Project Title: Cloud Threat Intelligence: Real-Time Attack Analysis with Microsoft Sentinel

Table of Contents

1. Executive Summary

This report details the design, implementation, and operational analysis of a **Cloud Threat Intelligence & Deception Platform** leveraging Microsoft Azure and Microsoft Sentinel. The core of the project involved configuring an Azure Virtual Machine as a controlled honeypot to attract and capture live cyber-attack data. This data was then seamlessly ingested into a **Microsoft Sentinel SIEM (Security Information and Event Management) solution** for real-time analysis, monitoring, and visualization. The project successfully demonstrated advanced capabilities in cloud security architecture, log management, KQL-driven threat hunting, and the transformation of raw attack data into actionable threat intelligence, providing a comprehensive understanding of current attack methodologies and origins.

2. Introduction

2.1 Purpose of Report

The purpose of this report is to document the methodologies, technical configurations, and analytical outcomes of the Cloud Threat Intelligence & Deception Platform project. It serves as a comprehensive record of practical experience gained in deploying a cloud-based honeypot, integrating with a leading SIEM solution (Microsoft Sentinel), and performing real-time analysis of cyber-attack data.

2.2 Project Scope & Objectives

The project focused on the following key objectives:

- **Cloud Honeypot Deployment:** To provision and configure an Azure Virtual Machine as an observable honeypot to attract and capture inbound malicious network traffic.
- **Robust Log Ingestion:** To establish a scalable mechanism for collecting comprehensive security event logs from the honeypot.
- **SIEM Integration & Configuration:** To integrate the log repository with Microsoft Sentinel for advanced security analytics, monitoring, and threat prioritization.
- **Threat Analysis & Visualization:** To apply Kusto Query Language (KQL) for in-depth data analysis and to visualize attack patterns, including geographical origins and frequency.

2.3 Professional Guidelines & Requirements

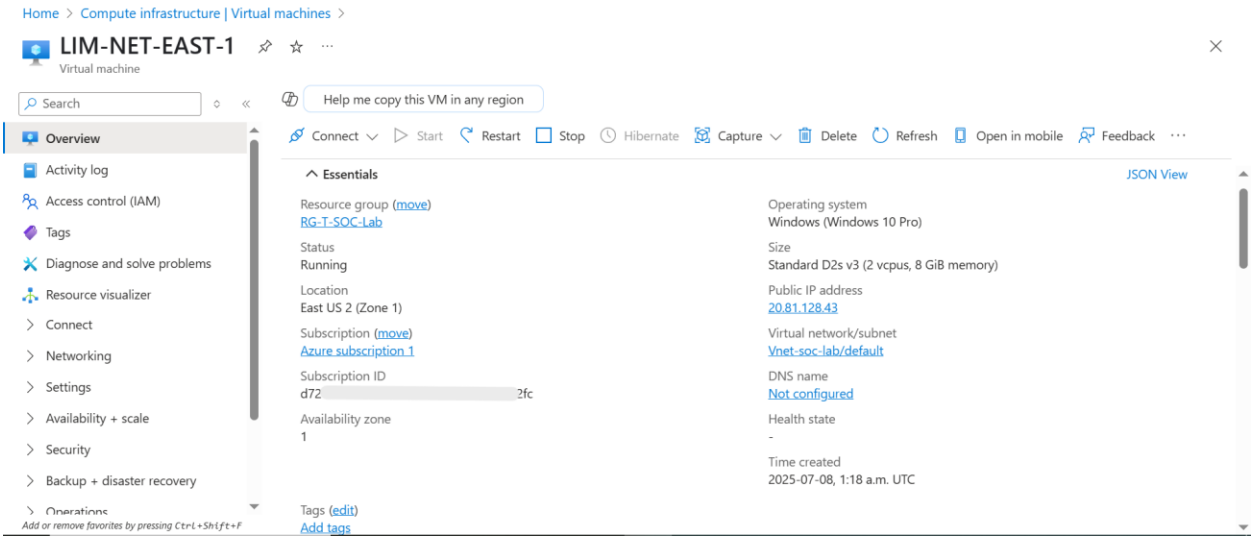
All project activities adhered to established cybersecurity best practices and professional guidelines, including:

