Operating System Lab

QIAU - Student Projects

Teacher : Nahid Taherkhani

Assist : Ramtin Kosari

Dear Student, in This Paper there are Several Projects, You Can Select any Project You Want to Reach at Least **12 Points**, each Task has Its Points Next to It and Its Complexity is Shown by Its Color Like :

Challenging | **Hard** | **Normal** | **Easy**

Documenting and Proposal are Also Needed and are Part of Project Points

Tasks :

1. Check OS Behavior from /proc - **4 Points**
2. Search Apache2 Access Logs - **5 Points**
3. Search Earthquake Logs - **6 Points**
4. C++ Parallel Sorting Algorithms - **8 Points**
5. Implementation of Process Scheduling Algorithms - **12 Points**
6. Implementation of Deadlock Management Algorithms, Including the Ostrich Algorithm - **10 Points**
7. Integration of System Calls into an Operating System Project Inspired by xv6 - **12+ Points**
8. Compile and Install Linux, Create Kernel Module - **9 Points**
9. Implement Peterson Algorithm - **7 Points**
10. Develop Device Driver for Specific Hardware - **10 Points**
11. Health Check and Alerting System - **6 Points**
12. Automated Directory Backup Script - **4 Points**
13. Password Generator Using /dev - **2 Point**
14. Remote Server Management Tool - **4 Points**
15. Shecan DNS Configure from /etc - **2 Points**
16. Database Backup Automation - **4 Points**

TASK 1

Check OS Behaviour from /proc - *Shell Script*

Shell Script to Extract Operating System Related Information

- CPU Related
    - Extract CPU's Total Average Speed (in MHz)
    - Check Amount of Cores
    - Check CPU Model
- Memory Related
    - Check Total Memory
    - Check Free Memory (Check Live for 5 Seconds)
- Version
    - Check Kernel Version
- OS File System Related
    - Maximum Amount of Files that Can be Opened
    - Maximum Size of Pipe Buffer
- OS Kernel Information
    - OS Type
    - OS Release
- Network
    - Show Network Interfaces Name
    - Show Wireless Network Interfaces Name
    - Show List of Available Network Protocols
    - Show Amount of Received Packets of Online Network Interface
- Show OS Uptime
- Show Input Devices of Bus

Needed Knowledge :

- Commands : cat, awk, echo, grep,
- Piping (|)

Note :

- Write Exactly each /proc Path You Used for Shell Script
- Write Shell Script to Show All Information Above in Terminal

