CASSIA INFRASTRUCTURE - DEPLOYMENT MANUAL

Date: January 2026

System: Ubuntu / Debian Linux

1. Overview

The Cassia Infrastructure manages the communication between BLE devices, the MQTT Broker, and the User Dashboard. This system is designed for **high resilience** in field deployments (hospitals), featuring:

- **Auto-Healing:** Services automatically restart if they crash.
- **Nuclear Port Cleanup:** Automatically kills zombie processes holding ports 1883, 8080, or 8083 during start/stop to prevent "Address already in use" errors.
- **Silent Alerting:** Crashes are reported silently to a Telegram group (Network Managers) rather than displaying confusing errors to hospital staff.
- **Strict Configuration:** Uses a Linux-optimized configuration that mirrors your specific deployment needs.

2. Prerequisites

2.1 Server Requirements

- **OS:** Ubuntu 22.04 LTS (Recommended) or Debian 11+
- **User:** Root or Sudo access
- **Internet:** Required for initial installation (GitHub downloads)

2.2 Telegram Alert Setup (Required)

Before installation, you must create a Telegram Bot to receive crash alerts.

1. Open Telegram and search for **@BotFather**.
2. Send the command: /newbot
3. Name your bot (e.g., "Hospital_Monitor_Bot").
4. Copy the **HTTP API TOKEN** provided (e.g., 123456:ABC-DEF...).
5. Search for your new bot and send it a message (e.g., "Hello").
6. Visit this URL in your browser (replace <TOKEN> with your token): https://api.telegram.org/bot<TOKEN>/getUpdates
7. Look for "id": 123456789 inside the "chat" object. This is your **Chat ID**.

3. Installation

1. **Copy Script:** Transfer cassia-setup-v2.sh to the server.
2. **Run Installer:**

   > sudo bash cassia-setup-v2.sh

3. **Select Option 1:** "Install / Reinstall".
4. **Configure Alerts:** Enter the Hospital Name, Region, Bot Token, and Chat ID when prompted. The system will send a test message to your phone.
5. **Interactive Configuration:** The script will pause and ask if you want to edit the configuration files (mqtt.conf, config.json).
   - **Note:** The installer generates a Linux-compatible mqtt.conf based on your requirements (Port 1883, Persistence, Anonymous Login).

4. Daily Operations

The system provides custom commands for easy management.

Start Services

Use this command to start the infrastructure. It will perform a "Nuclear Cleanup" of ports and ask if you want to edit configurations first.

> sudo cassia-start

Stop Services

Stops all services and **disables auto-restart**. Use this for maintenance.

> sudo cassia-stop

Restart Services

Use this if the system is acting strange. It forces a configuration reload.

> sudo cassia-restart

Check Status

Displays a dashboard of running services, listening ports, and access URLs.

> cassia-status

5. Configuration Files

All configuration files are located in /opt/cassia. You can edit them manually or use the interactive prompts in cassia-restart.

Component	File Path	Notes
MQTT Broker	/opt/cassia/mqtt.conf	Linux-adapted version of your template.
Orchestrator	/opt/cassia/orchestrator/config.json	Defines MQTT broker IP and BLE settings.
Dashboard	/opt/cassia/dashboard/config.json	Dashboard settings.

6. Logs & Debugging

Log files are stored in /var/log/cassia. You can view them using these shortcuts:

Live Monitoring (Tail)

- **MQTT:** view-mqtt
- **Orchestrator:** view-app
- **Dashboard:** view-dashboard
  (Press Ctrl+C to exit)

Historical Logs (Scrollable)

- **MQTT:** logs-mqtt
- **Orchestrator:** logs-app
- **Dashboard:** logs-dashboard
  (Press 'q' to exit)

File Manager

To see all files and open the directory:

> cassia-files

7. Troubleshooting

"Address already in use" / Boot Loop

- **Cause:** An old instance of Mosquitto or the Orchestrator is running in the background (Zombie process).
- **Fix:** Run sudo cassia-stop. The v2.7 script performs a "Nuclear Cleanup" that aggressively kills any process holding ports 1883, 8080, or 8083. Then run **sudo cassia-start**.

Alerts not arriving

- **Fix:** Check your internet connection (ping google.com).
- **Fix:** Check the config file: sudo cat /opt/cassia/alert_config.env. Ensure the Token and Chat ID are correct.

Dashboard not loading

- **Fix:** Run **cassia-status**. Ensure the Dashboard service is "Running" and Port 8080 is "Listening".
- **Fix:** Ensure you are using http://<SERVER_IP>:8080 and not localhost if accessing remotely.

8. Uninstallation

To completely remove the infrastructure:

1. Run **sudo bash cassia-setup-v2.sh**.
2. Select **Option 2 (Uninstall)**.
3. The script will:
    - Stop all services.
    - Kill all associated processes.
    - Remove all files in /opt/cassia.
    - Ask if you want to purge log files (/var/log/cassia).