- **Controlled Environment:** All honeypot activities were strictly confined to an isolated Azure Virtual Network environment to prevent any unintended impact on external systems or production networks.
- **Ethical Conduct:** The project focused solely on passive observation and logging of inbound attack attempts to a designated decoy, without engaging in any active exploitation or unauthorized interaction with external entities.
- **Data Integrity & Privacy:** All collected data was handled with appropriate security measures, focusing exclusively on attack metadata rather than personal identifiable information.
- **Documentation & Transparency:** All architectural designs, configurations, and analytical processes were meticulously documented to ensure clarity, reproducibility, and maintainability.
- **Scalability & Efficiency:** Design choices for log ingestion and SIEM integration prioritized scalability and efficient resource utilization inherent to cloud platforms.

3. Architectural Design & Honeypot Deployment

3.1 Azure Resource Provisioning

The project infrastructure was systematically provisioned within a dedicated **Azure Subscription**. This began with the creation of a **Resource Group** to logically encapsulate all project components. Within this Resource Group, a **Virtual Network (VNet)** was established, providing an isolated and segmented network environment. A **Windows Virtual Machine (VM)** was then provisioned within this VNet, complete with a dedicated **VM Account** for secure administrative access.



3.2 Honeypot Configuration & Network Visibility

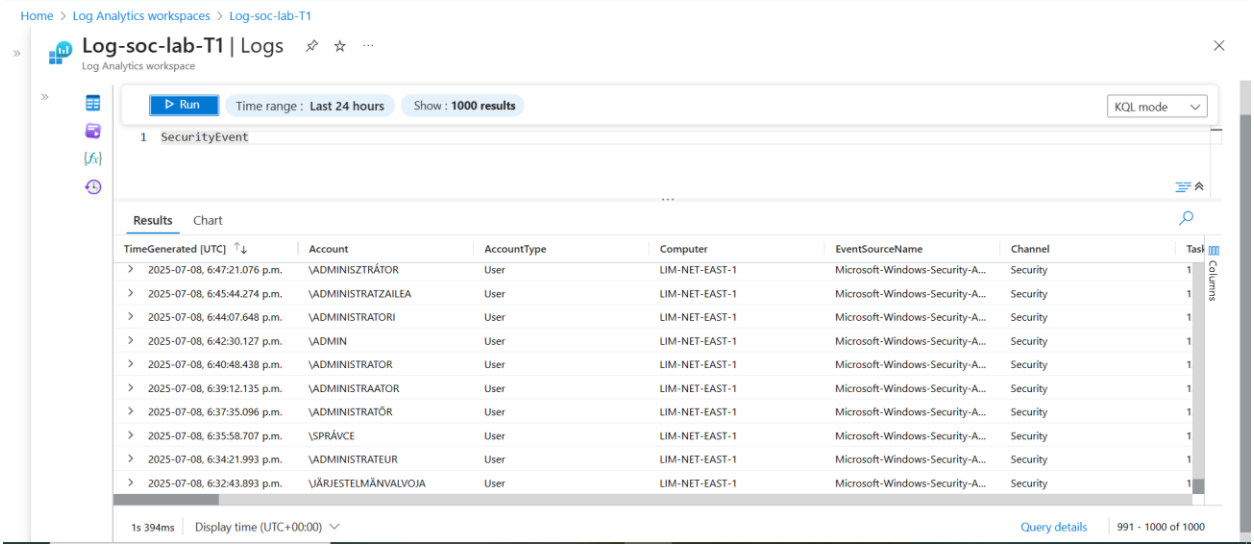
To serve its purpose as a honeypot, the deployed Windows VM was deliberately configured for maximum network visibility to attract reconnaissance and attack attempts. This involved systematically **disabling the private, public, and domain profiles of the Windows Defender Firewall** directly on the VM. This configuration effectively removed the primary host-based network barrier, making the VM overtly accessible from the internet. The success of this configuration was verified by successfully **pinging the VM's public IP address from a normal, external host computer**, confirming its network reachability.