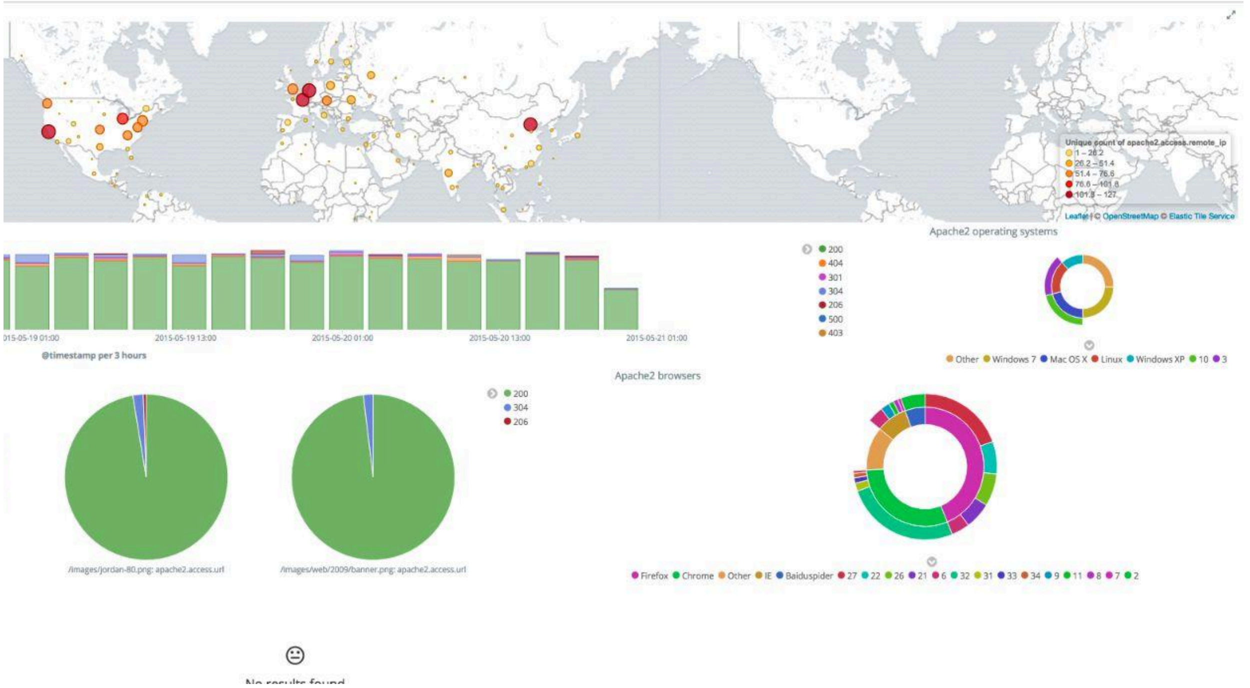
TASK 2

Search Apache2 Access Logs - *Shell Script, at Least 1 Hour*

Shell Script to Use Commands and Methods to Extract or Find Info in Log Files.

- Do These Commands to Clone Task Repository :
  - git clone https://github.com/40021441054102/OS-Lab-Projects.git
  - cd OS-Lab-Projects/Analyze-Apache-Logs
- Extract First 3 Parts of Iran’s IP Range from File **ir.csv** Like **5.134.128**
- Search **apache_logs.logs** File with Extracted IPs and See if there are any Requests from Iran

We have been Recorded 120000 Apache Access Logs and Unique IPs Map from 20th April 2017 till 27th April 2017, We Found that there Might be 1 Up to 26 Access Requests from Iran as Seen in **Apache2 Access Unique IPs Map** :



as You See in This Map, There Might be 1 Up to 26 Requests from Iran, Find All Available Requests in **apache_logs.logs** File from Iran Using IP Ranges in **ir.csv**

Needed Knowledge :

- Commands : cat, awk, echo, grep, read, IFS
- Piping (|), Shell Script Arrays and Iterating
- What is IP and IP Range?
- What is Git and Github ?

TASK 3

Search Earthquake Logs - *Shell Script, at Least 1 Hour*

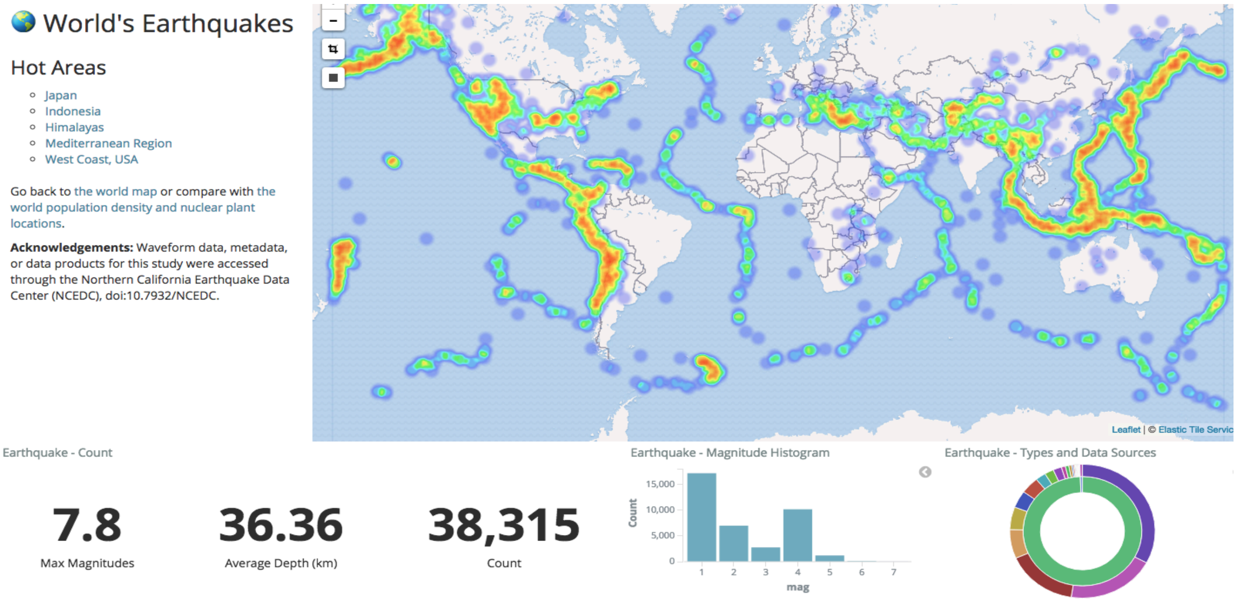
Shell Script to Use Commands and Methods to Extract or Find Info in Log Files.

