Tools for Implementing Secure Communication & Logging

This document outlines the recommended tools and technologies required to implement Secure Communication and Logging for the Unified OT Ransomware Protection Platform. These tools are aligned with Python and FastAPI systems while using MQTT for real-time secure messaging.

# 1. Encrypt Data at Rest

Tool: cryptography (Python library)  
Use AES-256 encryption to secure configuration files, backups, and logs before they are stored on disk.

# 2. Secure Data in Transit

Tool: TLS 1.3  
- For MQTT: Use Mosquitto MQTT broker with TLS/SSL.  
- For FastAPI: Run over HTTPS using SSL certificates (via uvicorn or Nginx).

# 3. Verify Message Integrity

Tool: Python hmac module or cryptography.hazmat.primitives.hmac  
Use HMAC to ensure that messages are not altered in transit by verifying their digital signatures.

# 4. Logging System Activity

Tool: logging (Python built-in)  
Log all critical actions such as alerts, backups, and isolations. Logs should be append-only or in read-only folders to prevent tampering.

# 5. Visualizing and Managing Logs

Tools: ELK Stack (Elasticsearch, Logstash, Kibana), or Wazuh  
Use these tools to analyze large volumes of logs, create dashboards, and generate alerts for anomalies.

# 6. Managing Secrets and Keys

Tool: HashiCorp Vault (or .env files for small setups)  
Secure storage and management of AES encryption keys and HMAC secrets.

# 7. Securing MQTT Communication

Tool: Mosquitto MQTT Broker  
Steps:  
- Enable TLS via mosquitto.conf  
- Authenticate using username/password  
- Log all MQTT events securely.

# Recommended Tool Stack

|  |  |  |
| --- | --- | --- |
| **Feature** | **Tool(s) to Use** | **Purpose** |
| Encrypt files/logs | Python cryptography (AES-256) | Protect data at rest. |
| Secure API & frontend comms | FastAPI + HTTPS (TLS 1.3 via Nginx) | Ensure encrypted communication. |
| Secure IoT/sensor data | MQTT + Mosquitto with TLS | Encrypted message transport. |
| Logging actions | Python logging, ELK/Wazuh (optional) | Track critical system events. |
| Prevent log tampering | Append-only logs / read-only folders | Ensure logs remain immutable. |
| Message integrity | Python hmac module | Verify authenticity of messages. |
| Secret/key management | HashiCorp Vault / .env files | Safely store and manage keys. |