2024

Principles of Operating Systems

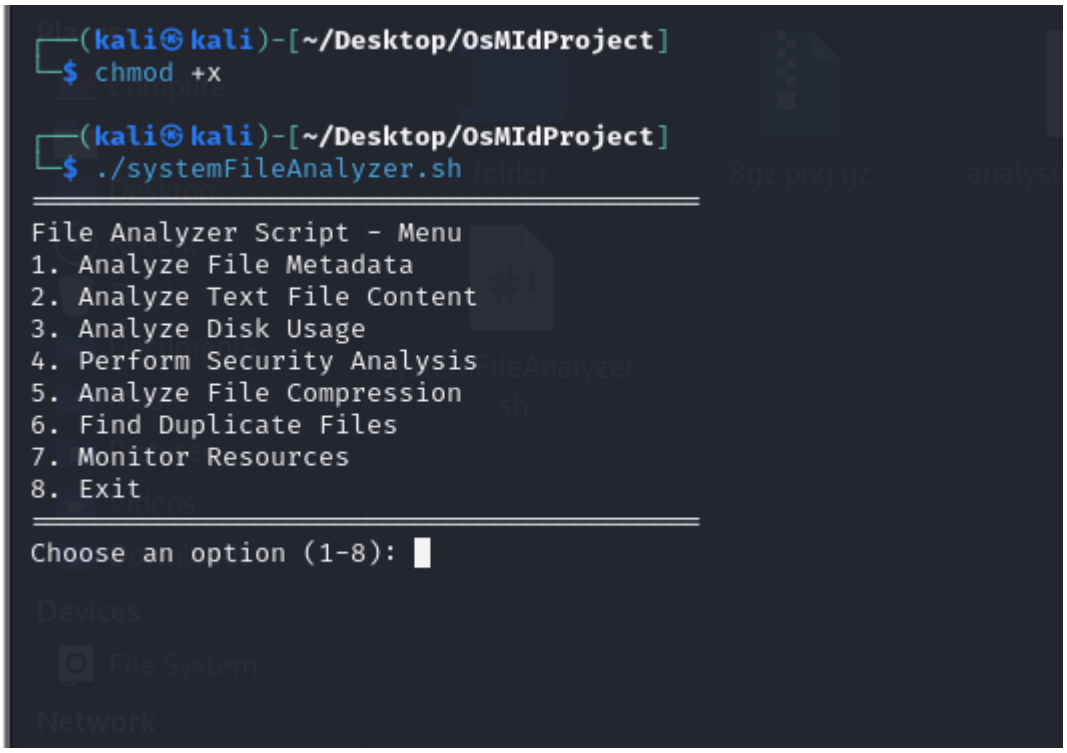
PROJECT REPORT

Contents

Project Overview: 2
Detailed Explanation of Each Feature 2
1. File Metadata Analysis 2
2. Text Content Analysis 3
3. Disk Usage Analysis 4
4. Security Analysis 4
5. Compression Analysis 5
6. Duplicate File Detection 6
7. Resource Monitoring 6
Report File Structure: 7
Summary of the System File Analyzer 7

ProjectOverview:

The *System File Analyzer* project aims to analyze and monitor file systems to offer insights into file metadata, content, security, system usage, and user behavior. This tool is intended to support system administrators and users in understanding file properties, ensuring data integrity, optimizing storage, and identifying security concerns on the operating system.



Detailed Explanation of Each Feature

1. File Metadata Analysis

- **Purpose**: To extract and display metadata information for a given file, such as file size, permissions, modification dates, and more.
- **How it works**:
    1. The script prompts the user for a file path.
    2. It verifies if the file exists and then uses the stat command to retrieve metadata and file to determine the file type.
    3. Results are both displayed on the console and appended to the report file.

```
Choose an option (1-8): 1
File Metadata Analysis

Enter the file path:
steg.png
File Path: steg.png

  File: steg.png
  Size: 31719           Blocks: 64         IO Block: 4096   regular file
Device: 8,1     Inode: 2889147     Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/    kali)   Gid: ( 1000/    kali)
Access: 2024-11-18 00:45:56.083329048 -0500
Modify: 2024-09-05 09:12:28.851963920 -0400
Change: 2024-11-18 00:45:56.079323448 -0500
 Birth: 2024-11-18 00:45:56.071312248 -0500

Detailed File Type:

steg.png: PNG image data, 170 x 148, 8-bit/color RGB, non-interlaced
```