4. Log Ingestion & Microsoft Sentinel Integration

Effective log collection and seamless integration with a SIEM solution were critical to transforming raw attack traffic into actionable security intelligence.

4.1 Log Analytics Workspace as Central Repository

A dedicated **Azure Log Analytics Workspace** was established. This workspace served as the secure, scalable, and centralized repository for all security event data generated by the honeypot VM. It forms the backbone for data storage and retrieval for subsequent analysis.



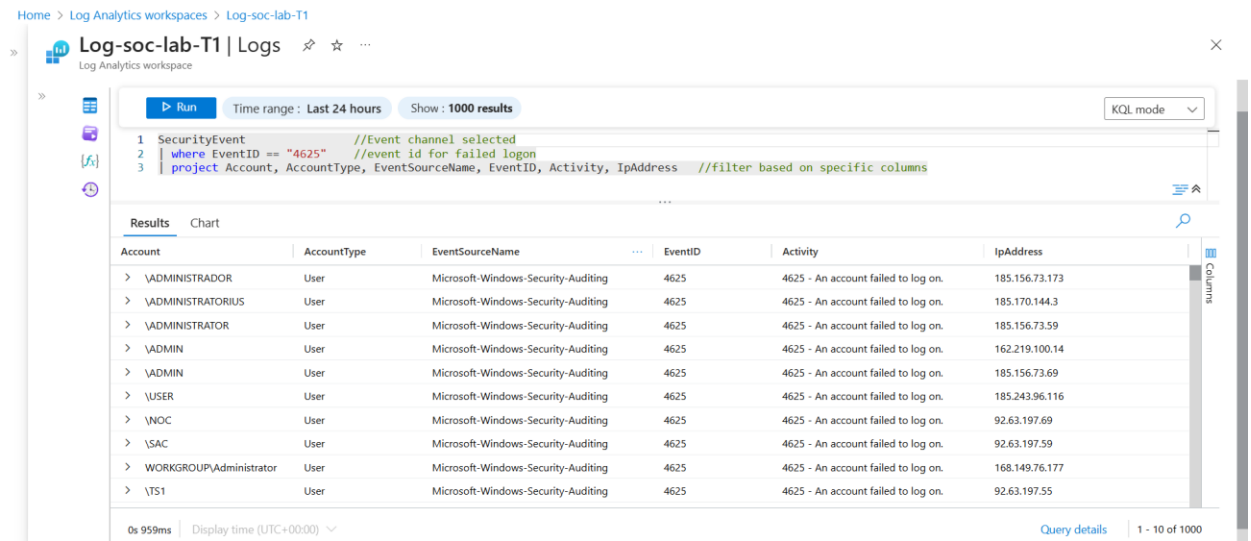
Azure Log Analytics Workspace

4.2 Microsoft Sentinel SIEM Linkage

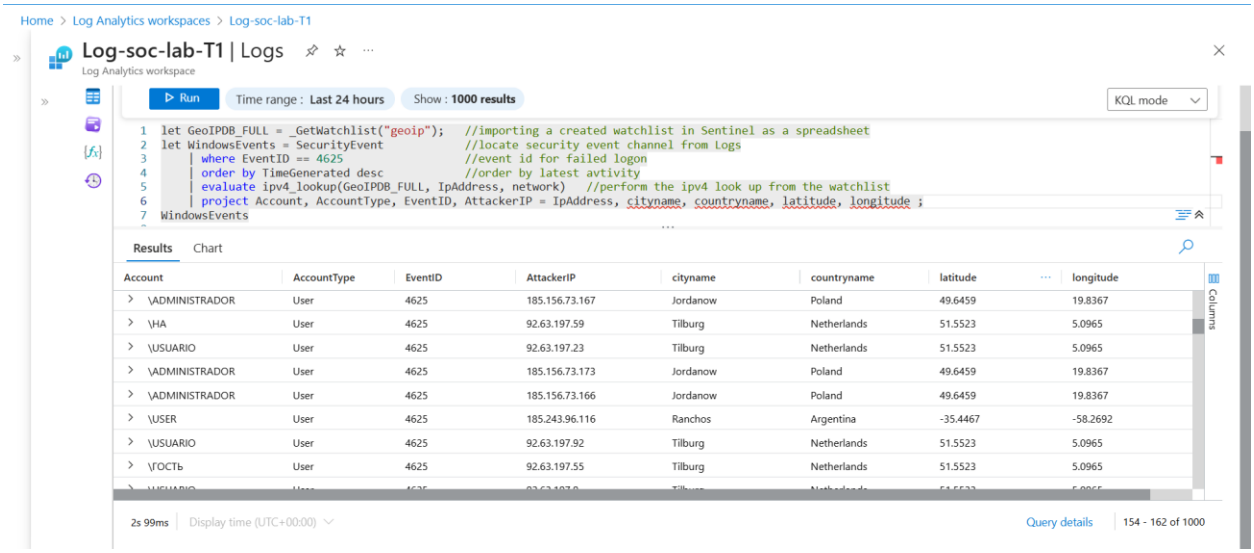
The Log Analytics Workspace was meticulously linked to **Microsoft Sentinel**, establishing it as the primary cloud-native **Security Information and Event Management (SIEM) tool** for this project. This crucial linkage enabled Sentinel to ingest, correlate, and analyze the vast streams of security event data from the honeypot.

4.3 Windows Security Event Collection & Data Collection Rule (DCR)

To ensure comprehensive capture of attack-related activities, the Windows honeypot VM was specifically configured to collect and forward its **Windows Security Event logs**. This was achieved through the implementation of a custom **Data Collection Rule (DCR-Windows)**. This DCR precisely defined which specific security events (e.g., failed login attempts, network connection attempts) were to be streamed from the VM directly to the Log Analytics Workspace, optimizing data ingestion for relevance and efficiency.



Windows Security Event logs



Windows Security Event logs with geographical location lookup from watchlist

5. Threat Monitoring, Analysis & Visualization

With a continuous flow of security logs into Microsoft Sentinel, the platform facilitated real-time threat intelligence gathering, in-depth analysis, and intuitive visualization.

5.1 Kusto Query Language (KQL) for Data Analysis

Kusto Query Language (KQL) was extensively utilized within Microsoft Sentinel's analytics capabilities. Complex KQL queries were developed to filter, parse, and analyze the massive volumes of ingested logs. These queries enabled the identification of specific attack patterns, common reconnaissance techniques, and the real-time prioritization of threats based on their characteristics and frequency.

