## Tools for Implementing Secure Communication & Logging

## The document establishes the precise tools along with recommended technologies needed to build the Secure Communication & Logging feature into the Unified OT Ransomware Protection Platform. The recommended tools support your Python and FastAPI system while using MQTT for implementation.

## 🔐 1. Encrypt Data at Rest

## • Tool - cryptography (Python library)

## The AES-256 encryption method must be used through this tool to protect config files and backups plus logs during storage operations.

## Data encryption happens before backups save to the disk as per the example.

## 🔒 2. Secure Data in Transit

## • Tool - TLS 1.3 - For MQTT: Use Mosquitto MQTT broker with TLS/SSL enabled.

## - For FastAPI: Run over HTTPS with SSL certificates via uvicorn or Nginx.

## The API traffic and device messages and system commands need protection through this tool.

## ✉️ 3. Verify Message Integrity

## • Tool - Python hmac module or cryptography.hazmat.primitives.hmac

## Digital signatures can be attached with this tool while it verifies that message contents have not been altered.

## 📋 4. Logging System Activity

## • Tool - logging (Python built-in)

## The system should record all alerts and backups as well as isolations with this tool.

## The storage of logs should happen in files that enable only reading or appending new data because tampering attempts become impossible.

## 📊 5. Visualizing and Managing Logs

## • Tools - ELK Stack (Elasticsearch, Logstash, Kibana), or Wazuh (optional)

## The system uses this tool for both big log analysis tasks and dashboard creation and alert preparation.

## 

## 🔑 6. Managing Secrets and Keys

## • Tool - HashiCorp Vault (or .env files for small setups) The system serves to secure AES encryption keys and HMAC secrets by implementing proper storage and management functions.

## 📡 7. Securing MQTT Communication

## • Tool - Mosquitto MQTT Broker

## • Steps:

## - Enable TLS via mosquitto.conf

## The system should authenticate users with username and password credentials.

## - Log MQTT events securely

## 🖥️ Recommended Tool Stack

|  |  |
| --- | --- |
| Feature | Tool(s) to Use |
| Encrypt files/logs | Python cryptography (AES-256) |
| Secure API & frontend comms | FastAPI + HTTPS (TLS 1.3 via Nginx) |
| Secure IoT/sensor data | MQTT + Mosquitto with TLS |
| Logging actions | Python logging, ELK/Wazuh (optional) |
| Prevent log tampering | Append-only logs / read-only folders |
| Message integrity | Python hmac module |
| Secret/key management | HashiCorp Vault / .env files |