AI Applications for Cybersecurity

Infrastructure Monitoring and Protection

Duration: 5 Days

Course Summary

This 5-day practical, hands-on training is designed for professionals who manage on-premises infrastructure independently across mixed Mac, Linux, and Windows environments. The programme focuses on building and enhancing a local AI setup — using tools like Ollama and Gemma — to monitor systems, analyse logs, detect threats, and automate routine operational tasks, all without exposing sensitive institutional data to cloud services. No deep programming or data science background is required. Participants will leave with a working local AI monitoring agent and a practical 30-day implementation plan.

Target Audience

This programme is designed for:

- Infrastructure managers handling Mac, Linux, and Windows servers
- Small teams managing on-premises systems without dedicated security staff
- Professionals already using local AI models wanting to go further
- Operations staff responsible for uptime, patching, and system health
- Anyone who needs AI to work smarter with limited team resources

No programming or data science background required. Basic comfort with command-line and file systems is helpful.

Note: All tools in this programme run locally on premises. No data is sent to cloud services at any point.

Day 1: AI in Cybersecurity Foundations and Threat Detection

Module 1: Local AI for Infrastructure — Concepts and Setup

- Local AI vs cloud AI
- Ollama and Gemma overview
- On-premises data safety
- Mixed OS environments
- Practical monitoring use cases

Lab Exercises:

1. Set up Ollama locally and run a Gemma model — send it a sample log file and ask it to summarise any errors found.
2. Map 5 routine infrastructure tasks to the local AI tools that can assist with each one.

Module 2: AI-Driven Threat Detection and Response

- Threat detection pipeline

- Indicator of compromise
- Automated triage
- MITRE ATT&CK framework
- Response playbooks

Lab Exercises:

3. Map a fictional multi-stage attack to the MITRE ATT&CK framework and identify the detection point for each stage.
4. Design an AI-driven threat detection and automated triage workflow for a fictional infrastructure environment.

Day 2: Infrastructure Monitoring and Log Analysis

Module 3: Monitoring Mac, Linux, and Windows with Prometheus and Grafana

- Multi-OS agent setup
- CPU, memory, disk tracking
- Prometheus configuration
- Grafana dashboards
- Threshold-based alerts

Lab Exercises:

5. Deploy Prometheus node exporters on Mac, Linux, and Windows hosts and visualise all three in a single Grafana dashboard.
6. Set baseline thresholds for each server type and configure Grafana alerts that fire when behaviour deviates from normal.

Module 4: Log Analysis and AI-Assisted Anomaly Detection

- Log collection basics
- ELK stack setup
- Log searching and filtering
- AI-assisted log analysis
- Anomaly dashboards

Lab Exercises:

7. Ingest server logs from Mac, Linux, and Windows hosts into the ELK Stack and build a unified log dashboard.
8. Feed a sample log file to a local Gemma model via Ollama and ask it to identify the top 5 unusual events in plain English.

Day 3: Network Security and Endpoint Monitoring

Module 5: Network Monitoring Without Deep Networking Knowledge

- Network monitoring basics
- Zeek for traffic capture
- Unusual connection detection
- Internal traffic patterns
- DNS anomaly spotting

Lab Exercises:

9. Use Zeek to analyse a provided PCAP file and identify the top 3 suspicious connection patterns in plain-English output.
10. Feed Zeek network alerts to a local Gemma model and ask it to summarise the most concerning findings.

Module 6: Keeping an Eye on Devices and Processes

- Wazuh on Mac, Linux, Windows
- File integrity monitoring
- Process behaviour alerts
- Suspicious activity detection
- Basic threat hunting

Lab Exercises:

11. Deploy Wazuh agents on Mac, Linux, and Windows endpoints and verify all three report events to the central Wazuh manager.
12. Simulate a suspicious file change on a test host and verify Wazuh raises a file integrity alert with correct severity.

Day 4: Predictive Alerts and Security Automation

Module 7: Predicting Problems Before They Happen

- Trend-based prediction
- Downtime forecasting
- Capacity planning basics
- AI early warning setup
- Preventing outages

Lab Exercises:

1. Use a provided script to train a simple prediction model on server metrics and generate an alert 30 minutes before a likely capacity issue.
2. Ask a local Gemma model to analyse a week of server metric trends and predict which server is most at risk of downtime.

Module 8: Security Automation and Incident Response

- Automation for small teams
- Shuffle SOAR setup

- Playbook design basics
- Auto-alerts and tickets
- Reducing manual work

Lab Exercises:

3. Build a Shuffle automation that notifies you instantly when Wazuh raises a high-severity alert on any host — no coding required.
4. Create a workflow that automatically logs every Wazuh alert to a shared tracking sheet so your team stays informed.

Day 5: AI Security Best Practices and Applied Project

Module 9: Keeping Data Private and AI Safe to Use

- Local vs cloud AI risk
- What data stays on-prem
- Gemma safe usage rules
- False positive management
- Human-in-the-loop

Lab Exercises:

5. Audit your AI monitoring setup for data leakage risks and define a policy for what data may and may not be processed by local AI models.
6. Design a 5-step human-in-the-loop review checklist for verifying Gemma-generated alerts before acting on them.

Module 10: Applied Project — Build Your Local AI Monitoring Agent

- Local AI agent design
- Ollama and Gemma integration
- Full monitoring stack
- Automation workflows
- 30-day rollout plan

Lab Exercises:

7. Integrate Ollama, Wazuh, Prometheus, and the ELK Stack into a unified local AI monitoring agent that analyses alerts and logs in plain English.
8. Produce a 30-day rollout plan mapping each tool to a specific daily task it will handle — from log review to downtime prediction to alert triage.