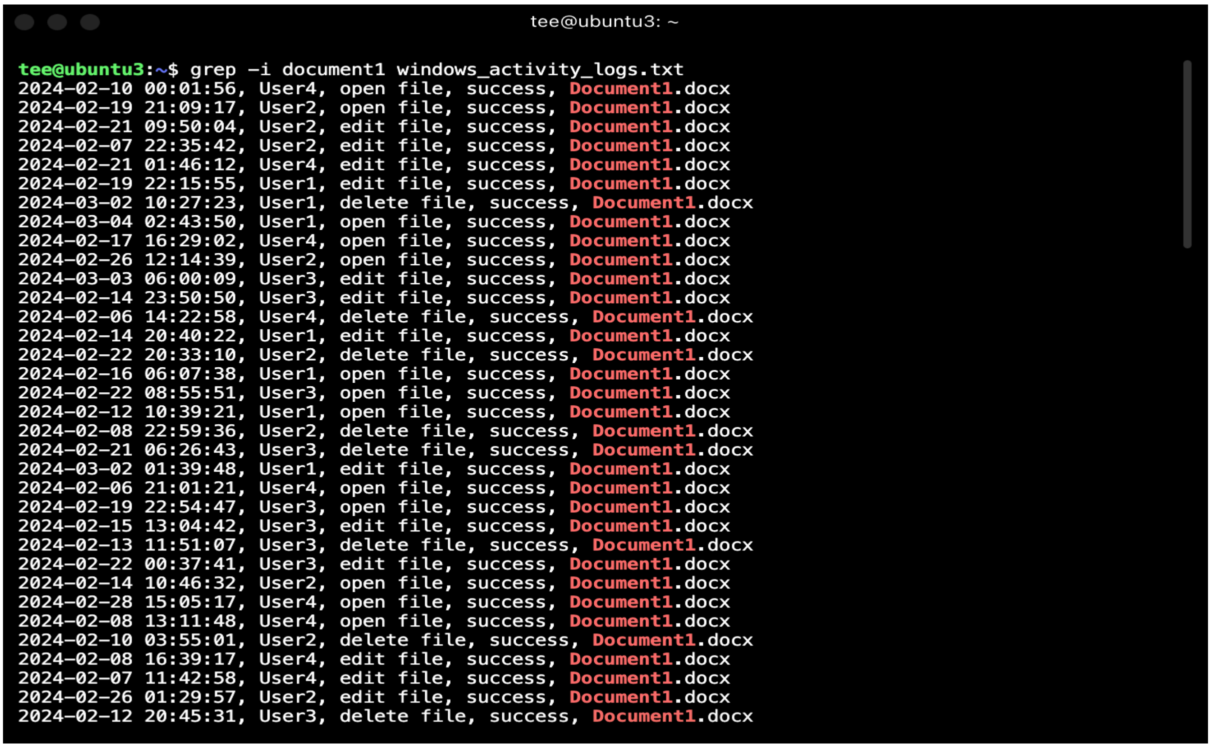
Run the following command to scan through the log file for User3: grep User3 windows_activity_logs.txt. This is illustrated below:



```
tee@ubuntu3:~$ grep User3 windows_activity_logs.txt
2024-03-05 08:44:39, User3, edit file, success, Presentation.pptx
2024-02-12 20:35:58, User3, logout, success,
2024-02-29 16:40:38, User3, open file, success, Report.pdf
2024-03-05 13:59:37, User3, login, success,
2024-02-29 05:15:46, User3, logout, success,
2024-02-26 20:02:08, User3, open file, success, Spreadsheet.xlsx
2024-02-23 18:55:26, User3, delete file, success, Report.pdf
2024-02-21 21:27:15, User3, edit file, success, Presentation.pptx
2024-03-01 12:03:40, User3, delete file, success, Presentation.pptx
2024-02-25 19:33:19, User3, open file, success, Report.pdf
2024-03-06 13:12:11, User3, delete file, success, Presentation.pptx
2024-03-02 22:41:08, User3, password failure, failed,
2024-02-15 13:25:47, User3, login, success,
2024-02-16 22:23:55, User3, logout, success,
2024-02-08 22:41:56, User3, login, success,
2024-02-12 00:19:41, User3, password failure, failed,
2024-03-03 06:00:09, User3, edit file, success, Document1.docx
2024-02-10 00:59:10, User3, logout, success,
2024-02-14 23:50:50, User3, edit file, success, Document1.docx
2024-02-08 12:50:13, User3, logout, success,
2024-02-22 08:55:51, User3, open file, success, Document1.docx
2024-02-21 06:26:43, User3, delete file, success, Document1.docx
2024-03-06 12:15:22, User3, open file, success, Presentation.pptx
2024-03-02 14:55:52, User3, edit file, success, Spreadsheet.xlsx
2024-02-09 14:40:04, User3, open file, success, Spreadsheet.xlsx
2024-02-09 16:18:35, User3, password failure, failed,
2024-02-12 18:46:16, User3, logout, success,
2024-03-05 18:19:28, User3, open file, success, Spreadsheet.xlsx
2024-02-09 14:51:06, User3, login, success,
2024-02-19 22:54:47, User3, open file, success, Document1.docx
2024-03-03 17:10:42, User3, password failure, failed,
2024-02-15 13:04:42, User3, edit file, success, Document1.docx
2024-02-18 23:40:47, User3, login, success,
2024-02-13 11:51:07, User3, delete file, success, Document1.docx
```

