Placement Empowerment Program

Cloud Computing and DevOps Centre

shell script for monitoring logs

Name: ERIC JEEVAN A   Department : ADS

Introduction

Log monitoring is an essential task for system administrators and developers. It helps track errors, warnings, and critical system messages in real-time. This guide will help you write a **Shell Script in Git Bash** to monitor server logs and alert you when errors are detected.

Overview

The script will:
✅ Monitor a log file in real-time.
✅ Search for the keyword **"Error"** in the logs.
✅ Alert you whenever an error is found.

Since Git Bash is used instead of Linux, we will ensure compatibility with Windows.

Step-by-Step Overview

Step 1: Open Git Bash

Step 2: Create a Log File

    touch server.log

- To add all files in the

    echo "Info: Service started" > server.log

    echo "Warning: High memory usage" >> server.log

    echo "Error: Service crashed" >> server.log

- d## Step 3: Write the Shell Script

```
#!/bin/bash
LOG_FILE="server.log"
KEYWORD="Error"
while true; do
    if tail -n 10 "$LOG_FILE" | grep "$KEYWORD"; then
        echo "ALERT: An error was found in the logs!"
    fi
    sleep 5  # Check logs every 5 seconds
```

done

Step 4: Make the Script Executable

chmod +x monitor_logs.sh

git commit -m "Initial commit"

Step 5: Run the Script

```
./monitor_logs.sh
```

Expected Outcome

While the script is running, open a new Git Bash window and type:

```
echo "Error: Database connection failed!" >> server.log
```

If the script is working correctly, you should see this message:

```
ALERT: An error was found in the logs!
```