### Filesystem Monitoring Project

#### Overview

The Filesystem Monitoring project is designed to monitor and track filesystem usage on servers in real-time. It provides a comprehensive solution for administrators to monitor storage utilization, receive alerts for high usage, and visualize filesystem statistics through a web interface.

#### Components

1. **Server-side Script (stats.py)**:
   * Monitors server IP address and filesystem usage using system commands (df on Linux).
   * Connects to an Oracle database (XE) using cx\_Oracle to store filesystem statistics.
   * Runs continuously, updating filesystem usage every 5 seconds.
2. **Web Interface (webapp.jsp)**:
   * Displays filesystem usage statistics retrieved from the database.
   * Uses AJAX to fetch and update data dynamically.
   * Highlights filesystems with usage over 90% in red and triggers email alerts using emailAlert.jsp.
3. **Email Alerting (emailAlert.jsp)**:
   * Receives JSON payload with server name, filesystem, and usage percentage from webapp.jsp.
   * Sends email notifications using Gmail SMTP (javax.mail) if usage exceeds 90%.
4. **Database (filesystem\_usage table in Oracle)**:
   * Stores server name, IP address, filesystem name, and usage percentage.
   * Populated by stats.py script via cx\_Oracle.

#### Workflow

1. **Server-side Script Execution**:
   * **Filesystem Monitoring (stats.py)**:
     + Retrieves server information (hostname and IP address).
     + Executes system command (df) to fetch filesystem usage.
     + Connects to Oracle database using cx\_Oracle.
     + Inserts filesystem usage statistics into filesystem\_usage table.
     + Runs continuously in a loop every 5 seconds for real-time monitoring.
2. **Web Interface**:
   * **Display (webapp.jsp)**:
     + Fetches filesystem usage data from the database using AJAX.
     + Updates the HTML table dynamically.
     + Checks usage percentage; triggers alert (color change and sound) if usage exceeds 90%.
     + Sends alert email via emailAlert.jsp.
3. **Email Alerting**:
   * **Email Notification (emailAlert.jsp)**:
     + Receives JSON payload with server details and filesystem usage.
     + Constructs and sends an email using javax.mail if usage is critical.

#### Setup and Dependencies

* **Python Dependencies**:
  + cx\_Oracle: Install using pip install cx\_Oracle.
  + Ensure Oracle Instant Client is installed and configured.
* **Java Dependencies**:
  + javax.mail: Ensure it's included in the project's classpath.
  + Use Maven or directly include the JAR files.
* **Database**:
  + Configure Oracle database (XE) with appropriate credentials (system/root123).

#### Usage

1. **Server-side Script (stats.py)**:
   * Execute stats.py on the server.
   * Ensure Python environment has cx\_Oracle and necessary permissions to execute system commands (df).
2. **Web Interface (webapp.jsp)**:
   * Deploy webapp.jsp on a web server (Tomcat, Apache, etc.).
   * Access the interface to view real-time filesystem statistics.
3. **Email Alerting (emailAlert.jsp)**:
   * Ensure emailAlert.jsp is correctly configured with Gmail SMTP settings.
   * Modify recipients and other settings as per requirement.

#### Enhancements

* **UI/UX Improvements**: Enhance web interface with graphs, historical data views.
* **Alerting Mechanisms**: Introduce SMS alerts, integrate with monitoring tools.
* **Security**: Implement secure practices for database and email configurations