Exercise 3

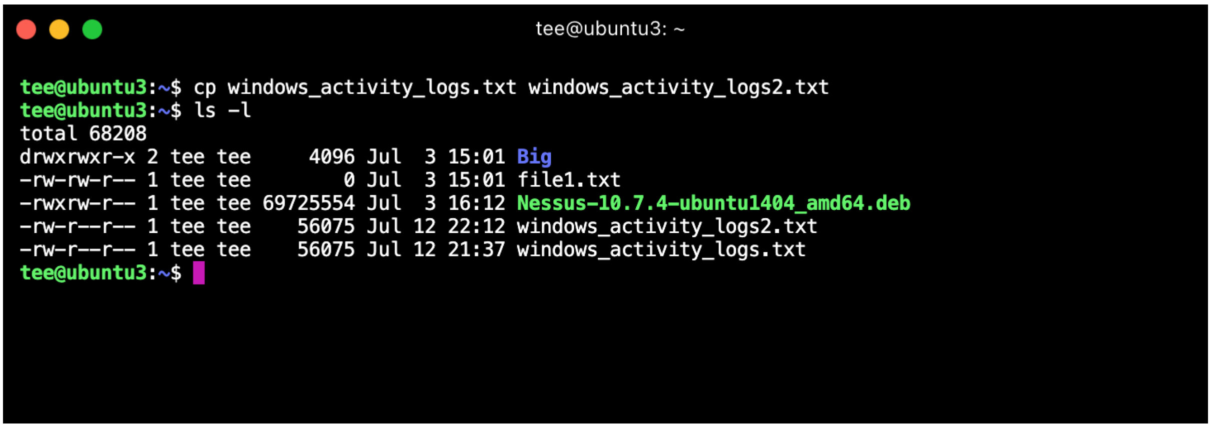
Search for 'document1' using the grep command template. This is illustrated below:



```
tee@ubuntu3:~$ grep -i document1 windows_activity_logs.txt
2024-02-10 00:01:56, User4, open file, success, Document1.docx
2024-02-19 21:09:17, User2, open file, success, Document1.docx
2024-02-21 09:50:04, User2, edit file, success, Document1.docx
2024-02-07 22:35:42, User2, edit file, success, Document1.docx
2024-02-21 01:46:12, User4, edit file, success, Document1.docx
2024-02-19 22:15:55, User1, edit file, success, Document1.docx
2024-03-02 10:27:23, User1, delete file, success, Document1.docx
2024-03-04 02:43:50, User1, open file, success, Document1.docx
2024-02-17 16:29:02, User4, open file, success, Document1.docx
2024-02-26 12:14:39, User2, open file, success, Document1.docx
2024-03-03 06:00:09, User3, edit file, success, Document1.docx
2024-02-14 23:50:50, User3, edit file, success, Document1.docx
2024-02-06 14:22:58, User4, delete file, success, Document1.docx
2024-02-14 20:40:22, User1, edit file, success, Document1.docx
2024-02-22 20:33:10, User2, delete file, success, Document1.docx
2024-02-16 06:07:38, User1, open file, success, Document1.docx
2024-02-22 08:55:51, User3, open file, success, Document1.docx
2024-02-12 10:39:21, User1, open file, success, Document1.docx
2024-02-08 22:59:36, User2, delete file, success, Document1.docx
2024-02-21 06:26:43, User3, delete file, success, Document1.docx
2024-03-02 01:39:48, User1, edit file, success, Document1.docx
2024-02-06 21:01:21, User4, open file, success, Document1.docx
2024-02-19 22:54:47, User3, open file, success, Document1.docx
2024-02-15 13:04:42, User3, edit file, success, Document1.docx
2024-02-13 11:51:07, User3, delete file, success, Document1.docx
2024-02-22 00:37:41, User3, edit file, success, Document1.docx
2024-02-14 10:46:32, User2, open file, success, Document1.docx
2024-02-28 15:05:17, User4, open file, success, Document1.docx
2024-02-08 13:11:48, User4, open file, success, Document1.docx
2024-02-10 03:55:01, User2, delete file, success, Document1.docx
2024-02-08 16:39:17, User4, edit file, success, Document1.docx
2024-02-07 11:42:58, User4, edit file, success, Document1.docx
2024-02-26 01:29:57, User2, edit file, success, Document1.docx
2024-02-12 20:45:31, User3, delete file, success, Document1.docx
```