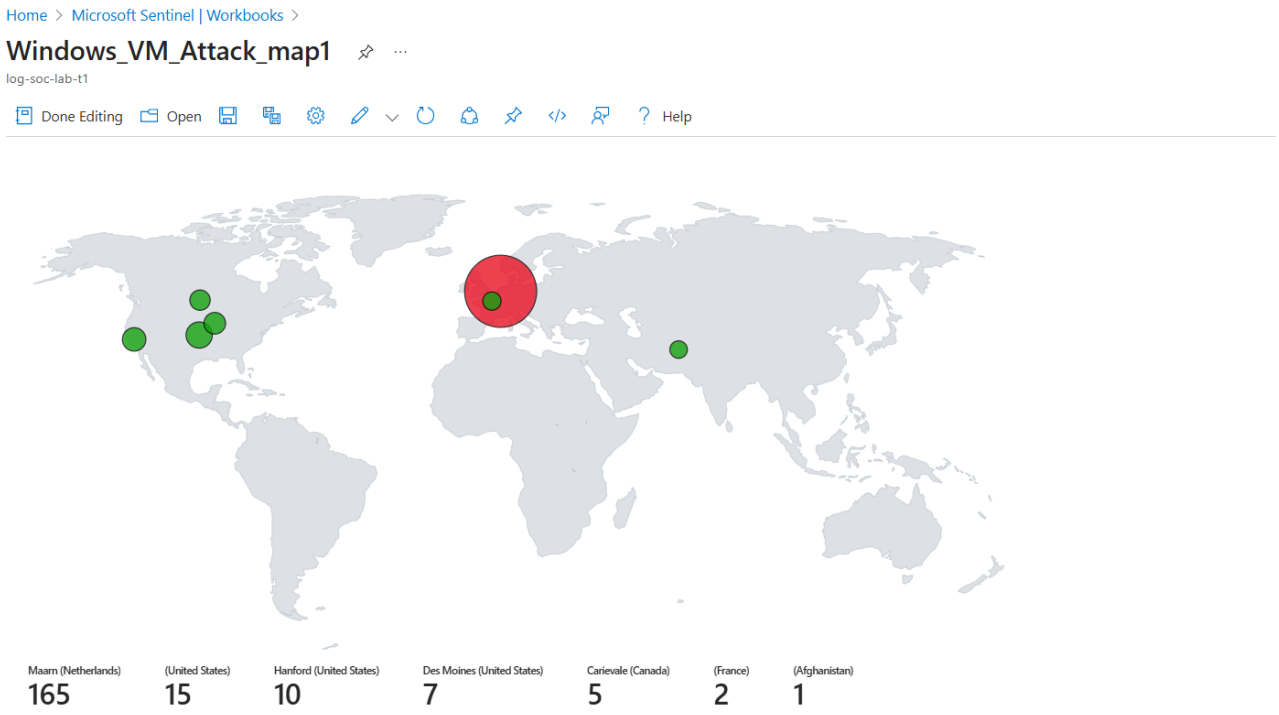
5.2 IP Geolocation & Attack Mapping

Source IP addresses from the captured attack logs were enriched through a crucial data correlation step. An **external CSV file**, containing IP-to-longitude and latitude mappings, was leveraged to geolocate the origin of each attack attempt. This provided geographical context to the raw log data.

