Project Title:

"Automated Backup and Recovery System with Security Features"

Introduction

The management of student examination results is a vital task in educational institutions, y et many still rely on manual or semi-digital methods that are slow, prone to errors, and diffi cult to maintain. With growing student numbers, these traditional processes struggle to pro vide timely and accurate results. Modern web technologies offer an opportunity to automat e and simplify result processing. A web-based student result management system provides a centralized, secure, and easily accessible platform for administrators, teachers, and stu dents. This project aims to design and implement such a system to improve accuracy, efficie ncy, and accessibility in managing student academic records.

- **Problem Statement**

Data is one of the most valuable assets for individuals, businesses, and institutions. Howeve r, data is often lost due to accidental deletion, hardware failure, cyber-attacks (such as ra nsomware), or system crashes. Manual backup solutions are prone to human error, time-co nsuming, and often unreliable.

There is a need for an automated, secure, and reliable backup and recovery system that en sures continuous data availability while maintaining confidentiality and integrity.

- **Objectives**

1. Main Objective

To design and develop a secure and efficient web-based system for managing, processing, and accessing student examination results.

2.Specific Objective

1. To design and implement an automated backup system that runs on scheduled inte rvals without user intervention.

2. To ensure data confidentiality by integrating encryption mechanisms.
3. To provide recovery features with multiple versions of data (version control).
4. To integrate notifications or logging for backup status monitoring.
5. To test system performance in terms of speed, reliability, and data integrity afte r recovery.

- **Scope of the Project**

The system will be implemented on Linux/Windows environments.

It will support local backup (external hard drive/secondary partition) and cloud backup (Go ogle Drive/AWS/SFTP server).

The system will include encryption (AES/SSL) for secure data transfer and storage.

The solution will target students, SMEs, and institutions needing affordable backup solution s.

Limitations: Requires internet for cloud backup, and large files may increase processing tim e.

- **Methodology**

1. Requirements Analysis: Identify storage needs, frequency of backup, and security requirements.
2. System Design:

Define architecture (Client ↔ Backup Engine ↔ Storage Location).

Select encryption method (AES or RSA).

Define scheduling (daily, weekly, on-event).

3. Implementation:

Use Linux cron jobs / Windows Task Scheduler for automation.

Use rsync/robocopy for incremental backups.

Write automation scripts in Python/Shell.

Integrate encryption before storing backups.

Connect to cloud services using APIs (Google Drive, AWS S3).

4. **Testing:**

Perform data loss scenarios (deletion, corruption, ransomware simulation).

Test recovery speed and integrity.

Test encryption/decryption reliability.

5. **Documentation & Deployment:**

Provide user manual.

Deploy prototype for demonstration.

- **Expected Results**

A fully functional backup and recovery system that:

- Runs automatically based on schedule.
- Encrypts data before storage.
- Provides multiple versions of backup for rollback.
- Restores lost files efficiently.
- Performance metrics report (speed, success rate, security evaluation).

- **Tools & Technologies**

- Programming: Python, Bash/Shell scripting
- Backup Utilities: rsync (Linux), robocopy (Windows)
- Scheduling: Cron jobs (Linux), Task Scheduler (Windows)

- Encryption: Python cryptography library (AES-256), OpenSSL
- Cloud Services (optional): Google Drive API, AWS S3, FTP/SFTP
- Database (optional): SQLite/MySQL for logging backup activity