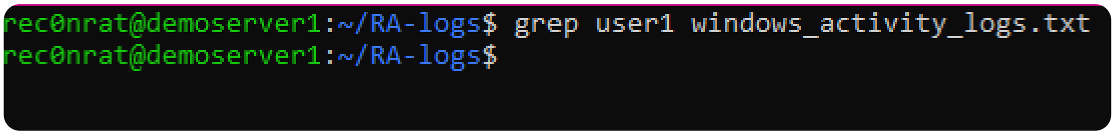
Using grep to Search Logs

Written by Tyler Weiss, 06 MAR 2024

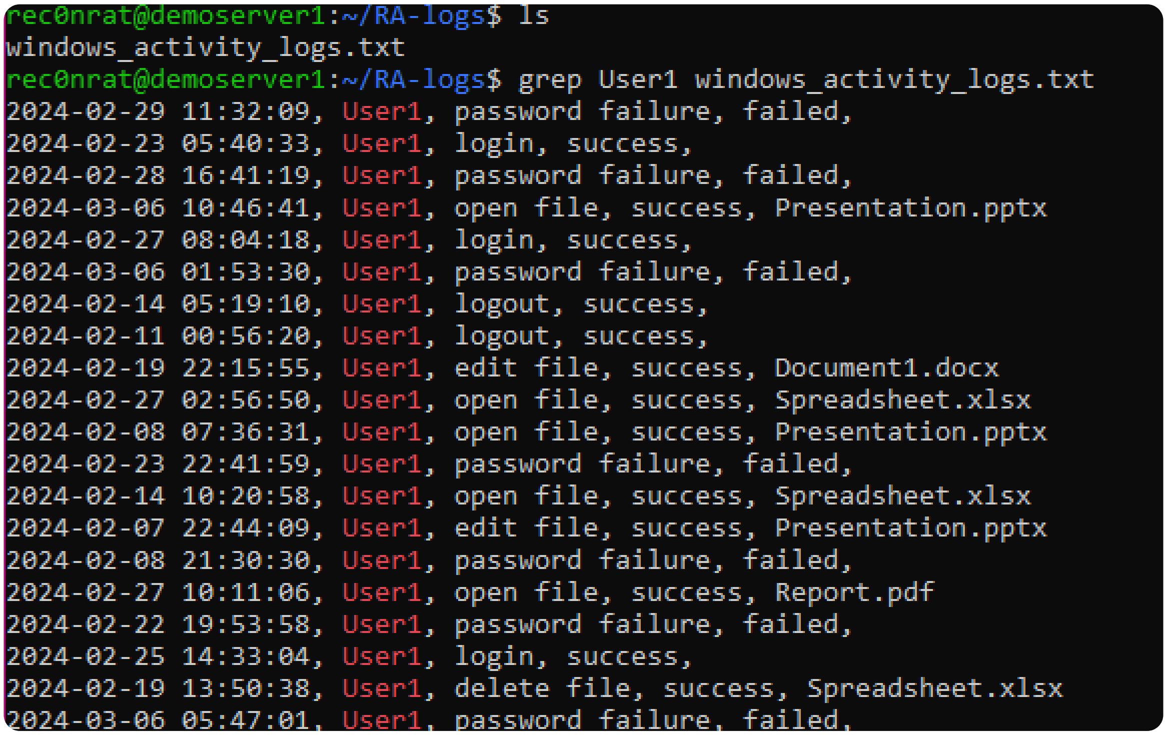
---

grep for 'user1' in log 'windows_activity_logs.txt'. No output is printed to the console because Linux and the grep command are both case sensitive and no line in the file contains 'user1'.



```
rec0nrat@demoserver1:~/RA-logs$ grep user1 windows_activity_logs.txt
rec0nrat@demoserver1:~/RA-logs$
```

grep for 'User1' in log 'windows_activity_logs.txt'.