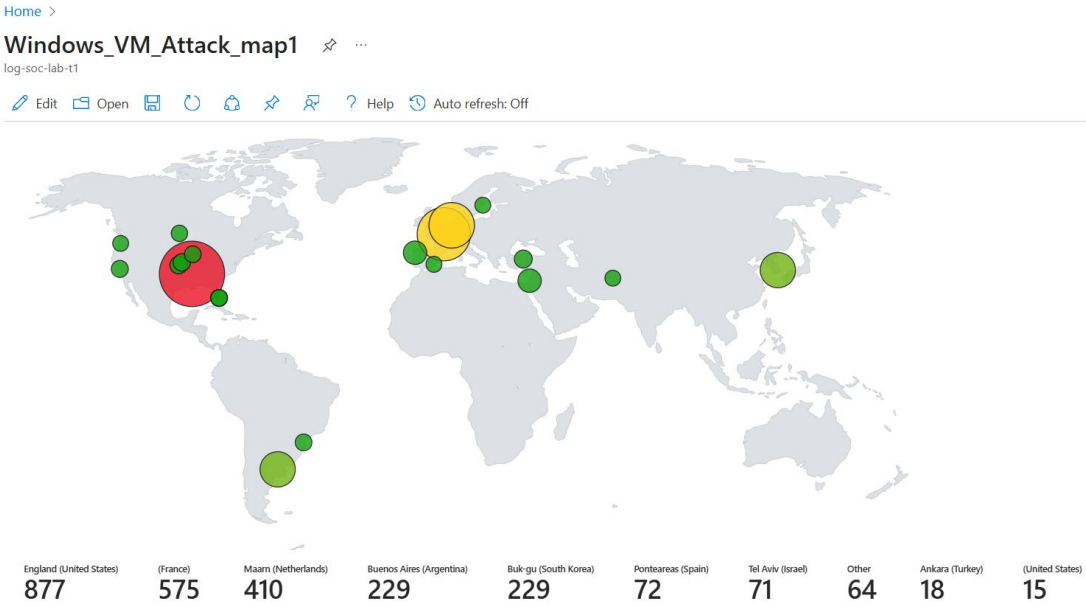
5.3 Attack Data Visualization

Microsoft Sentinel's robust visualization features, particularly its **Workbooks**, were employed to transform complex log data into intuitive and actionable threat intelligence. Dashboards were created to visually represent:

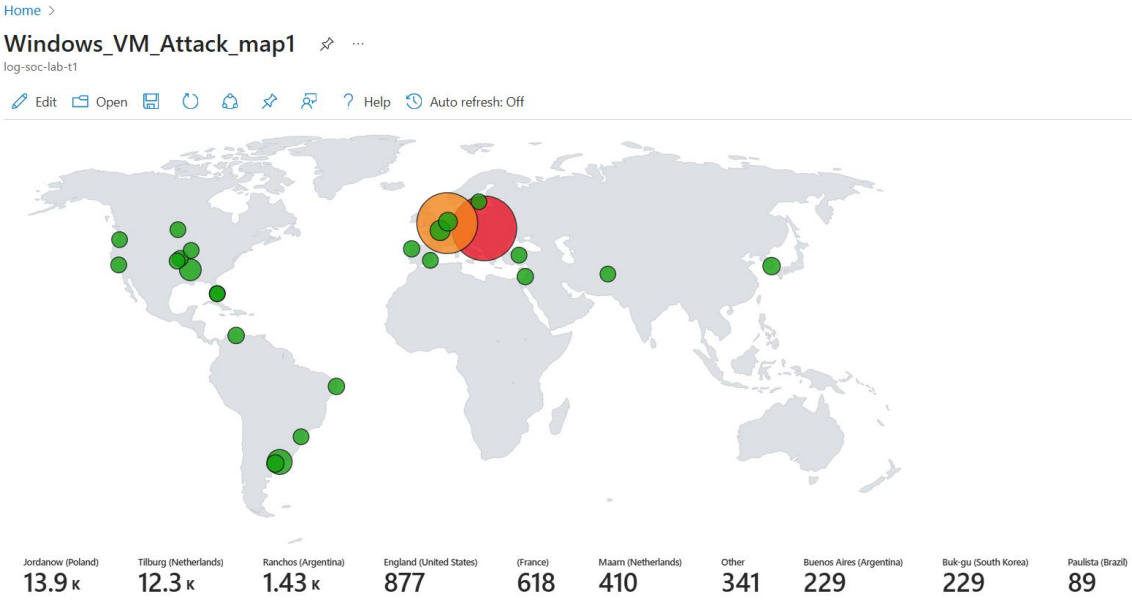
- **Geographical Distribution:** The global origins of attack attempts, highlighting regions with higher malicious activity.
- **Frequency Trends:** Attack volume over time, identifying peak activity periods.
- **Attack Types:** Common attack vectors and targeted ports.



