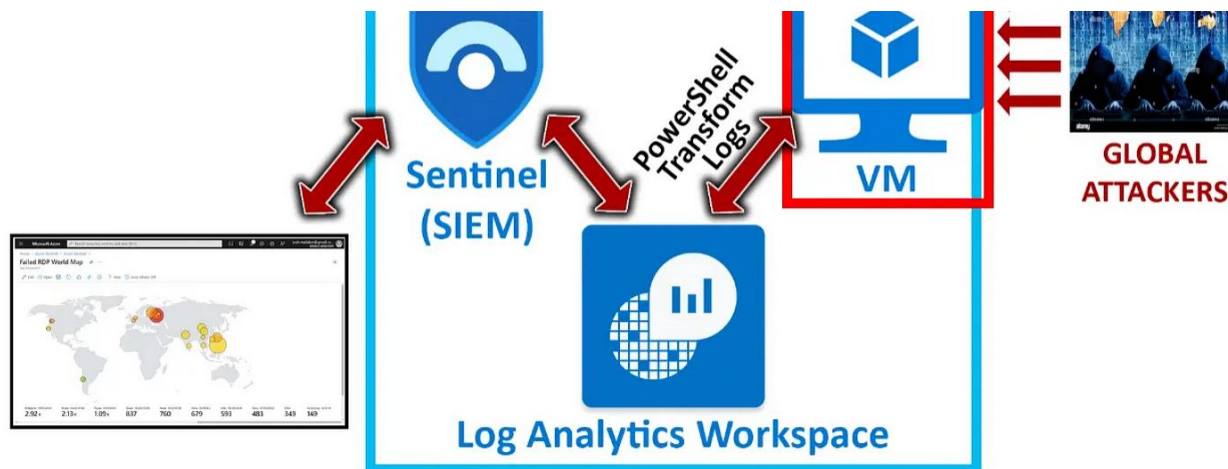


Employment of SIEM Tools to Log and Map Data Taken from an Exposed Virtual Machine

Javier A. Alvarado

05/22/2024

I. Introduction



A. Purpose of the Document

To illustrate the process of creating a virtual machine, eliminating its network defenses, and recording the login attempts from attackers all over the globe using Microsoft Azure.

B. Overview of Microsoft Sentinel

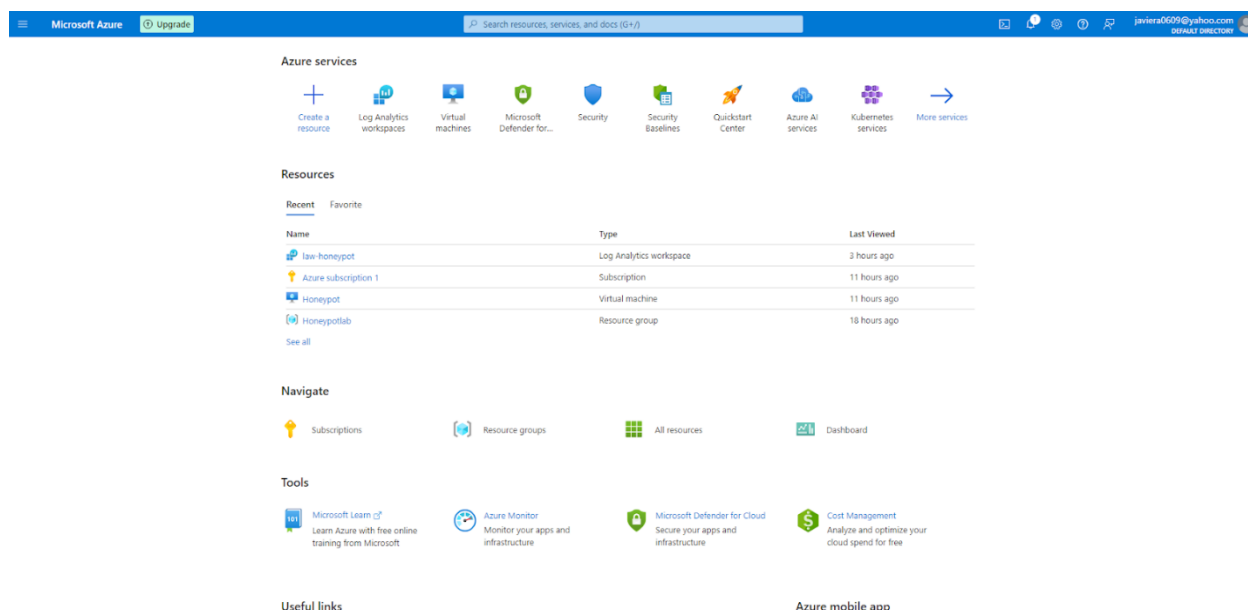
Microsoft Sentinel is a scalable, cloud-native security information and event management (SIEM) and security orchestration automated response (SOAR) solution. It leverages the power