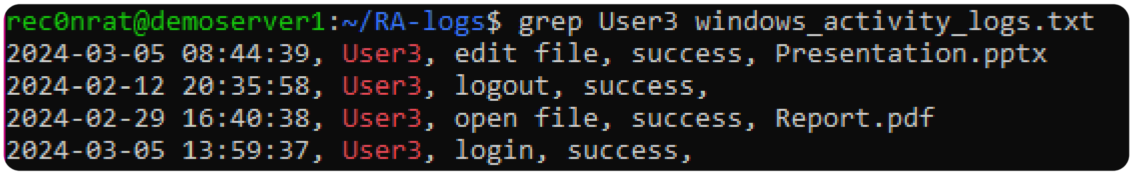


```
rec0nrat@demoserver1:~/RA-logs$ ls
windows_activity_logs.txt
rec0nrat@demoserver1:~/RA-logs$ grep User1 windows_activity_logs.txt
2024-02-29 11:32:09, User1, password failure, failed,
2024-02-23 05:40:33, User1, login, success,
2024-02-28 16:41:19, User1, password failure, failed,
2024-03-06 10:46:41, 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,
```

Output from grep is printed to the console showing the argument 'User1' highlighted in red.

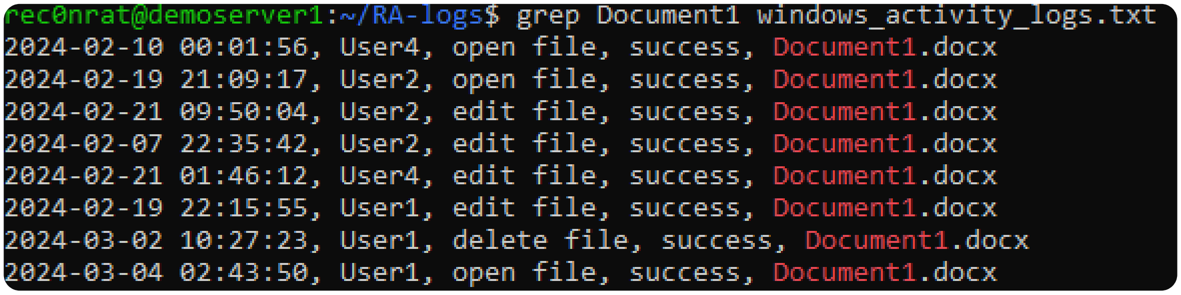
- The argument 'user1' and 'User1' are different due to capitalization.
- grep outputs all lines in the file containing the argument.

grep for 'User3' in log 'windows_activity_logs.txt'. The output should output all lines containing the word string 'User3' from the searched log.



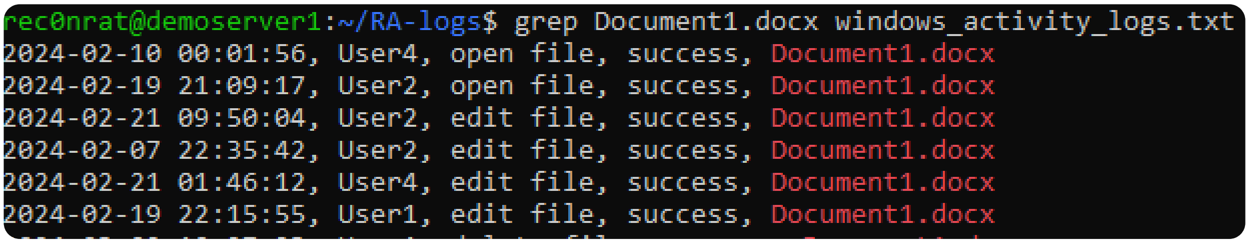
```
rec0nrat@demoserver1:~/RA-logs$ grep User3 windows_activity_logs.txt
2024-03-05 08:44:39, User3, edit file, success, Presentation.pptx
2024-02-12 20:35:58, User3, logout, success,
2024-02-29 16:40:38, User3, open file, success, Report.pdf
2024-03-05 13:59:37, User3, login, success,
```

grep for 'Document1' in log 'windows_activity_logs.txt'.



```
rec0nrat@demoserver1:~/RA-logs$ grep 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
```

grep for 'Document1.docx' in log 'windows_activity_logs.txt'.



