

Project Title: "PingStorm – Internet Service Latency Checker"

Concept:

Students will build a Bash tool called **PingStorm** that measures and compares internet latency to several **major online services** (like Google, Facebook, TikTok, etc.). The tool will simulate a basic "speed test" by checking how quickly each of these websites responds to a ping, then analyze and visualize the results.

The project will use real **ping** commands to test reachability and response time, allowing students to understand **network latency**, work with **arrays and parsing**, and create **useful and visually clean CLI tools**.

Main Features:

1. Target Service List

Includes a predefined list of major online platforms (such as google.com, facebook.com, tiktok.com, youtube.com, and netflix.com). The tool should loop through each service for testing.

2. Ping Execution

For each domain, the tool sends multiple pings and extracts the average latency. It must handle errors gracefully (e.g., when a host is unreachable or times out).

3. Result Analysis

The tool identifies the fastest and slowest responding services, calculates an overall average latency, and produces a sorted ranking by speed.

4. Terminal Visualization

Latency data is displayed in a user-friendly format using aligned text or ASCII bar charts. The fastest and slowest results should be visually distinguished.

5. Export to CSV

Optionally, the tool allows exporting results to a CSV file with headers, so the data can be viewed in a spreadsheet.

6. Logging System

All key actions and results are logged to a central file (`pingstorm.log`) with timestamps and module identifiers

7. Control System

Create a script that starts/stops/status/shows the last 10 log lines for pingStorm.

Step-by-Step Instructions: Processing and Using the Ping Results

1. Store Results

After pinging each domain, extract the average latency from the output. Store the results in a structured format like a plain text file, where each line contains the domain name and its average response time. This file will be used for analysis and visualization.

2. Analyze the Data

Read the stored results and:

- Identify the fastest and slowest services.
- Sort all entries by latency.
- Display a ranking of services from fastest to slowest.
- Optionally calculate the average latency across all services.

3. Visualize the Results

Create a terminal-friendly output that shows each service's latency as a horizontal bar or aligned row. Normalize bar lengths to visually compare response times. Optionally use color to highlight fast, average, and slow connections.

4. Export the Data (Optional)

Allow the user to export results to a .csv file with headers like "Domain,Latency (ms)" for use in spreadsheets or external tools.

5. Final Output Example

PINGSTORM REPORT

Tested 5 services

Average Latency: 60.1 ms

✅ Fastest: google.com (32.4 ms)


😞 Slowest: tiktok.com (89.1 ms)

Visual:

google.com | ██████████ 32.4 ms

facebook.com | ██████████ 58.7 ms

tiktok.com | ██████████ 89.1 ms

 Results saved to ping_results.csv

6. Implement Logging (Required)

All scripts in the PingStorm tool should write log entries to a central file named `pingstorm.log`. This log will help track the operation of each module and is an essential part of the system.

17 July What to Log:

- The date and time of each action
- The name of the script or module executing the action
- The target domain or service being tested
- The result of the action (latency, timeout, unreachable, etc.)
- Any errors or warnings

Format:

Each log entry should follow a consistent format that includes:

- Timestamp
- Script/module name
- Description of the action or result

Entries should be appended to the log file and never overwrite previous data. The log file should be readable by all team members.

Optional Enhancements:

- Add log tags such as [INFO], [WARNING], [ERROR] for clarity.
- Allow users to view or clear the log via command-line options.

Further Reading and Resources

♦ Understanding Network Latency

- "What is latency?" by Cloudflare:
<https://www.cloudflare.com/learning/performance/glossary/what-is-latency/>
- "Understanding Network Latency" by Globalping:
<https://blog.globalping.io/understanding-network-latency-a-guide-for-beginners/>

♦ Understanding DNS

- "Explain DNS Like I'm Five" on DEV Community:
<https://dev.to/ben/explain-dns-like-i-m-five-442o>
- "How DNS services impact network latency" by Kadiska:
<https://kadiska.com/how-dns-services-impact-network-latency/>

♦ Creating Terminal User Interfaces (TUI)

- "Writing a TUI in BASH" by Dylan Araps:
<https://github.com/dylananaraps/writing-a-tui-in-bash>
- "Writing terminal GUIs with dialog and jq" on Medium:
<https://kodegeek-com.medium.com/writing-beautiful-terminal-gui-on-linux-with-dialog-and-jq-5a2852f24db8>
- "How to create terminal GUI?" (Reddit discussion):
https://www.reddit.com/r/bash/comments/sopdmw/how_to_create_terminal_gui/