2. Text Content Analysis

- **Purpose**: To analyze text files for word and character counts.
- **How it works**:
    1. The script prompts the user to specify a text file.
    2. It calculates the word count (wc -w) and character count (wc -m) and logs these details.
    3. This section only works with text files and logs the analysis results to the report.

```
Choose an option (1-8): 2
Text Content Analysis

Enter the text file path:
analysis_report.txt
File Path: analysis_report.txt

Word Count:

87 analysis_report.txt

Character Count:

999 analysis_report.txt
```

3. Disk Usage Analysis

- **Purpose**: To evaluate disk usage for either the entire system or a specific file.
- **How it works**:
    1. The user chooses between checking system-wide disk usage (using df) or disk usage for a specific file (using du).

```
Choose an option (1-8): 3
Disk Usage Analysis

1. Check disk usage of the entire system
2. Check disk usage of a specific file
Choose an option (1-2): 1
System-wide Disk Usage:
_____________________________________________________________________________________

Filesystem                Size  Used  Avail  Use%  Mounted  on
_____________________
/dev/sda1                 79G   26G   49G    35%   /

_____________________________________________________________________________________
```

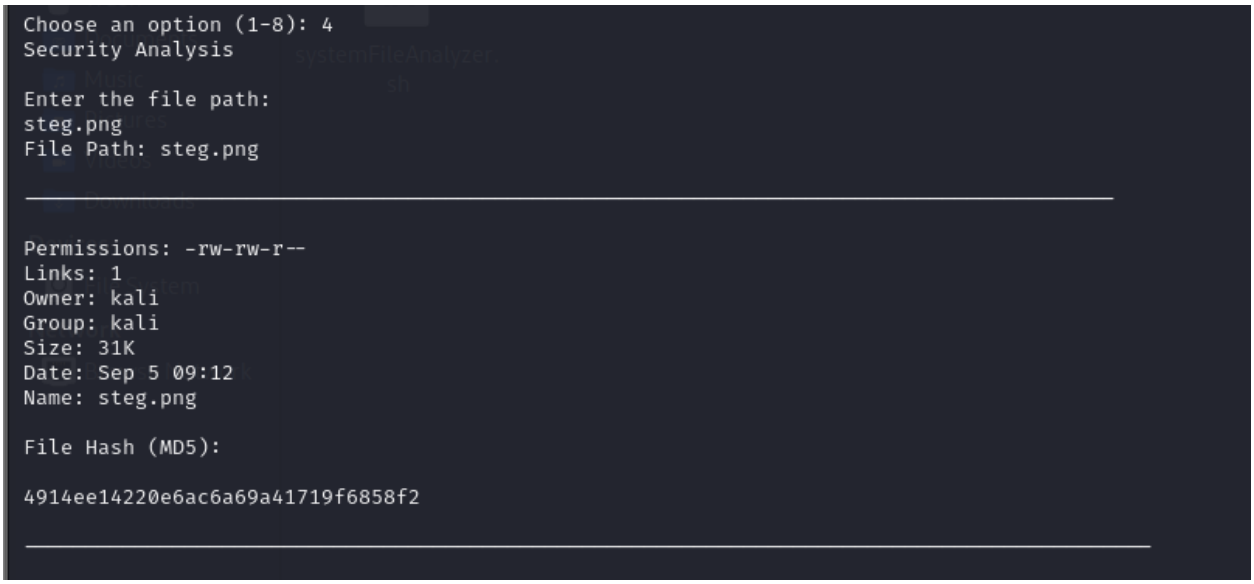
    2. Disk space usage statistics are output to both the console and the report file for review.

```
Choose an option (1-8): 3
Disk Usage Analysis

1. Check disk usage of the entire system
2. Check disk usage of a specific file
Choose an option (1-2): 2
Enter the file path:
rev.py
Disk Usage of File: rev.py

_____________________________________________________________________________

4.0K    rev.py

_____________________________________________________________________________
```

4. Security Analysis

- **Purpose**: To examine the security attributes of a file, including permissions, ownership, and file hash.
- **How it works**:
    1. The user enters the file path, and the script fetches security details using ls -lh.

2. It also generates the MD5 hash of the file using md5sum, helping to verify file integrity.
3. This information is saved to the report file for security audits or reviews.



5. Compression Analysis

- **Purpose**: To determine if a file is compressed and, if so, to identify the compression type.
- **How it works**:
    1. The script uses the file command to check the compression status and type of a given file.
    2. This feature is useful for confirming file storage formats and is logged in the report file.

