

# Week 12 Assignment

## Scenario: Secure Disk Partitioning and File System Configuration for a Financial Institution

**Background:** You are a system administrator responsible for managing the infrastructure of a financial institution. Security and data integrity are of utmost importance in this environment, where sensitive financial information is stored and processed. To ensure the confidentiality, integrity, and availability of data, you need to implement secure disk partitioning and file system configuration practices.

### Requirements:

#### 1. Infrastructure Overview:

- The infrastructure consists of a dedicated server hosting financial applications and databases. (Assume your AWS instance created with Terraform)
- The server runs a Linux operating system (e.g., Ubuntu) and is equipped with an additional 5 Gig disk for storage. An example terraform config that will show you how to attach a volume can be found [here](#)

#### 2. Disk Partitioning:

- Partition the new disk to segregate files effectively.
- Allocate partition sizes as follows:
  - `/var/log/audit`: 1GB for audit logs (to store audit records securely)
  - `/sysadmin`: 200MB for sys ad related files
  - `/executives`: 500MB for the CEO, CFO, CIO to utilize
  - `/analysts`: 200MB for the analysts to use.

#### 3. File System Configuration:

- Configure file systems on each partition with the following specifications:
  - Use ext4 for each partition to ensure compatibility and reliability.
  - Set appropriate parameters for each file system, including block size, inode count, and reserved blocks, to optimize performance and storage utilization.
  - Implement mount options such as `noexec`, `nosuid`, and `nodev` for `/analysts` to make sure they don't breek anything.

#### 4. Automation with Ansible:

- Utilize Ansible automation to streamline the disk partitioning and file system configuration process.
- Define Ansible variables to parameterize partition sizes, file system types, and configuration parameters, allowing for easy customization and scalability.
- Implement error handling and validation checks within the Ansible playbook to ensure robustness and reliability of the automation workflow.

#### 5. Documentation and Testing:

- Document the entire setup process, including step-by-step instructions for executing the Ansible playbook and verifying the infrastructure configuration.
- Provide explanations for design decisions, configuration choices, and any challenges encountered during the setup process.
- Conduct thorough testing of the infrastructure to validate the effectiveness of the disk partitioning and file system configurations.
- Include test cases, procedures, and outcomes in the documentation, highlighting any issues and their resolutions.

### Assignment Deliverables:

#### 1. Ansible Playbook:

- Develop an Ansible playbook to automate the disk partitioning and file system configuration tasks based on the provided requirements.
- Implement dynamic behavior using Ansible variables to customize configuration settings for each partition and file system.

#### 2. Documentation:

- Provide comprehensive documentation outlining the setup process, configuration choices, and

testing results.

- Include explanations of how Ansible variables are used to enhance playbook flexibility and scalability.
- Ensure the documentation is clear, well-organized, and provides sufficient detail for understanding the setup process and rationale behind configuration decisions.

**Submission Guidelines:** - Submit the Ansible playbook file along with the documentation of your configuration choices and testing results. - Ensure the documentation is formatted neatly and includes relevant screenshots, code snippets, and explanations where necessary.

**Assessment Criteria:** - Effectiveness of the Ansible playbook in automating disk partitioning and file system configuration tasks. - Demonstrated understanding of Ansible variables and their usage in enhancing playbook flexibility and scalability. - Clarity, completeness, and organization of documentation, including detailed explanations of design decisions and testing outcomes.