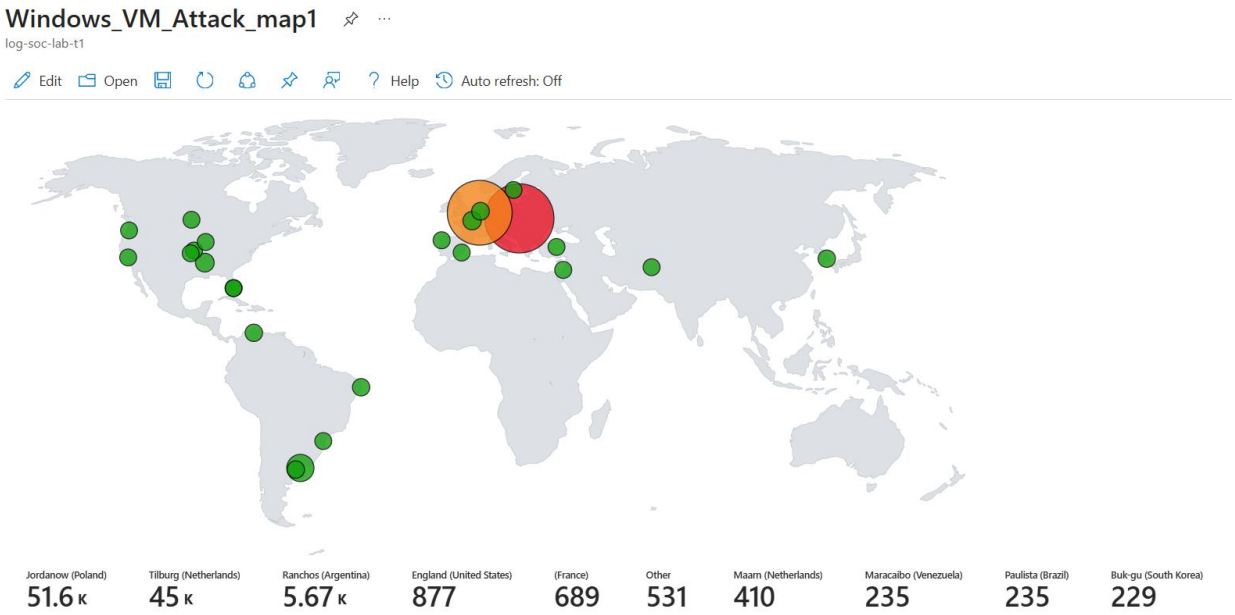
Overall Map Scan 1



Overall Map Scan 2



Overall Map Scan 3



Overall Map Scan 4

6. Project Outcomes & Demonstrated Capabilities

This project served as a comprehensive, hands-on demonstration of critical cybersecurity skills in a cloud environment:

- **Cloud Security Architecture:** Proficiency in designing and deploying secure (or intentionally insecure honeypot) infrastructures in Azure.
- **Deception Technology:** Practical experience in configuring and operating honeypots for threat intelligence.
- **Log Management & SIEM Operations:** Expertise in ingesting, processing, and analyzing security logs using Log Analytics and Microsoft Sentinel.
- **Threat Hunting & Analysis:** Demonstrated ability to identify, analyze, and prioritize real-time cyber-attacks using KQL.
- **Data Visualization:** Skills in transforming raw security data into actionable visual insights for threat intelligence.
- **Understanding of Attack Landscape:** Gained direct insight into common attack methodologies, tools, and geographical origins in a live environment.

7. Conclusion

The Cloud Threat Intelligence & Deception Platform project successfully achieved its objectives, providing a robust framework for monitoring and analyzing live cyber-attacks within a controlled Azure environment. By leveraging Microsoft Sentinel, the project

showcased a complete pipeline from honeypot deployment and log ingestion to advanced KQL-driven analysis and compelling data visualization. This initiative significantly enhanced practical skills in cloud security, SIEM operations, and proactive threat intelligence, serving as a strong foundation for future roles in cybersecurity analysis and defense.

8. Appendix

- Detailed KQL Queries Used for Threat Analysis
- Sample Raw Attack Log Excerpts
- Network Diagram of Azure Honeypot Deployment
- External IP Geolocation CSV File
- Screenshots of specific Sentinel Analytics Rules or Workbooks