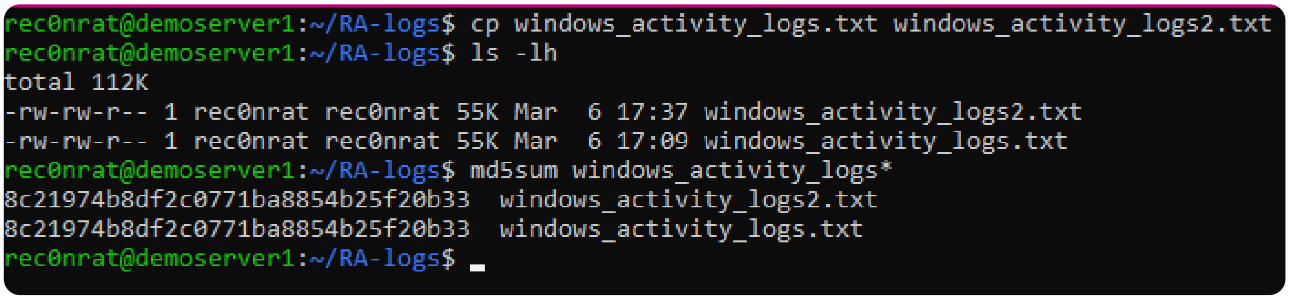
```
rec0nrat@demoserver1:~/RA-logs$ grep Document1.docx windows_activity_logs.txt
2024-02-10 00:01:56, User4, open file, success, Document1.docx
2024-02-19 21:09:17, User2, open file, success, Document1.docx
2024-02-21 09:50:04, User2, edit file, success, Document1.docx
2024-02-07 22:35:42, User2, edit 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
```

Notice that searching for 'Document1' and 'Document1.docx' yields different results.

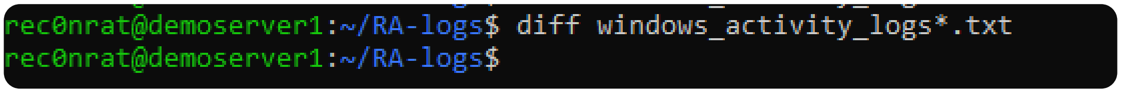
- Observe that the highlighting is different in both grep searches.
- The argument being search for is highlighted in red.

Copy 'windows_activity_logs.txt' to 'windows_activity_logs.txt2'.

- Validate that both text documents exist and are the same using 'md5sum' and 'diff'.



```
rec0nrat@demoserver1:~/RA-logs$ cp windows_activity_logs.txt windows_activity_logs2.txt
rec0nrat@demoserver1:~/RA-logs$ ls -lh
total 112K
-rw-rw-r-- 1 rec0nrat rec0nrat 55K Mar  6 17:37 windows_activity_logs2.txt
-rw-rw-r-- 1 rec0nrat rec0nrat 55K Mar  6 17:09 windows_activity_logs.txt
rec0nrat@demoserver1:~/RA-logs$ md5sum windows_activity_logs*
8c21974b8df2c0771ba8854b25f20b33  windows_activity_logs2.txt
8c21974b8df2c0771ba8854b25f20b33  windows_activity_logs.txt
rec0nrat@demoserver1:~/RA-logs$ 
```



```
rec0nrat@demoserver1:~/RA-logs$ diff windows_activity_logs*.txt
rec0nrat@demoserver1:~/RA-logs$
```

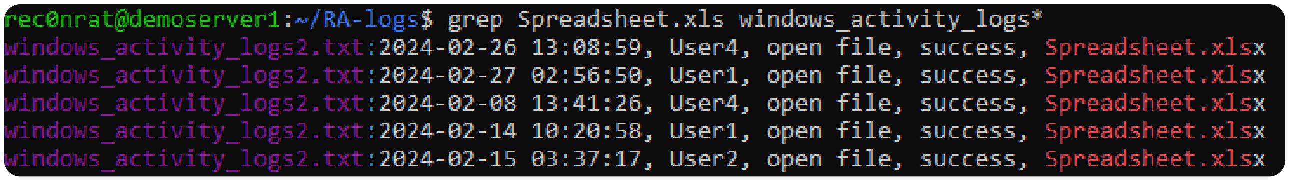
  Notice that the hash values are a match showing that there is no difference between the files.