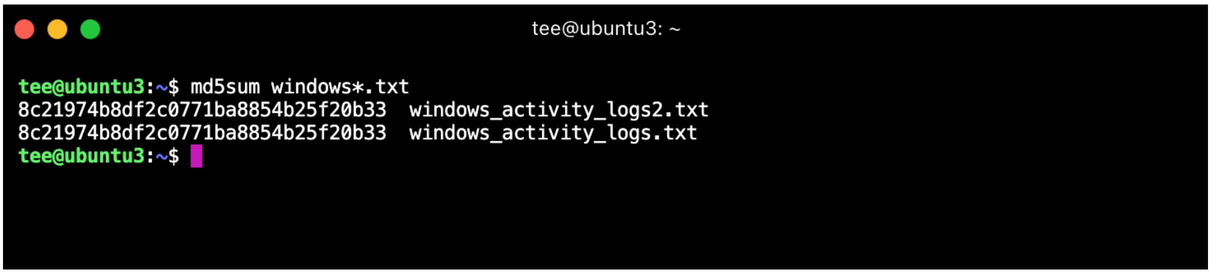
Exercise 4

The next step is to make a copy of the windows_activity_logs.txt file, and name it windows_activity_logs2.txt. This is illustrated below:



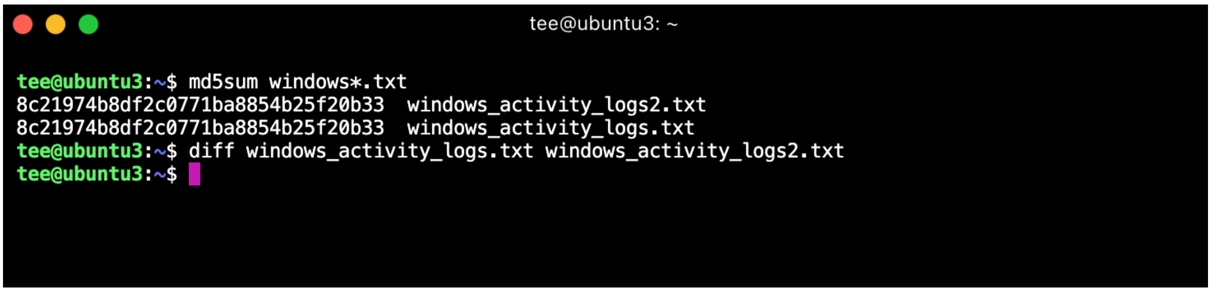
```
tee@ubuntu3:~$ cp windows_activity_logs.txt windows_activity_logs2.txt
tee@ubuntu3:~$ ls -l
total 68208
drwxrwxr-x 2 tee tee     4096 Jul  3 15:01 Big
-rw-rw-r-- 1 tee tee        0 Jul  3 15:01 file1.txt
-rwxrw-r-- 1 tee tee 69725554 Jul  3 16:12 Nessus-10.7.4-ubuntu1404_amd64.deb
-rw-r--r-- 1 tee tee    56075 Jul 12 22:12 windows_activity_logs2.txt
-rw-r--r-- 1 tee tee    56075 Jul 12 21:37 windows_activity_logs.txt
tee@ubuntu3:~$ 
```

