Azure SOC Lab – Microsoft Sentinel End-to-End Detection & Response Project

📌 Overview

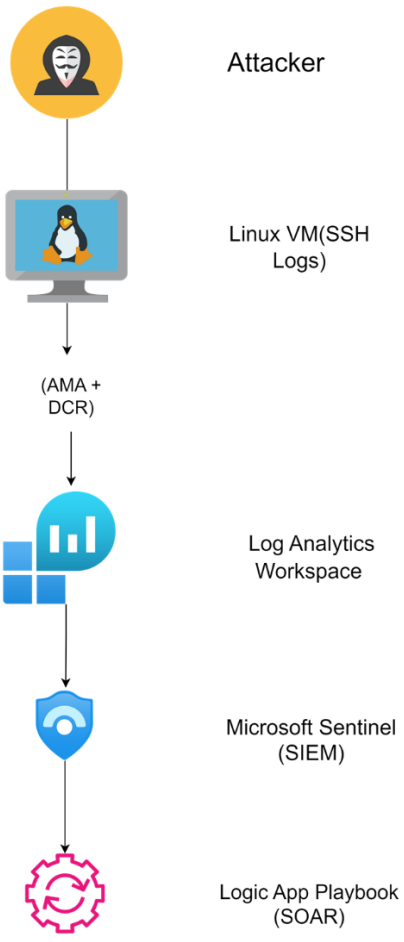
This project demonstrates a complete **SOC Level 1 style security monitoring lab** built on **Microsoft Azure + Microsoft Sentinel (SIEM/SOAR)**.

The lab covers the **full security lifecycle**:

Log Collection → Detection Engineering → Incident Investigation → Automation (SOAR) → Lab Preservation (IaaC)

It is designed as a **hands-on, portfolio-ready project** for SOC Analyst fresher roles.

Lab Architecture



Core Components

- **Azure Virtual Machine (Ubuntu Linux)**
  Acts as the monitored endpoint and log source (SSH authentication logs).
- **Network Security Group (NSG)**
  Controls inbound traffic (SSH allowed for lab access).
- **Azure Monitor Agent (AMA)**
  Installed on the VM to collect telemetry.
- **Data Collection Rule (DCR)**
  Defines what logs are collected (Linux Syslog) and where they are sent.
- **Log Analytics Workspace**
  Central log storage and query engine.
- **Microsoft Sentinel (SIEM)**
  Performs detection, analytics, incidents, investigation, and automation.
- **Logic App Playbook (SOAR)**
  Automatically tags SSH brute-force incidents.
- **Sentinel Training Solution**
  Provides simulated incidents (e.g., Solorigate) for investigation practice.

🔐 Security Use Cases Implemented

- SSH Brute-Force Detection (custom KQL rule)
- Entity Mapping (IP, Account, Host)
- Incident Creation & Correlation
- Investigation of:
  - Brute-force attempts
  - Disabled account sign-in abuse
  - Solorigate (SolarWinds) network beacon IOC
- Automated Incident Tagging using Playbook
- Detection Debugging (query tuning based on real logs)

🧪 Phases Covered in This Lab

Phase	Description
Phase 1	Azure VM deployment (Linux)
Phase 2	Microsoft Sentinel onboarding
Phase 3	Network Security Group hardening
Phase 4	AMA + Data Collection Rule setup
Phase 5	Sentinel Training Solution & analytics
Phase 6	Incident investigation (brute force & Solorigate)
Phase 7	Detection debugging & log validation
Phase 8	Playbook automation (SOAR)
Phase 9	Lab export & preservation (Cloud Shell JSON export)

🧠 Key SOC Concepts Demonstrated

- Log pipeline design (AMA + DCR)
- KQL-based detection engineering
- Entity mapping for investigations
- MITRE ATT&CK alignment
- Alert → Incident → Investigation workflow
- SOAR automation using Logic Apps
- Cloud-as-Code (CaaS) mindset via ARM JSON export

🛠️ Tools & Technologies

- Microsoft Azure
- Microsoft Sentinel (SIEM)
- Azure Monitor Agent (AMA)
- Log Analytics Workspace
- KQL (Kusto Query Language)
- Azure Logic Apps (SOAR)
- Azure Cloud Shell
- ARM Template (JSON Export)

🎯 Learning Outcomes

By completing this lab, you demonstrate:

- Practical SOC Level 1 skills
- Real detection engineering (not just theory)
- Hands-on Microsoft Sentinel usage
- End-to-end incident handling
- Basic SOAR automation
- Cloud lab preservation for reproducibility

This project can be showcased in:

- GitHub portfolio
- Resume (SOC / Cloud Security projects)
- Interviews (practical discussion points)

⚠️ Important Notes

- This is a **learning lab**, not a production-hardened environment.
- SSH is intentionally exposed for testing detections.
- Some Azure resources (extensions, internal sub-resources) are not fully exportable via ARM template. This is expected Azure behaviour.

📜 License / Usage

- This project is intended for **educational and portfolio use**.
- Do not use this setup as-is in production environments.