- Note that using the wild card character allows for both files to be processed by 'md5sum' at the same time creating and easier comparison of the hash values.

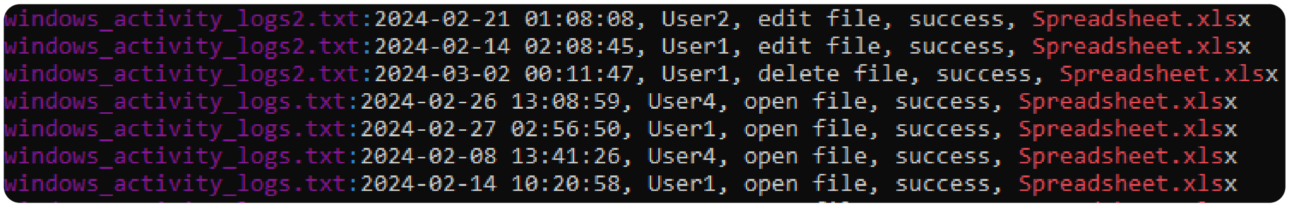
- 'diff windows_activity_logs*.txt ' has no output because the files being compared have no differences between them. The files are exactly the same except for the file name.

grep for 'Spreadsheet.xls' in both 'windows_activity_logs.txt' and 'windows_activity_logs2.txt'.

- Use the wild card character in the grep command to reference both files.



```
rec0nrat@demoserver1:~/RA-logs$ grep Spreadsheet.xls windows_activity_logs*
windows_activity_logs2.txt:2024-02-26 13:08:59, User4, open file, success, Spreadsheet.xlsx
windows_activity_logs2.txt:2024-02-27 02:56:50, User1, open file, success, Spreadsheet.xlsx
windows_activity_logs2.txt:2024-02-08 13:41:26, User4, open file, success, Spreadsheet.xlsx
windows_activity_logs2.txt:2024-02-14 10:20:58, User1, open file, success, Spreadsheet.xlsx
windows_activity_logs2.txt:2024-02-15 03:37:17, User2, open file, success, Spreadsheet.xlsx
```

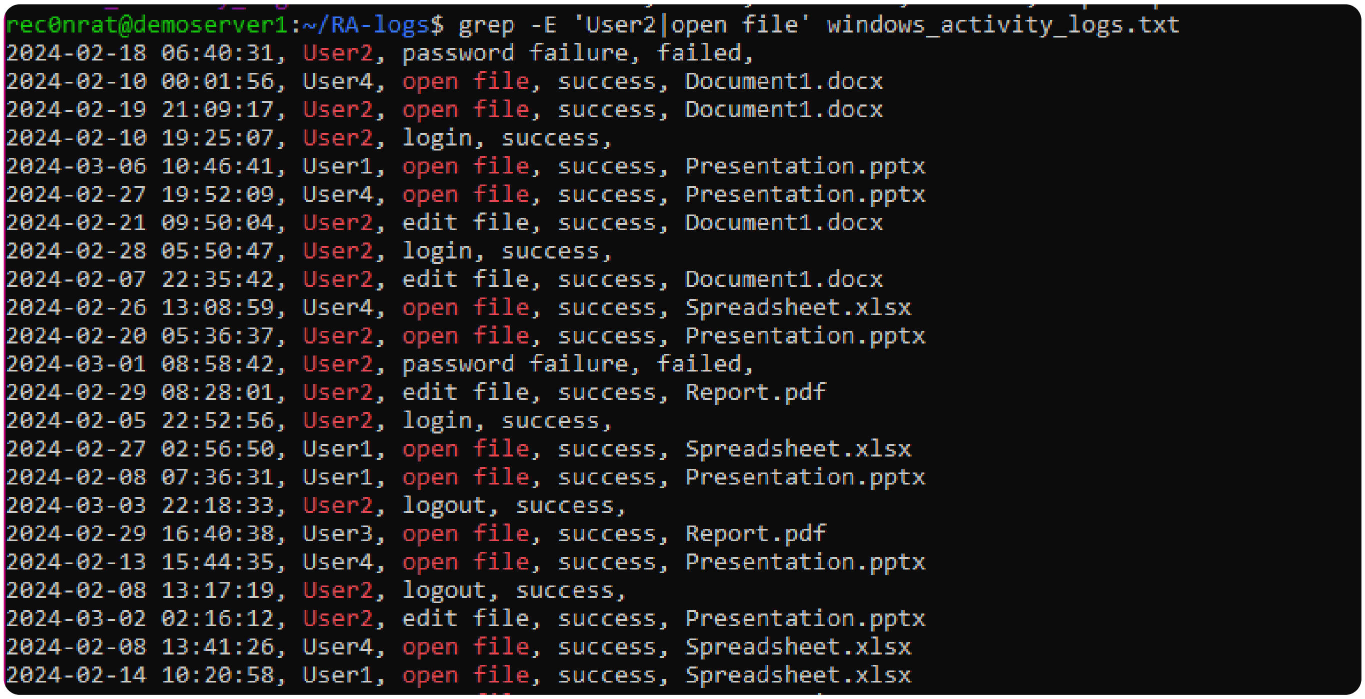


```
windows_activity_logs2.txt:2024-02-21 01:08:08, User2, edit file, success, Spreadsheet.xlsx
windows_activity_logs2.txt:2024-02-14 02:08:45, User1, edit file, success, Spreadsheet.xlsx
windows_activity_logs2.txt:2024-03-02 00:11:47, User1, delete file, success, Spreadsheet.xlsx
windows_activity_logs.txt:2024-02-26 13:08:59, User4, open file, success, Spreadsheet.xlsx
windows_activity_logs.txt:2024-02-27 02:56:50, User1, open file, success, Spreadsheet.xlsx
windows_activity_logs.txt:2024-02-08 13:41:26, User4, open file, success, Spreadsheet.xlsx
windows_activity_logs.txt:2024-02-14 10:20:58, User1, open file, success, Spreadsheet.xlsx
```