The next step is to compare the hash of the original windows log file and the duplicate.



```
tee@ubuntu3:~$ md5sum windows*.txt
8c21974b8df2c0771ba8854b25f20b33  windows_activity_logs2.txt
8c21974b8df2c0771ba8854b25f20b33  windows_activity_logs.txt
tee@ubuntu3:~$ 
```

As shown above, the hash of both files are identical. We can also compare both files using the 'diff' command as illustrated below:



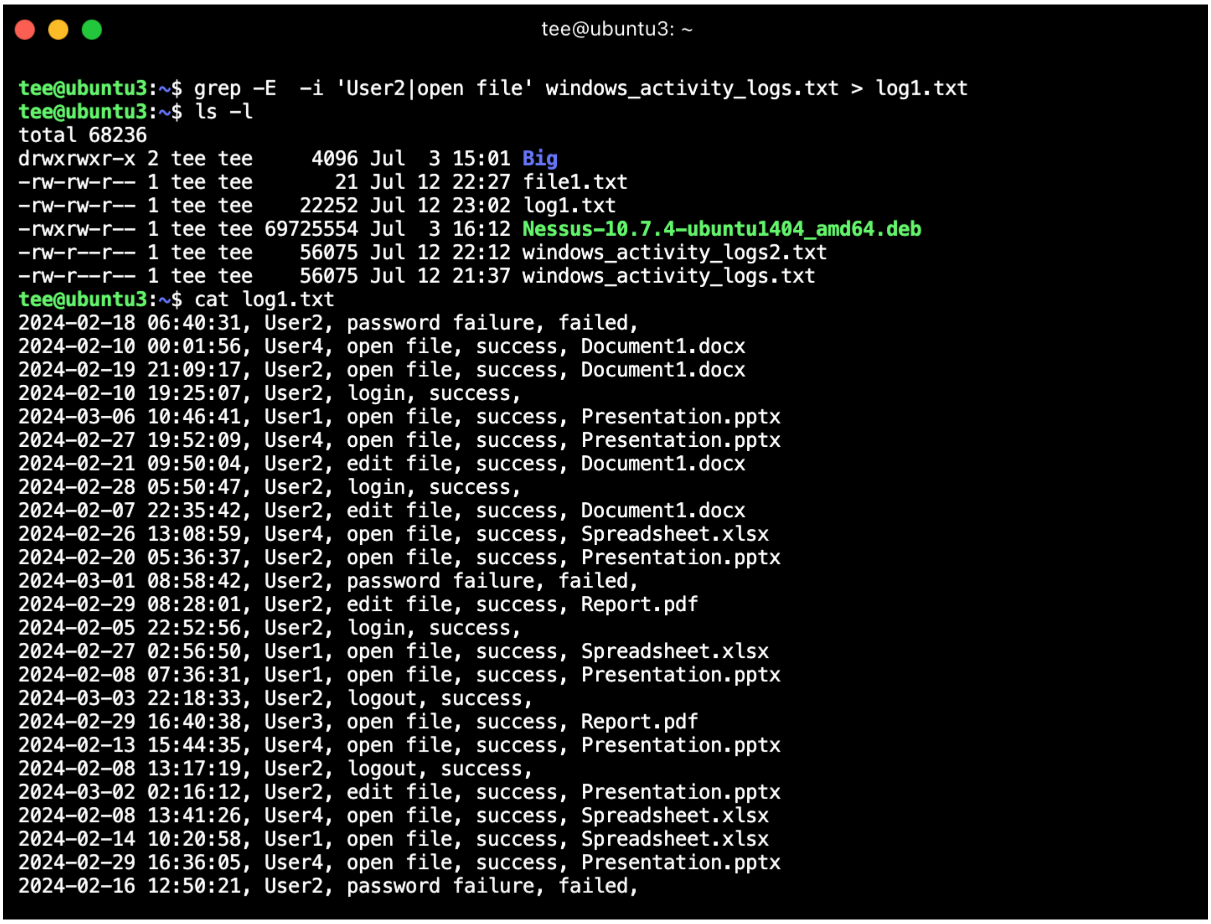
```
tee@ubuntu3:~$ md5sum windows*.txt
8c21974b8df2c0771ba8854b25f20b33  windows_activity_logs2.txt
8c21974b8df2c0771ba8854b25f20b33  windows_activity_logs.txt
tee@ubuntu3:~$ diff windows_activity_logs.txt windows_activity_logs2.txt
tee@ubuntu3:~$ 
```