6. Duplicate File Detection

- **Purpose**: To identify duplicate files based on content by calculating their MD5 hash values.
- **How it works**:
    1. The user specifies a directory, and the script calculates and compares MD5 hashes of all files in the directory.
    2. Files with matching hashes are logged as duplicates in the report file, helping in identifying redundant files and optimizing storage.

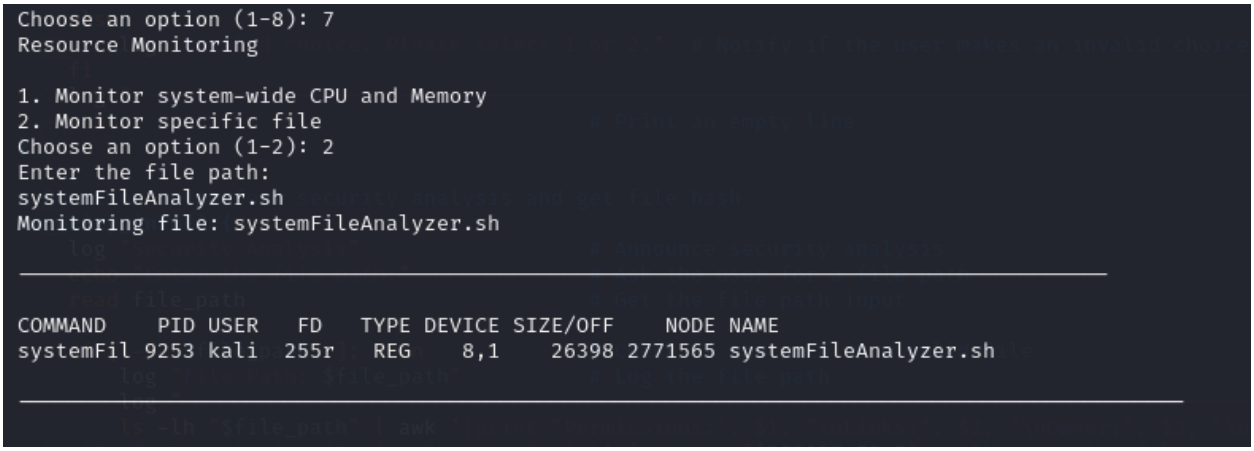
```
Choose an option (1-8): 6
Duplicate Detection

Enter the directory path:
/home/kali/Desktop/OsMIdProject
Directory Path: /home/kali/Desktop/OsMIdProject

--------------------------------------------------------------------------------

35b283642bfc2e156cb781e4283bfc93  /home/kali/Desktop/OsMIdProject/folder/rev.py
35b283642bfc2e156cb781e4283bfc93  /home/kali/Desktop/OsMIdProject/rev.py
4914ee14220e6ac6a69a41719f6858f2  /home/kali/Desktop/OsMIdProject/folder/steg.png
4914ee14220e6ac6a69a41719f6858f2  /home/kali/Desktop/OsMIdProject/steg.png

--------------------------------------------------------------------------------
```

7. Resource Monitoring

- **Purpose**: To monitor system resources, either at a system-wide level or for a specific file.
- **How it works**:
    1. The user can choose to view overall CPU and memory usage (via top) or file-specific resource usage (via du and lsof for active processes using the file).

```
Choose an option (1-8): 7
Resource Monitoring

1. Monitor system-wide CPU and Memory
2. Monitor specific file
Choose an option (1-2): 1
System-wide CPU and Memory Usage:

--------------------------------------------------------------------------------

top - 01:05:26 up 21 min,  2 users,  load average: 0.05, 0.15, 0.17
Tasks: 210 total,   2 running, 208 sleeping,   0 stopped,   0 zombie
%Cpu(s):  4.3 us,  6.4 sy,  0.0 ni, 89.4 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   2260.6 total,   1141.0 free,    779.0 used,    502.6 buff/cache
MiB Swap:   1024.0 total,   1024.0 free,      0.0 used.   1481.6 avail Mem

--------------------------------------------------------------------------------
```

2. This helps in monitoring resource consumption and identifying resource-heavy files or processes, with results saved to the report.



Report File Structure:

All outputs are saved to a central report file, `analysis_report.txt`, allowing users to keep track of their analyses over time. This report file begins with a title and timestamp and logs the results of each action as it's performed.

Summary of the System File Analyzer

This Bash script is designed to help users understand various aspects of files and system resource usage. It covers file metadata, disk usage, file security, compression status, duplicate detection, and resource monitoring. Users interact with the script through a command-line interface (CLI), where they can choose different analysis options from a menu. Each selected option performs specific tasks and outputs the results in a report file (`analysis_report.txt`) for easy reference.