Observe that the console output prints all lines containing the argument 'Spreadsheet.xls' in both files.

- The name file being searched is highlighted in purple.
- The line from the file is printed to the right of the file name which is separated by a colon ':'.

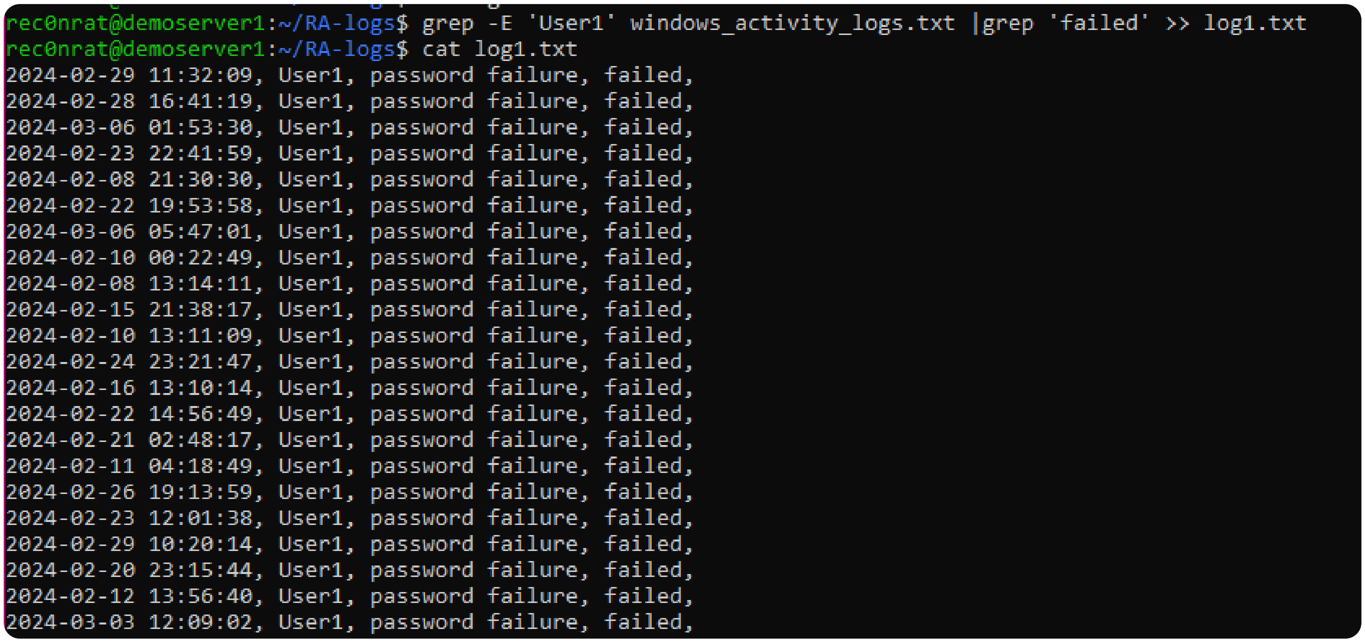
grep for the string 'User2' or 'open file' in 'windows_activity_logs.txt'. Use the '|' character, that represents the logical OR to search the logs for either string.



```
rec0nrat@demoserver1:~/RA-logs$ grep -E 'User2|open file' windows_activity_logs.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
```