Not having any output returned confirms that the two files are identical.

Exercise 6

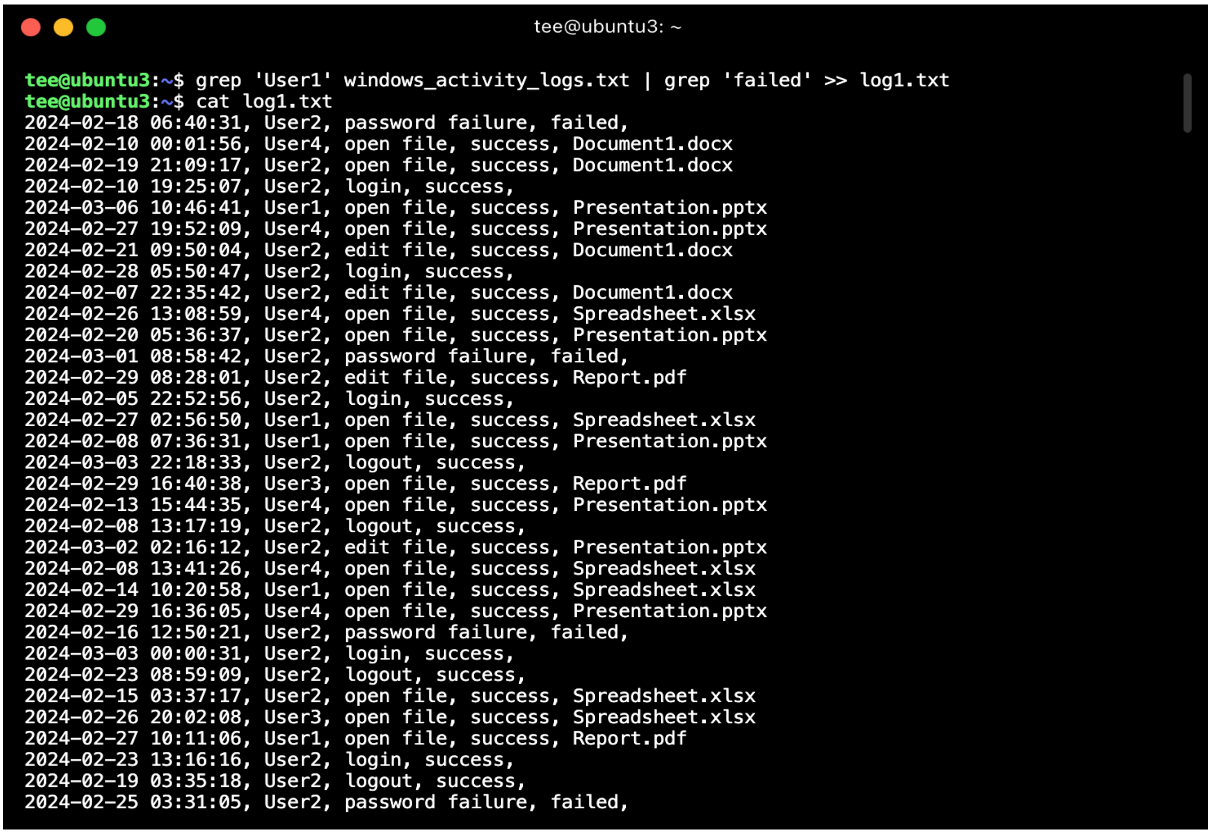
Using grep with logical operators. Grep can be used to search for two different variables in a file. There are two options. The first option is used to find lines that carry both variables being searched for (logical AND), the second option is to look for either of the two variables (logical OR).