- Do These Commands to Clone Task Repository :
  - git clone https://github.com/40021441054102/OS-Lab-Projects.git
  - cd OS-Lab-Projects/Analyze-Earthquake-Logs
- Extract Date, Time, Coordinates, Depth and Magnitude of each 38000 Earthquakes from **earthquakes.txt**, for Example in This Sample :

  2016/11/12 21:07:33.74,-6.3431,103.5635,42.23,4.60,Mb,,111,2,0.64,us,201611122014

  We Only Want **These** Parts :

  - 2016/11/12 21:07:33.74 - Date and Time
  - -6.3431,103.5635 - Coordinates
  - 42.23 - Depth
  - 4.60 - Magnitude
- Filter Only Records between Iran's Coordinates :
  - between Latitudes of 24° and 40° N
  - between Longitudes of 44° and 64° E
- Find Earthquake with Highest Magnitude in Iran



🌏 World's Earthquakes

Hot Areas

- Japan
- Indonesia
- Himalayas
- Mediterranean Region
- West Coast, USA

Go back to the world map or compare with the world population density and nuclear plant locations.

Acknowledgements: Waveform data, metadata, or data products for this study were accessed through the Northern California Earthquake Data Center (NCEDC), doi:10.7932/NCEDC.

Earthquake - Count

7.8 Max Magnitudes

36.36 Average Depth (km)

38,315 Count

Leaflet | © Elastic Tile Servic
Earthquake - Magnitude Histogram
15,000
10,000
5,000
0
Count
1
2
3
4
5
6
7
mag
Earthquake - Types and Data Sources

Needed Knowledge :

- Commands : cat, awk, echo, grep, read, IFS
- Piping (|), Shell Script Arrays and Iterating
- What are Coordinates ? Longitude ? Latitude ? E ? N ?
- What is Git and Github ?

TASK 4

C++ Parallel Sorting Algorithms - *Programming, Multi - Threading*

Implement Bubble Sort, Insertion Sort and Quicksort in Multi Threading Using C++ Thread, Semaphore and … Library

- Implement Parallel Bubble Sort, Insertion Sort and Quicksort
- Use C++ Threading Libraries Like **threads**
- Compare each Sorting Algorithms with and without Using Threads
- You have to Sort Array of Random Numbers Generated with **random** Library, Use Array Size of 5000 Indexes
- Calculate and Show Speed of each Sorting Algorithms with and without Using Threads
- Explore Setting Thread Affinities to Specific CPU Cores to Control the Assignment of Threads to Cores. Evaluate the Impact on Performance.

TASK 5

Implementation of Process Scheduling Algorithms - *Algorithms, Programming*

Create a Simulation Tool that Allows Users to Explore and Compare the Performance of Various Process Scheduling Algorithms in the Context of an Operating System

- Define a Data Structure to Represent Processes, Including Attributes Like Process ID, Arrival time, Burst time, Priority, etc.
- Implement Several Process Scheduling Algorithms, Including Shortest Job First (**SJF**), First-Come-First-Serve (**FIFO**), Shortest Remaining Time (**SRT**), Multilevel Feedback Queue (**MLFQ**), and Round Robin (**RR**).
- Generate a Set of Test Processes with Random Attributes for Simulation.
- Handle Concurrency and Parallelism Aspects, Especially in the Case of Non-Exclusive Scheduling Algorithms Where Multiple Cores are Involved.
- You Can Use any Programming Language You Want Like C++, Python and …
- Define Metrics for Evaluating the Performance of Scheduling Algorithms Like Turnaround Time, Waiting Time, Response Time

TASK 6

Implementation and Simulation of Deadlock Management Algorithms, Including the Ostrich Algorithm - *Programming, Algorithms*

Implementing and Simulating Deadlock Management Algorithms, with a Focus on the Ostrich Algorithm. Deadlocks are Situations where Processes Cannot Proceed because each is Waiting for the Other to Release a Resource, Creating a Cyclic Waiting Condition.

- Implement Ostrich Algorithm, Designed to Detect and Handle Deadlocks in a System.
- Develop a Simulation Environment that Replicates Scenarios Where Deadlocks Can Occur.
- Implement Mechanisms to Simulate Processes, Resource Allocation, and Deadlock Detection.