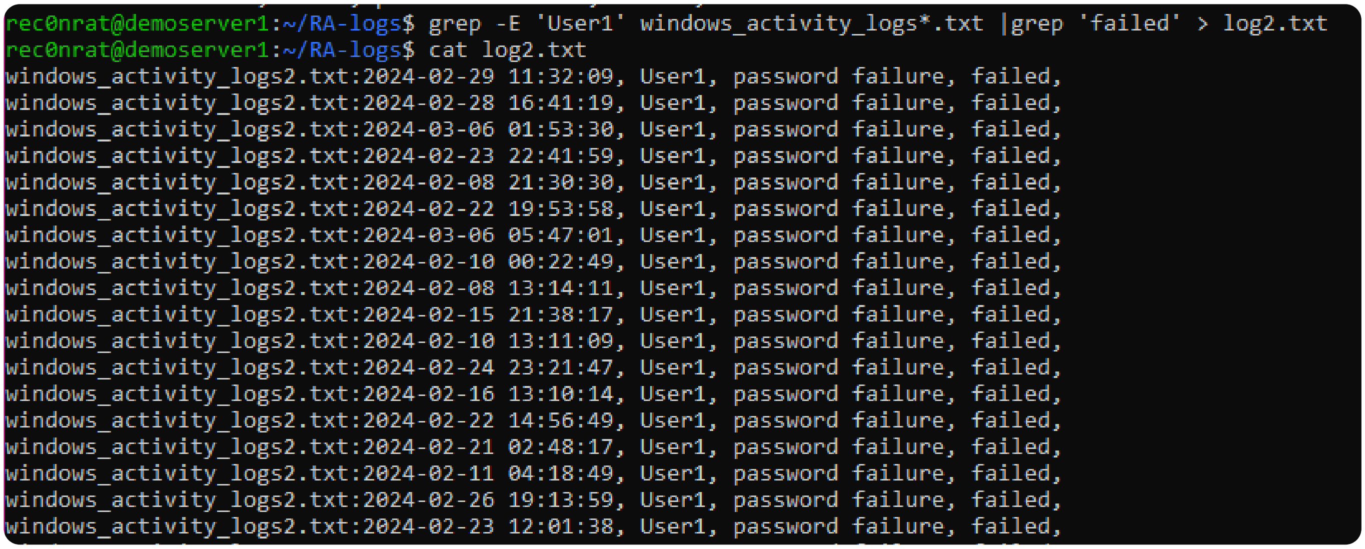
The output of from grep displays all lines in log that contain contain either string or both strings.

grep for the string 'User1' and 'failed' in 'windows_activity_logs.txt' and append the result to 'log1.txt'. In order to grep for both strings grep for the first string and then pipe the result into grep searching for the second string. The result should be the output of searching for the first string refined by the search for the second string. That output will then be appended to 'log1.txt' using the characters '>>' and can be verified by using 'cat' to show the contents of that file.