To use the logical OR, use the -E argument with the grep command, and separate the variables with a pipe '|'. The command should read as follows: grep -E 'pattern1|pattern2' filename. This is illustrated below searching for any log containing the variables 'User2; or 'open file'. The output will be sent to a file named log1.txt:



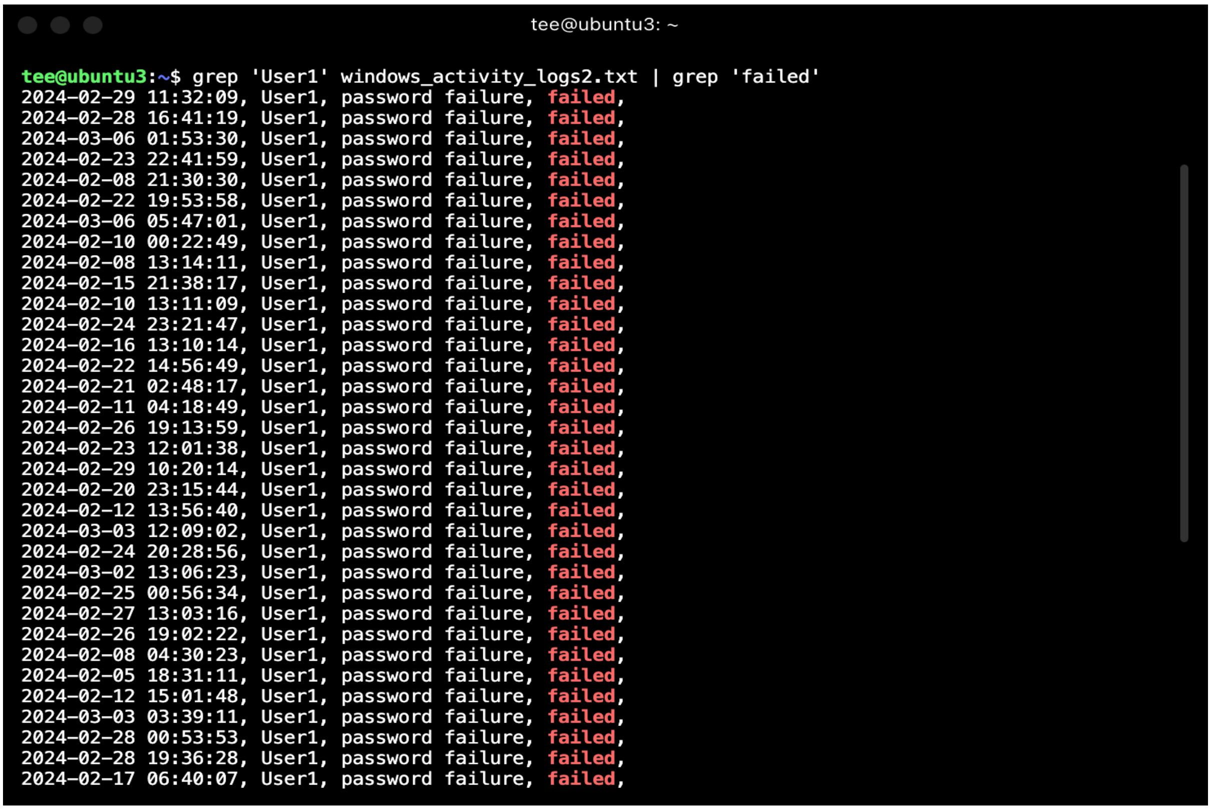
```
tee@ubuntu3:~$ grep -E  -i 'User2|open file' windows_activity_logs.txt > log1.txt
tee@ubuntu3:~$ ls -l
total 68236
drwxrwxr-x 2 tee tee     4096 Jul  3 15:01 Big
-rw-rw-r-- 1 tee tee       21 Jul 12 22:27 file1.txt
-rw-rw-r-- 1 tee tee    22252 Jul 12 23:02 log1.txt
-rwxrw-r-- 1 tee tee 69725554 Jul  3 16:12 Nessus-10.7.4-ubuntu1404_amd64.deb
-rw-r--r-- 1 tee tee    56075 Jul 12 22:12 windows_activity_logs2.txt
-rw-r--r-- 1 tee tee    56075 Jul 12 21:37 windows_activity_logs.txt
tee@ubuntu3:~$ cat log1.txt
2024-02-18 06:40:31, User2, password failure, failed,
2024-02-10 00:01:56, User4, open file, success, Document1.docx
2024-02-19 21:09:17, User2, open file, success, Document1.docx
2024-02-10 19:25:07, User2, login, success,
2024-03-06 10:46:41, User1, open file, success, Presentation.pptx
