

# Bash Scripting Suite for System Maintenance

**By:** Subhashree Dash  
**Institute:** ITER, SOA University  
**Department:** Computer Science and Engineering

**Objective:** This project demonstrates the use of Linux shell scripting for performing system maintenance tasks like data backup, log monitoring, and simulated system update using bash scripts.

## Script: backup.sh

Creates a compressed backup (.tar.gz) of a sample directory.

```
#!/bin/bash
# backup.sh - create a tar.gz backup of sample_data directory
TIMESTAMP=$(date +%Y%m%d_%H%M%S)
SRC_DIR="sample_data"
OUT_FILE="backup_${TIMESTAMP}.tar.gz"
tar -czf "$OUT_FILE" "$SRC_DIR"
echo "Backup created: $OUT_FILE"
ls -lh "$OUT_FILE"
```

## Script: system\_update.sh

Simulates a system update process.

```
#!/bin/bash
# system_update.sh - simulated update (no sudo in this environment)
echo "Simulating system update..."
echo "Would run: sudo apt-get update && sudo apt-get upgrade -y"
echo "Simulation complete."
```

## Script: log\_monitor.sh

Scans the system log file for ERROR and WARNING entries.

```
#!/bin/bash
# log_monitor.sh - scan system.log for ERROR or WARNING and report counts
LOGFILE="system.log"
if [ ! -f "$LOGFILE" ]; then
    echo "Log file not found: $LOGFILE"
    exit 1
fi
ERRORS=$(grep -c "ERROR" "$LOGFILE" || true)
WARNINGS=$(grep -c "WARNING" "$LOGFILE" || true)
echo "Scanned $LOGFILE"
echo "ERROR entries: $ERRORS"
echo "WARNING entries: $WARNINGS"
echo "Recent lines with ERROR or WARNING:"
grep -nE "ERROR|WARNING" "$LOGFILE" || true
```

## Script: maintenance\_menu.sh

Provides an interactive menu to run all scripts easily.

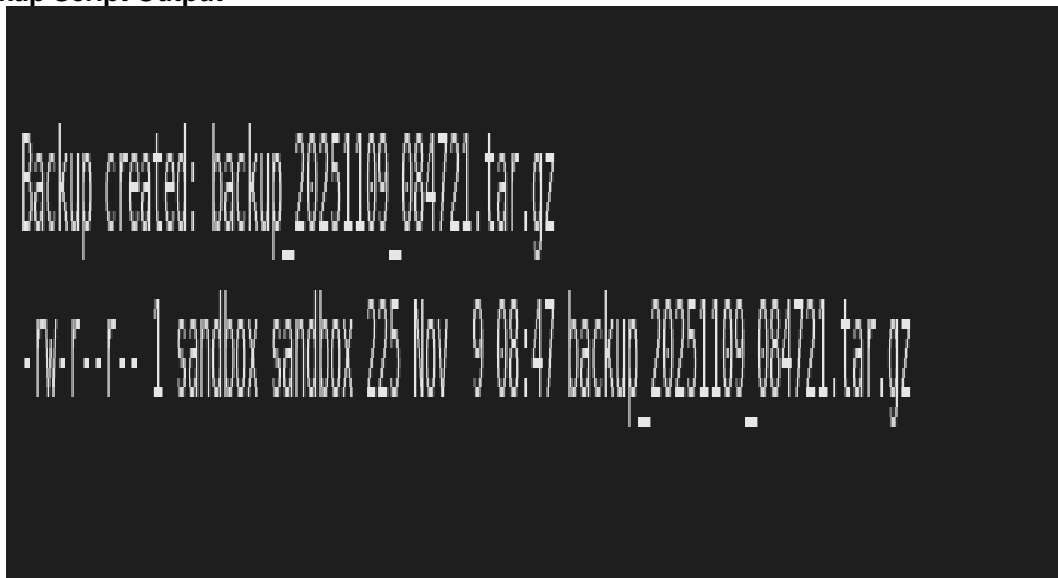
```
#!/bin/bash
```

```
# maintenance_menu.sh - simple menu to run maintenance scripts
PS3="Choose an option: "
options=("Create backup" "Simulate system update" "Run log monitor" "Exit")
select opt in "${options[@]}; do
    case $REPLY in
        1) ./backup.sh ;;
        2) ./system_update.sh ;;
        3) ./log_monitor.sh ;;
        4) echo "Exiting"; break ;;
        *) echo "Invalid option";;
    esac
done
```

## Execution Screenshots

The following screenshots show the outputs obtained after running each script.

### Backup Script Output



### System Update Simulation Output

```
Simulating system update...  
Would run: sudo apt-get update && sudo apt-get upgrade -y  
Simulation complete.
```

#### Log Monitor Script Output

```
Scanned system.log  
ERROR entries: 1  
WARNING entries: 1  
Recent lines with ERROR or WARNING:  
3:WARNING: 2025-11-01 12:30 Disk usage high  
4:ERROR: 2025-11-02 09:12 Failed to connect to DB
```

#### Maintenance Menu Output

```
1) Create backup 3) Run log monitor
2) Simulate system update 4) Exit
Choose an option: Scanned system.log
ERROR entries: 1
WARNING entries: 1
Recent lines with ERROR or WARNING:
3:WARNING: 2025-11-01 12:30 Disk usage high
4:ERROR: 2025-11-02 09:12 Failed to connect to DB
```

## Conclusion:

This capstone project demonstrates practical applications of Linux shell scripting for system management. The Bash Scripting Suite can be extended further to include automated cleanup, resource monitoring, and integration with cron jobs for scheduled maintenance.