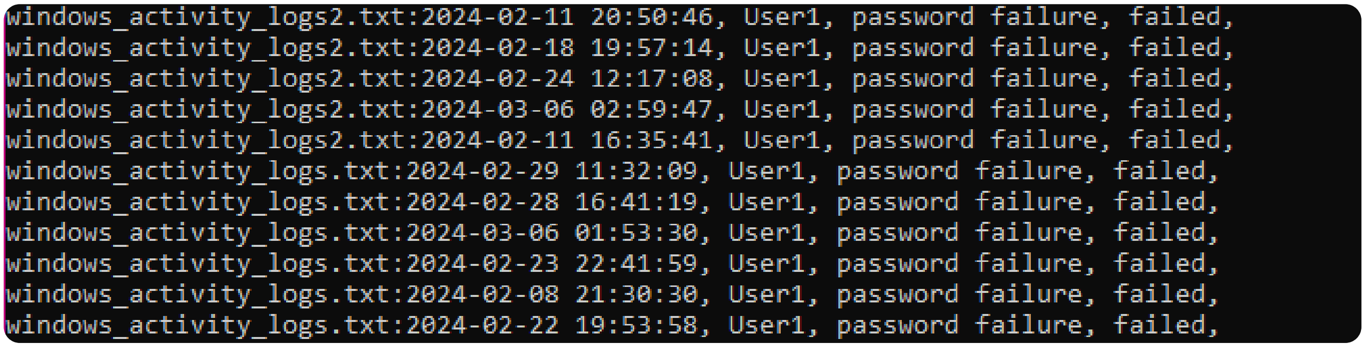


```
rec0nrat@demoserver1:~/RA-logs$ grep -E 'User1' windows_activity_logs.txt |grep 'failed' >> log1.txt
rec0nrat@demoserver1:~/RA-logs$ cat log1.txt
2024-02-29 11:32:09, User1, password failure, failed,
2024-02-28 16:41:19, User1, password failure, failed,
2024-03-06 01:53:30, User1, password failure, 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,
```

grep for the string 'User1' and 'failed' in both files starting with 'windows_activity_logs' and redirect the output to 'log2.txt'. In order to grep for both strings grep for the first string and then pipe the result into grep searching for the second string. The result should be the output of searching for the first string refined by the search for the second string. That output will then be redirected to 'log2.txt' using the characters '>' and can be verified by using 'cat' to show the contents of that file. The lines in the log file will begin with the name of the file the line was found in because multiple files were searched.



```
rec0nrat@demoserver1:~/RA-logs$ grep -E 'User1' windows_activity_logs*.txt |grep 'failed' > log2.txt
rec0nrat@demoserver1:~/RA-logs$ cat log2.txt
windows_activity_logs2.txt:2024-02-29 11:32:09, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-28 16:41:19, User1, password failure, failed,
windows_activity_logs2.txt:2024-03-06 01:53:30, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-23 22:41:59, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-08 21:30:30, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-22 19:53:58, User1, password failure, failed,
windows_activity_logs2.txt:2024-03-06 05:47:01, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-10 00:22:49, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-08 13:14:11, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-15 21:38:17, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-10 13:11:09, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-24 23:21:47, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-16 13:10:14, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-22 14:56:49, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-21 02:48:17, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-11 04:18:49, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-26 19:13:59, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-23 12:01:38, User1, password failure, failed,
```



```
windows_activity_logs2.txt:2024-02-11 20:50:46, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-18 19:57:14, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-24 12:17:08, User1, password failure, failed,
windows_activity_logs2.txt:2024-03-06 02:59:47, User1, password failure, failed,
windows_activity_logs2.txt:2024-02-11 16:35:41, User1, password failure, failed,
windows_activity_logs.txt:2024-02-29 11:32:09, User1, password failure, failed,
windows_activity_logs.txt:2024-02-28 16:41:19, User1, password failure, failed,
windows_activity_logs.txt:2024-03-06 01:53:30, User1, password failure, failed,
windows_activity_logs.txt:2024-02-23 22:41:59, User1, password failure, failed,
windows_activity_logs.txt:2024-02-08 21:30:30, User1, password failure, failed,
windows_activity_logs.txt:2024-02-22 19:53:58, User1, password failure, failed,
```