2024-02-27 19:52:09, User4, open file, success, Presentation.pptx
2024-02-21 09:50:04, User2, edit file, success, Document1.docx
2024-02-28 05:50:47, User2, login, success,
2024-02-07 22:35:42, User2, edit file, success, Document1.docx
2024-02-26 13:08:59, User4, open file, success, Spreadsheet.xlsx
2024-02-20 05:36:37, User2, open file, success, Presentation.pptx
2024-03-01 08:58:42, User2, password failure, failed,
2024-02-29 08:28:01, User2, edit file, success, Report.pdf
2024-02-05 22:52:56, User2, login, success,
2024-02-27 02:56:50, User1, open file, success, Spreadsheet.xlsx
2024-02-08 07:36:31, User1, open file, success, Presentation.pptx
2024-03-03 22:18:33, User2, logout, success,
2024-02-29 16:40:38, User3, open file, success, Report.pdf
2024-02-13 15:44:35, User4, open file, success, Presentation.pptx
2024-02-08 13:17:19, User2, logout, success,
2024-03-02 02:16:12, User2, edit file, success, Presentation.pptx
2024-02-08 13:41:26, User4, open file, success, Spreadsheet.xlsx
2024-02-14 10:20:58, User1, open file, success, Spreadsheet.xlsx
2024-02-29 16:36:05, User4, open file, success, Presentation.pptx
2024-02-16 12:50:21, User2, password failure, failed,
```

