# CLI Lab 5 – Instructor Version with Solutions

Duration: 1h20

Working Directory: ~/cli\_lab5

Files Used: fake\_syslog.txt, errors.txt, warnings.txt, summary.txt, final\_report.txt

## Step-by-Step with Solutions

1. Create a working directory and navigate into it.

mkdir -p ~/cli\_lab5 && cd ~/cli\_lab5

1. Copy or create the file fake\_syslog.txt in that folder.

cp /path/to/fake\_syslog.txt .

1. Extract lines containing 'error' and redirect to errors.txt. Do the same for 'warn' into warnings.txt.

grep -i "error" fake\_syslog.txt > errors.txt  
grep -i "warn" fake\_syslog.txt > warnings.txt

1. Generate basic statistics for errors.txt: line count, word count, character count, longest line. Save all to summary.txt.

wc -l errors.txt >> summary.txt  
wc -w errors.txt >> summary.txt  
wc -c errors.txt >> summary.txt  
wc -L errors.txt >> summary.txt

1. Preview and save the first 5 lines of errors.txt and warnings.txt to first\_errors.txt and first\_warnings.txt.

head -n 5 errors.txt > first\_errors.txt  
head -n 5 warnings.txt > first\_warnings.txt

1. Extract timestamps (first 2 fields from each line) from errors.txt and warnings.txt. Save their frequencies to error\_times.txt and warning\_times.txt.

cut -d' ' -f1,2 errors.txt | sort | uniq -c | sort -nr > error\_times.txt  
cut -d' ' -f1,2 warnings.txt | sort | uniq -c | sort -nr > warning\_times.txt

1. Append the summary and timestamp statistics into a file named final\_report.txt.

cat summary.txt > final\_report.txt  
echo '' >> final\_report.txt  
cat error\_times.txt >> final\_report.txt  
echo '' >> final\_report.txt  
cat warning\_times.txt >> final\_report.txt

1. Bonus: Try doing the timestamp count in a single pipeline (no intermediate files).

grep -i error fake\_syslog.txt | cut -d' ' -f1,2 | sort | uniq -c | sort -nr | head -n 5

1. Bonus: Identify the longest line in errors.txt using advanced CLI tools.

awk '{ print length, $0 }' errors.txt | sort -nr | head -n 1

1. Create a final report file combining all previous results (summary, error\_times, warning\_times).

Already done in step 7.

## Expected Results & Comments

* errors.txt should contain 6 lines (with ERROR/error).
* warnings.txt should contain 4 lines (with WARNING/warning).
* Most frequent timestamp might be '2023-11-01 10:01:03' or another depending on log content.
* The longest line is likely the one with 'Unable to authenticate user'.
* Pipelines help avoid temp files and make chained operations faster.
* Useful commands: grep -i, wc -L, cut -f1,2, uniq -c, sort -nr

## Reflection Questions with Solutions

1. How many lines contain 'error' and 'warn'?

There are 6 lines containing 'error' and 4 lines containing 'warn' in the fake\_syslog.txt file.

1. What are the five most frequent timestamps associated with errors?

Use: grep -i error fake\_syslog.txt | cut -d' ' -f1,2 | sort | uniq -c | sort -nr | head -n 5  
Answer will vary depending on log content, but you will get the top 5 timestamps with the most errors.

1. What is the longest line (in characters) found in errors.txt?

Use: awk '{ print length, $0 }' errors.txt | sort -nr | head -n 1  
Expected result: line around 70–80 characters (e.g., 'Unable to authenticate user').

1. What is the difference between using > and >> in your summary file?

`>` overwrites the target file, while `>>` appends content to the existing file.

1. Which options of grep and wc did you find most useful and why?

`grep -i` for case-insensitive search, `grep -n` for line numbers, `wc -l` for line count, `wc -L` for longest line length.

1. What did using a pipeline help you do more efficiently?

Pipelines let you chain multiple commands without creating intermediate files, saving time and reducing disk usage.