Project Report

31/12/2025

23-CS-500 (Bilal)
23-CS-523 (Moaz)
23-CS-583 (Faizan)

1. Introduction
1.1. Project Overview

VirtuFS is a sophisticated storage automation framework designed to emulate enterprise-grade storage management systems. Built entirely using Bash shell scripting, it provides a virtualized file system environment where users can create, manage, secure, and monitor virtual drives without requiring physical storage devices or privileged access. The system incorporates real-world practices including Infrastructure as Code (IaC) principles, automated lifecycle management, and continuous monitoring.

1.2. Purpose and Scope

Primary Purpose: To demonstrate practical implementation of storage automation concepts through a hands-on, production-like system. The project serves as both a learning platform and a proof-of-concept for lightweight storage automation.

Scope Includes:

- Virtual drive creation and management
- Secure backup creation with encryption
- Automated scheduling and rotation policies
- User permission management
- Real-time monitoring and alerting
- Comprehensive logging and reporting
- Email-based notification system

Out of Scope:

- Hardware-level storage management
- Network-attached storage integration
- Graphical user interface
- Multi-user concurrent access management

2. Problem Statement

Modern IT infrastructure requires efficient storage management solutions that are:

- **Automated:** Reducing manual intervention and human error
- **Secure:** Protecting sensitive data from unauthorized access

- **Reliable:** Ensuring data availability and integrity
- **Monitorable:** Providing visibility into storage health
- **Cost-effective:** Operating without expensive proprietary software

Traditional approaches often involve:

- Manual backup procedures prone to errors
- Unencrypted backups risking data breaches
- No automated cleanup leading to storage bloat
- Lack of notification systems for failures
- Complex setups requiring specialized knowledge

VirtuFS addresses these challenges through a unified, script-based solution.

The primary objectives of VirtuFS are:

1. To create a virtual file system environment using directories as virtual drives.
2. To automate secure backup creation using encryption techniques.
3. To implement automatic backup scheduling and cleanup mechanisms.
4. To provide user access control and permission management.
5. To maintain logs and system health reports.
6. To notify the administrator via email about backup and cleanup operations.
7. To demonstrate DevOps automation concepts using Linux tools.

3. System Architecture

VirtuFS follows a modular architecture where each functionality is divided into logical components. The system is based on a root directory named VirtuFS, created inside the user's home directory.

3.1 Directory Structure

```
VirtuFS/
├── data/
│   └── drives/        # Virtual drives
├── mounts/            # Mounted drives (symbolic links)
├── backups/
│   ├── plain/         # Temporary decrypted backups
│   └── encrypted/     # Encrypted backup archives
├── logs/
│   └── virtuFS.log    # System log file
└── virtuFS.sh         # Main automation script
```

Each directory serves a specific role in the system, ensuring clean separation of responsibilities and maintainability.

4. Drive Management Module

4.1 Drive Creation

VirtuFS allows the creation of virtual drives using directories. Each drive is created inside the data/drives directory. The script verifies whether a drive already exists before creating a new one, preventing accidental overwrites.

4.2 Drive Deletion

Drive deletion is protected by ownership checks. Only the owner of the drive is allowed to delete it. This ensures security and prevents unauthorized data removal.

4.3 Mount and Unmount Mechanism

Mounting in VirtuFS is implemented using symbolic links. When a drive is mounted, a symbolic link is created inside the mounts directory pointing to the actual drive directory. Unmounting removes the symbolic link without deleting the actual data.

This approach simulates real Linux mount and unmount behavior in a safe and user-space environment.

5. Backup and Restore System

5.1 Backup Creation

The backup system is one of the core features of VirtuFS. It performs the following steps:

1. Compresses the selected drive into a .tar.gz archive.
2. Encrypts the archive using AES-256-CBC encryption via OpenSSL.
3. Deletes the plain archive after encryption for security.
4. Stores the encrypted backup in the backups/encrypted directory.

This ensures that sensitive data remains protected even if backup files are accessed by unauthorized users.

5.2 Backup Restoration

Restoration involves decrypting the encrypted archive and extracting it back into the drives directory. This allows full recovery of data in case of accidental deletion or system failure.

6. Automation and Scheduling

6.1 Cron-Based Auto Backup

VirtuFS supports automated backups using cron jobs. The user can define a cron schedule (e.g., daily at 2 AM), and the system automatically triggers backups without manual intervention.

6.2 Short-Term Auto Backup

In addition to cron-based scheduling, VirtuFS includes a short-term auto backup feature that runs continuously. For example:

- Backup every 60 seconds
- Delete backups older than 120 seconds

This feature is useful for demonstrations, testing, and high-frequency backup scenarios.

