

# CSCI 130 Linux Fundamentals - Week 9 (October 21, 2025) Assignment Submission

---

## Assignment Details

### Part 1 - Extract IP addresses of the visitors (10 points)

Using the NASA log file, extract IP addresses of the visitors. It is a first field but there may be a domain name in place of the IP address. I want you to extract IP addresses only. It is a web log from 1995; therefore, all IP addresses will be in the IPv4 (the ones we talked about in the class). Do not submit the entire file. Submit the first 20 entries (pipe through the head)

### Part 2 - Comparing number of unique visitors during different times of the day (10 points)

Using the NASA log file, count the unique number of visitors between 10:00 am and 11:59 am. Compare that number with that one between 4:00 pm (16:00) and 6:00 pm (18:00) Note that the time is a 4th field in the web log. You'll probably want to create an expression with the range something like that (for the time between 16:00 and 17:59): `/1995:1[67]:[0-5][0-9]/` Visitor is on the first field, therefore, after the match you may want to print the first field and pipe it through the sort command with -u flag for the unique visitors: `sort -u`

### How to submit

Submit a relevant history of your commands with appropriate comments what you did and what you observed

## Assignment Submission Notes

### Part 1 - Extract IP Addresses

```
cat NASA_access_log_Aug95 | egrep -o '([0-9]{1,3}\.){3}[0-9]{1,3}' | head -n 20
```

#### Explanation:

1. `cat NASA_access_log_Aug95`: This command reads the contents of the NASA log file.
2. `egrep -o '([0-9]{1,3}\.){3}[0-9]{1,3}'`: This command uses extended grep to extract only the IP addresses from the log file. The regular expression matches patterns that look like IPv4 addresses.
3. `head -n 20`: This command limits the output to the first 20 entries.

#### Output (first 20 IP addresses):

```
133.43.96.45
133.43.96.45
```

```
133.43.96.45
133.43.96.45
133.43.96.45
133.43.96.45
133.43.96.45
133.43.96.45
133.68.18.180
133.68.18.180
133.68.18.180
133.68.18.180
133.43.96.45
133.43.96.45
133.68.18.180
133.43.96.45
133.68.18.180
205.163.36.61
205.163.36.61
205.163.36.61
```

## Part 2 - Comparing Unique Visitors

```
#!/bin/bash

# Variables for the two time ranges
morning_pattern='1995:1[01]:[0-5][0-9]'      # 10:00-11:59 (matches 10:XX and 11:XX)
evening_pattern='1995:1[678]:[0-5][0-9]'    # 16:00-18:59 (matches 16:XX, 17:XX, 18:XX)

# File path to the NASA log file
log_file="NASA_access_log_Aug95"

# Check if the log file exists
if [ ! -f "$log_file" ]; then
    echo "Log file not found: $log_file"
    exit 1
fi

# Count unique visitors in the morning (10:00 am - 11:59 am)
echo "Counting unique visitors between 10:00 AM and 11:59 AM..."
unique_morning=$(egrep -a "$morning_pattern" "$log_file" | awk '{print $1}' | sort -u | wc -l)
# Display the list of unique visitors in the morning
echo "Unique visitors in the morning:"
egrep -a "$morning_pattern" "$log_file" | awk '{print $1}' | sort -u

echo "-----"

# Count unique visitors in the evening (4:00 pm - 5:59 pm)
echo "Counting unique visitors between 4:00 PM and 5:59 PM..."
unique_evening=$(egrep -a "$evening_pattern" "$log_file" | awk '{print $1}'
```

```

| sort -u | wc -l)
# Display the list of unique visitors in the evening
echo "Unique visitors in the evening:"
egrep -a "$evening_pattern" "$log_file" | awk '{print $1}' | sort -u

# Display results
echo "=====
echo "Unique visitors comparison:"
echo "Morning (10:00-11:59): $unique_morning visitors"
echo "Evening (16:00-17:59): $unique_evening visitors"
echo "=====

# Calculate difference and display which time period had more unique
visitors
if [ "$unique_evening" -gt "$unique_morning" ]; then
    diff=$((unique_evening - unique_morning))
    echo "Evening had $diff more unique visitors than morning"
elif [ "$unique_morning" -gt "$unique_evening" ]; then
    diff=$((unique_morning - unique_evening))
    echo "Morning had $diff more unique visitors than evening"
else
    echo "Both time periods had the same number of unique visitors"
fi

```

### Explanation:

1. The script defines two time patterns for morning (10:00-11:59) and evening (16:00-18:59). I used regular expressions (`1995:1[01]:[0-5][0-9]` and `1995:1[678]:[0-5][0-9]`) to match the respective time ranges and assigned them to variables (`morning_pattern` and `evening_pattern`) to make it easier later in the script.
2. I also defined a variable `log_file` to hold the path to the NASA log file, making it easier to reference throughout the script.
3. Next, the script checks if the NASA log file exists. If not, it prints an error message and exits (`exit 1`).
4. After confirming the file exists, the script counts unique visitors for both time periods using `grep`, `awk`, `sort -u`, and `wc -l`. I also added the `-a` flag to `grep` so that the output is clean (avoids binary file warnings).
5. I also output the list of unique visitors for both time periods.
6. I output the counts for both time periods (`echo "Morning (10:00-11:59): $unique_morning visitors"; echo "Evening (16:00-17:59): $unique_evening visitors"`).
7. Finally, the script compares the two counts and displays which time period had more unique visitors, or if they were equal.

### Example Output:

```

Counting unique visitors between 10:00 AM and 11:59 AM...
Unique visitors in the morning:
02-17-05.comsvc.calpoly.edu
07mb369b.uni-duisburg.de
1032015.ksc.nasa.gov

```

```
...
-----
Counting unique visitors between 4:00 PM and 5:59 PM...
Unique visitors in the evening:
04-dynamic-c.wokingham.luna.net
052.215.goodnet.com
1032015.ksc.nasa.gov
...
=====
Unique visitors comparison:
Morning (10:00-11:59): 11872 visitors
Evening (16:00-17:59): 16639 visitors
=====
Evening had 4767 more unique visitors than morning
```