BARQ Lite – Two-Server Ops Project (Full Setup, Java JAR)

You are given **two Linux servers**: s1 and s2.

Your task is to **set up everything from scratch** for a company called **BARQ** and automate operational tasks using **Bash** and **Ansible**.

Scenario

BARQ runs a small Java-based application.
Currently, the servers have no logs, no application release file, and no TLS certificates.
Your job is to create these resources, deploy the application, and automate operations.

Part 0 – Environment Setup

On **both servers**, create the initial resources:

1. **Application Logs**
    - Create the path /var/log/barq/.
    - Create at least two .log files:
        - One with **yesterday's date** in its name.
        - One with **today's date**.
2. **Java Application Release**
    - You will be provided with a prebuilt JAR file: barq-lite.jar.
    - This application must run with:
    - java -jar barq-lite.jar
    - When started, it **writes logs** to /var/log/barq/barq.log.
3. **TLS Certificate Folder**
    - Create /etc/ssl/patrol/.
    - Generate at least one short-lived self-signed .crt for testing.

Part 1 – Log Compression & Retention (Bash + Cron)

- Write a Bash script that:
    - Compresses all log files from **yesterday** in /var/log/barq/ to .gz.
    - Deletes compressed logs older than **7 days**.
- Save as /usr/local/bin/log-lite.sh (must be executable).
- Schedule it in cron to run **daily at 01:10**.

Part 2 – Java Application Deployment (Ansible)

- Use Ansible to:
    - Upload barq-lite.jar from your local machine to both servers.
    - Store each release under /opt/barq/releases/<release_id>/.
    - Update /opt/barq/current symlink to point to the latest JAR.
    - Deploy **serially** (one server at a time).
- Add a **systemd service file** called barq.service to:
    - Run the JAR at boot with:
    - java -jar /opt/barq/current/barq-lite.jar
    - Ensure logs appear in /var/log/barq/barq.log.

Part 3 – TLS Certificate Report (Bash + Cron)

- Write a Bash script that:
    - Scans /etc/ssl/patrol/*.crt.
    - Creates /var/reports/cert-lite.txt with:
    - cert_name | NotAfter_date | days_remaining
- Save as /usr/local/bin/cert-lite.sh (must be executable).
- Schedule it in cron to run **daily at 07:00**.

Requirements

- Bash scripts must:
    - Quote all variables.
    - Handle errors gracefully.
- Ansible tasks must:
    - Be **idempotent** (re-running them should not cause unnecessary changes).
    - Use appropriate modules (file, copy, cron, systemd, etc.).
- The Java application must be deployed and confirmed running on **both servers**.
- Application logs must be written under /var/log/barq/.

Bonus Challenge – Docker (Optional)

For extra credit:

- Package barq-lite.jar into a Docker image (e.g., FROM openjdk:17-jdk).
- Write an Ansible task to:
    - Build the image (or pull it if you push it to a registry).
    - Run it as a container on each server, exposing it on port 8080.
- Verify that visiting http://<server_ip>:8080 shows the app output.

Deliverables

- Bash scripts (log-lite.sh, cert-lite.sh).
- Ansible:
    - Inventory for both servers.
    - Playbooks/roles to:
        - Create initial directories/files if missing.
        - Deploy scripts and set up cron jobs.
        - Deploy the Java application as a service.
- README.md with:

    - How to run your playbooks.
    - Any assumptions.
    - Any package dependencies.

Submission

Send:

- A .zip or Git repository with all files.