TASK 7

Integration of System Calls into an Operating System Project Inspired by xv6 - *Programming, System Calls*

Incorporation of a System Call Mechanism Into an Operating System Project, Drawing Inspiration from Projects Like xv6

- Implement a Set of System Calls that Mimic xv6-Like Functionality within the Operating System Project.
- Utilize System Calls to Extend the Project's Functionality, Enabling New Operations and Interactions.
- Explore the Impact of System Calls on the Overall Behavior of the Operating System.
- Log Detailed Information about System Call Invocations, Responses, and Their Impact on the Operating System.
- Write RTC Driver, USB Driver and Network Interface Driver for xv6 OS

Needed Knowledge :

- What is xv6 Operating System ?
- What is a Driver ?
- How to Use System Calls in C Family Programming Language ?

Note :

- Depends on Your Creativity in Which Part of Project You are Implementing, You Will Gain More Extra Points

TASK 8

Compile, Install and Work with Modules of Linux - *Programming, Linux*

Provide Participants with Hands-on Experience in Working with Linux Kernel Modules. Participants will Compile, Install, and Edit a Simple Kernel Module, Gaining Insight Into Kernel-Level Programming

- Download Linux from Github
- Choose a Linux Distribution with Kernel Headers Installed Like Ubuntu, Fedora
- Obtain a Sample Kernel Module Source Code. This Can be a Simple Module Like Sort Algorithms, Search or ...
- Compile Your Linux
- Compile Your Linux Module and Install It
- Verify Your Module Output

TASK 9

Implement Peterson Algorithm - *Programming, Algorithm*

Implement Peterson's Algorithm, a Classic Algorithm for Achieving Mutual Exclusion in a Concurrent Programming Environment

- Research and understand the principles behind Peterson's algorithm for achieving mutual exclusion in a two-process environment
- Choose a programming language suitable for concurrent programming like C++, Python or Java
- Set up a development environment with the necessary tools and compilers
- Write a program that implements Peterson's algorithm for two concurrent processes attempting to enter a critical section.
- Ensure that the algorithm satisfies the following properties:
    - Mutual Exclusion: Only one process can enter the critical section at a time.
    - Progress: If a process is not in the critical section and wants to enter, it eventually succeeds.
    - Bounded Waiting: There exists a bound on the number of times other processes can enter the critical section while the process is waiting.
- Modify your program to create and manage multiple threads or processes (more than two).
- Simulate concurrent access to a shared resource with critical sections.

TASK 10

Develop Device Driver for Specific Hardware - *System Call, Electronics*

Develop a Device Driver for a Custom USB Peripheral Sensor. The Sensor Must be Able to Collects Environmental Data Like Temperature or Anything Else You Want, and Communicates with the Host System via USB.

- Study the Datasheet and Specifications of the Custom Sensor to Understand Its Communication Protocol, Register Layout, and Data Format.
- Write a USB Device Driver for the Sensor, Implementing Functions for Initialization, Data Transfer, and Interrupt Handling.
- Integrate the Driver Into the Linux Kernel Source Tree or Build System, Ensuring Compatibility with the USB Subsystem and Device Enumeration.
- Functional USB Device Driver for the Custom Sensor
- Integration Patch or Module for the Linux Kernel

TASK 11

Health Check and Alerting System - *Shell Script, Service, Operating System*

Develop a Systemd Service to Run Shell Script at Start of Your Linux Operating System that Check CPU Usage, CPU Temperature and Disk Usage and if Status Reaches to the Critical Configured Value, Alert it and Save Report into Log File

- Upon system startup, the Systemd service will trigger the execution of the shell script.
- The shell script will retrieve real-time data on CPU usage, CPU temperature, and disk usage.
- If Any of the Monitored Parameters Exceed a Predefined Critical Value, an Alert Will be Generated.
- Predefined Critical Value Can be Anything You Decide
- The Alert Will Contain Details About the Parameter That Triggered the Alert and the Current System Status.
- Simultaneously, a Detailed Report Will be Saved Into a Log File for Further Analysis.
- The system will continue to monitor and generate alerts as long as it is Running.

Needed Knowledge :

- What is Systemd ?
- How to Get CPU Usage and Temperature in Linux ?