To use the Logical AND, simply chain both grep commands using the pipe symbol. The command should look like this: grep ‘pattern1’ filename | grep ‘pattern2’. This is also illustrated below searching for the variables ‘User1’ and ‘failed’. The output will be appended to log1.txt:



```
tee@ubuntu3:~$ grep 'User1' windows_activity_logs.txt | grep 'failed' >> log1.txt
tee@ubuntu3:~$ cat log1.txt
2024-02-18 06:40:31, User2, password failure, failed,
2024-02-10 00:01:56, User4, open file, success, Document1.docx
2024-02-19 21:09:17, User2, open file, success, Document1.docx
2024-02-10 19:25:07, User2, login, success,
2024-03-06 10:46:41, User1, open file, success, Presentation.pptx
2024-02-27 19:52:09, User4, open file, success, Presentation.pptx
2024-02-21 09:50:04, User2, edit file, success, Document1.docx
2024-02-28 05:50:47, User2, login, success,
2024-02-07 22:35:42, User2, edit file, success, Document1.docx
2024-02-26 13:08:59, User4, open file, success, Spreadsheet.xlsx
2024-02-20 05:36:37, User2, open file, success, Presentation.pptx
2024-03-01 08:58:42, User2, password failure, failed,
2024-02-29 08:28:01, User2, edit file, success, Report.pdf
2024-02-05 22:52:56, User2, login, success,
2024-02-27 02:56:50, User1, open file, success, Spreadsheet.xlsx
2024-02-08 07:36:31, User1, open file, success, Presentation.pptx
2024-03-03 22:18:33, User2, logout, success,
2024-02-29 16:40:38, User3, open file, success, Report.pdf
2024-02-13 15:44:35, User4, open file, success, Presentation.pptx
2024-02-08 13:17:19, User2, logout, success,
2024-03-02 02:16:12, User2, edit file, success, Presentation.pptx
2024-02-08 13:41:26, User4, open file, success, Spreadsheet.xlsx
2024-02-14 10:20:58, User1, open file, success, Spreadsheet.xlsx
2024-02-29 16:36:05, User4, open file, success, Presentation.pptx
2024-02-16 12:50:21, User2, password failure, failed,
2024-03-03 00:00:31, User2, login, success,
2024-02-23 08:59:09, User2, logout, success,
2024-02-15 03:37:17, User2, open file, success, Spreadsheet.xlsx
2024-02-26 20:02:08, User3, open file, success, Spreadsheet.xlsx
2024-02-27 10:11:06, User1, open file, success, Report.pdf
2024-02-23 13:16:16, User2, login, success,
2024-02-19 03:35:18, User2, logout, success,
2024-02-25 03:31:05, User2, password failure, failed,
```

This same command can also be used on other files as shown below:



```
tee@ubuntu3:~$ grep 'User1' windows_activity_logs2.txt | grep 'failed'
2024-02-29 11:32:09, User1, password failure, failed,
2024-02-28 16:41:19, User1, password failure, failed,
2024-03-06 01:53:30, User1, password failure, failed,
2024-02-23 22:41:59, User1, password failure, failed,
2024-02-08 21:30:30, User1, password failure, failed,
2024-02-22 19:53:58, User1, password failure, failed,
2024-03-06 05:47:01, User1, password failure, failed,
2024-02-10 00:22:49, User1, password failure, failed,
2024-02-08 13:14:11, User1, password failure, failed,
2024-02-15 21:38:17, User1, password failure, failed,
2024-02-10 13:11:09, User1, password failure, failed,
2024-02-24 23:21:47, User1, password failure, failed,
2024-02-16 13:10:14, User1, password failure, failed,
2024-02-22 14:56:49, User1, password failure, failed,
2024-02-21 02:48:17, User1, password failure, failed,
2024-02-11 04:18:49, User1, password failure, failed,
2024-02-26 19:13:59, User1, password failure, failed,
2024-02-23 12:01:38, User1, password failure, failed,
2024-02-29 10:20:14, User1, password failure, failed,
2024-02-20 23:15:44, User1, password failure, failed,
2024-02-12 13:56:40, User1, password failure, failed,
2024-03-03 12:09:02, User1, password failure, failed,
2024-02-24 20:28:56, User1, password failure, failed,
2024-03-02 13:06:23, User1, password failure, failed,
2024-02-25 00:56:34, User1, password failure, failed,
2024-02-27 13:03:16, User1, password failure, failed,
2024-02-26 19:02:22, User1, password failure, failed,
2024-02-08 04:30:23, User1, password failure, failed,
2024-02-05 18:31:11, User1, password failure, failed,
2024-02-12 15:01:48, User1, password failure, failed,
2024-03-03 03:39:11, User1, password failure, failed,
2024-02-28 00:53:53, User1, password failure, failed,
2024-02-28 19:36:28, User1, password failure, failed,
2024-02-17 06:40:07, User1, password failure, failed,
```

This shows the various ways through which grep can be used to filter through log files.