7. Backup Rotation and Cleanup

To prevent storage overflow, VirtuFS includes automatic cleanup of old backups. The system deletes encrypted backup files older than a specified retention period. This ensures efficient disk space utilization and implements a basic backup rotation strategy.

8. Email Notification System

VirtuFS integrates email notifications using the msmtp mail transfer agent. Notifications are sent to the administrator's email address after:

- Successful backup creation
- Deletion of old backups

This feature provides real-time monitoring and improves reliability by informing the user about system activities.

9. User Access Control

VirtuFS includes access control mechanisms that allow:

- Changing the owner of a drive
- Modifying permissions using chmod
- Viewing owner and permission information

These features demonstrate Linux file permission management and multi-user system awareness.

10. Logging and Monitoring

All system activities are logged into a centralized log file. The logging system records:

- Informational messages
- Warnings
- Errors

Additionally, the health module provides recent backup activity and storage usage statistics, allowing administrators to monitor system status efficiently.

11. Security Considerations

Security is a key focus of VirtuFS. Major security measures include:

- AES-256 encryption for backups
- Restricted file permissions for sensitive files
- Ownership verification before destructive actions
- Secure email configuration using app passwords

12. Key Features and Strengths of VirtuFS

1. **Automated Virtual Drive Management**

VirtuFS enables the creation, deletion, mounting, and unmounting of virtual drives through a single Bash script. This simulates real Linux storage behavior while remaining safe for user-space execution, making it ideal for learning and experimentation.

2. **Modular Bash Script Design**

The script is written in a modular structure where each functionality is implemented as a separate function. This improves readability, maintainability, debugging efficiency, and future extensibility.

3. **Strong Data Encryption Using OpenSSL**

VirtuFS uses AES-256-CBC encryption to secure backup archives. This ensures confidentiality of stored data and demonstrates real-world security practices used in enterprise systems.

4. **Automated Backup Creation**

The system can generate backups automatically without manual intervention. This reduces human error and reflects industry-standard DevOps automation workflows.

5. **Short-Term High-Frequency Backup Support**

VirtuFS supports continuous short-term backups (e.g., every 60 seconds), which is useful for demonstrations, testing environments, and scenarios requiring frequent data protection.

6. **Backup Rotation and Retention Policy**

Old backups are automatically deleted based on a defined retention period. This prevents disk space exhaustion and demonstrates efficient storage lifecycle management.

7. **Secure Backup Restoration Mechanism**

Encrypted backups can be safely decrypted and restored to their original location. This feature ensures data recovery in case of accidental deletion or system failure.

8. **Email Notification System**

VirtuFS sends email notifications after backup creation and cleanup events using msmtp. This enables real-time monitoring and aligns with professional system alerting practices.

9. **Cron-Based Scheduling Support**

The project supports cron jobs for long-term scheduled backups. This allows unattended execution and demonstrates Linux task scheduling skills.

10. **User Ownership Verification**

Before performing critical operations such as deletion or backup, VirtuFS checks drive ownership. This prevents unauthorized access and enforces basic security policies.

11. **Permission Management Capabilities**

Administrators can change permissions and ownership of drives using built-in commands. This highlights understanding of Linux file permission and access control concepts.

12. **Comprehensive Logging System**

All operations are logged with timestamps and severity levels (INFO, WARN, ERROR). Logs provide accountability, traceability, and easier troubleshooting.

13. **System Health and Status Monitoring**

VirtuFS includes commands to display system status, storage usage, and recent backup activity. This supports proactive system monitoring.

14. **Lightweight and Dependency-Minimal Design**

The project relies only on standard Linux utilities such as Bash, OpenSSL, cron, and msmtp. This makes it portable, efficient, and easy to deploy.

15.Industry-Relevant DevOps Learning Tool

VirtuFS closely mirrors real DevOps tasks such as automation, backup strategies, security enforcement, and monitoring. It serves as a strong portfolio project demonstrating job-ready skills.

13. Limitations

- Email delivery depends on correct SMTP configuration
- Script requires Linux environment
- No graphical user interface (CLI-based)

14. Future Enhancements

Future improvements may include:

- Cloud storage integration (AWS S3)
- Web-based dashboard
- Incremental backups
- Multi-drive backup policies
- System resource monitoring

15. Conclusion

VirtuFS is a comprehensive DevOps storage automation project that effectively demonstrates Linux automation, security, and system administration concepts. The project reflects practical industry-relevant skills and provides a strong foundation for further enhancement. Its modular design, automation features, and security mechanisms make it a valuable academic and professional learning tool.