TASK 12

Automated Directory Backup Script - *Shell Script, Service, File System*

Develop a Systemd Service to Run Shell Script at Start of Your Linux Operating System that Backups any Directories You Want with Compressing It and Moving It to Backups Directory Located at User Home Directory

- Upon System Startup, the Systemd Service Will Trigger the Execution of the Shell Script.
- The Shell Script Will Identify Specified Directories to be Backed Up.
- It Will Compress the Contents of each Directory Into a Single Archive File Using Appropriate Compression Algorithms.
- The Compressed Archive Files Will be Moved to a Designated Backup Directory Located within the User's Home Directory.
- The System Will Continue to Perform These Automated Backups at each Startup, Ensuring Data Redundancy and Protection.

TASK 13

Password Generator Using /dev - *Shell Script, Operating System*

Develop a Shell Script to Generate a Random Password Using Characters of “0123456789qwertyuiopasdfghjklzxcvbnmQWERTYUIOPLKJHGFDSAZXCVBNM@#$” in User Defined Size and for Doing This, Use Random / URandom Devices Located at Kernel Device Modules Directory (/dev)

- The Shell Script Will Prompt the User to Specify the Desired Length of the Password.
- It Will Utilize the random or urandom Device to Generate Random Characters.
- The Generated Password Will Consist of Characters from the Provided Character Set: 0123456789qwertyuiopasdfghjklzxcvbnmQWERTYUIOPLKJHGFDSAZXCVBNM@#$
- The Script Will Ensure that each Character in the Password is Randomized.
- Finally, the Generated Password Will be Displayed to the User and Also Will be Copied

Needed Knowledge :

- Where is /dev and What Can I See Inside It ?

TASK 14

Remote Server Management Tool - *Shell Script, Operating System, SSH*

Develop a Shell Script to Establish Connections to Remote Servers Securely and has Options to Copy, Move, Delete and Create Files and Directories and Also Process Managements Like View, Start, Stop and Restart

- Secure Remote Connection : The Script Will Establish SSH Connections to Remote Servers Securely, Ensuring Encrypted Communication.
- File Management : Administrators Can Perform Various File Operations Remotely, Including Copying, Moving, Deleting, and Creating Files and Directories.
- Process Management : Users Can View, Start, Stop, and Restart Processes Running on Remote Servers, Providing Control Over System Resources.
- Information Retrieval : The Tool Will Also Allow Users to Retrieve Information about the Remote Server, Such as System Statistics, Disk Usage, and Network Configuration.
- Error Handling : Proper Error Handling Mechanisms Will be Implemented to Handle Connection Failures, Command Execution Errors, and Other Potential Issues Gracefully.

TASK 15

Shecan.ir DNS Configure from /etc - *Shell Script, Operating System Configs*

Develop a Shell Script to Overwrite DNS Configuration File in /etc, Backup Old DNS and Config Shecan.ir’s DNS Nameservers and Also Restart Network to Use New DNS

- Backup Existing Configuration : The Script Will Create a Backup of the Existing DNS Configuration File in /etc to Ensure Easy Rollback if Needed.
- Overwrite DNS Configuration : It Will Overwrite the Existing DNS Configuration File with New Settings, Including Shecan.ir's Nameservers.
- Restart Network Service : The Script Will Restart the Network Service to Apply the Changes Immediately, Ensuring that the System Uses the New DNS Configuration.

Needed Knowledge :

- Where is /etc and What Can I See Inside It ?
- Linux Network Configurations

TASK 16

Database Backup Automation - *Shell Script, Service, Database*

Develop a Systemd Service to Run Shell Script at Start of Your Linux Operating System that Backups any Database You Want with Compressing It and Moving It to DB_Backups Directory Located at User Home Directory

- System Startup Execution : The Systemd Service Will Trigger the Execution of the Shell Script at the Start of the Linux Operating System.
- Database Backup : The Shell Script Will Connect to Specified Databases and Initiate Backup Operations, Ensuring Data Integrity and Consistency
- Compression : Upon Successful Backup, the Script Will Compress the Backup Files to Reduce Storage Space Usage.
- Backup Directory Management : The Compressed Backup Files Will be Moved to a Designated Directory, such as "DB_Backups," Located within the User's Home Directory for Organization and Easy Retrieval.
- Shell Script Must Contains SQL Database Options that Execute Commands Like List of Tables, List of All Items of Table and ...

Needed Knowledge :

- What is Database and How to Create It ?
- What is Table of Database
- What is Query of Database

The End

- Carpe Diem