of artificial intelligence (AI) to help detect, prevent, and respond to threats across your entire organization. By integrating with various data sources, Microsoft Sentinel provides comprehensive visibility and analysis of security data, enabling proactive threat detection and incident response.

C. Importance of Log Collection

Log collection is critical for maintaining a secure, compliant, and efficient IT environment. It supports threat detection, incident response, compliance, performance monitoring, and proactive security measures. By systematically collecting and analyzing log data, organizations can enhance their overall security posture, ensure regulatory compliance, and quickly address any issues that arise.

II. Prerequisites



A. Azure Subscription

Signing up for an Azure Subscription for access to the Virtual Machines, Servers, and Log Analytic Workspaces.

B. Azure Active Directory

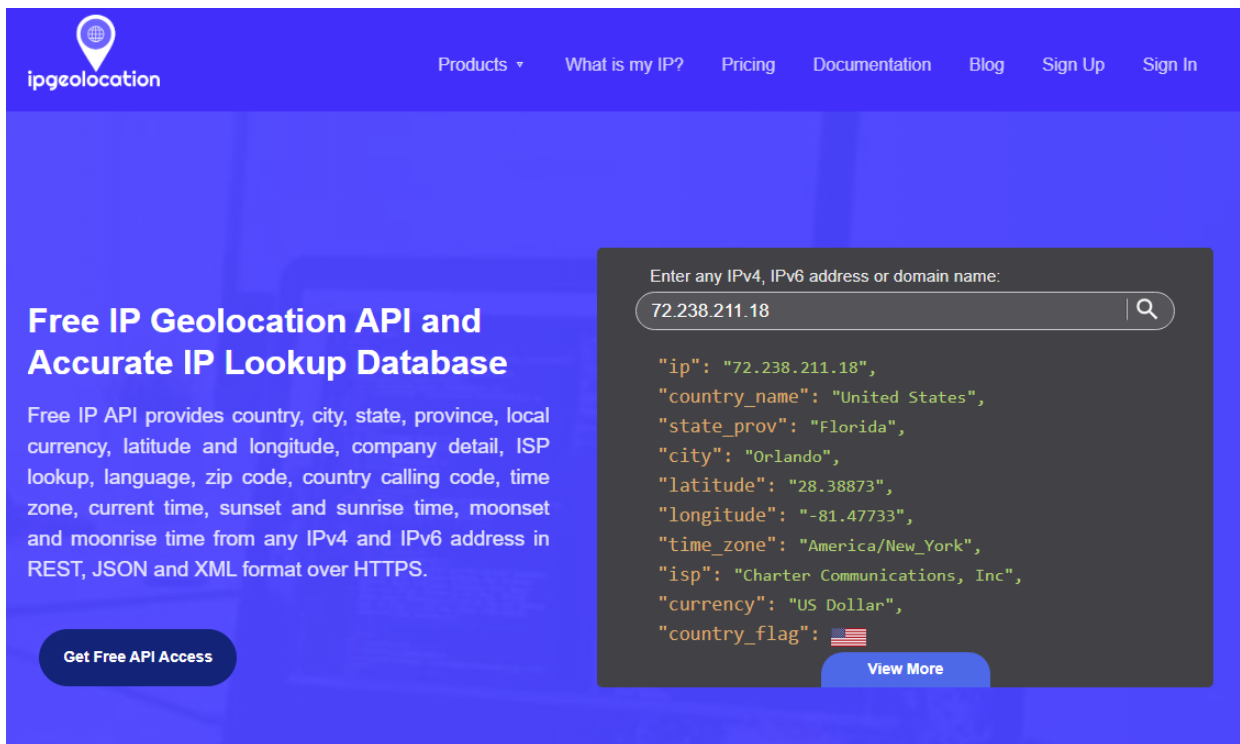
Azure Active Directory (Azure AD) is a crucial component for managing identities and access in the cloud and hybrid environments.

C. Necessary Permissions

To create and manage resources in Azure, users need appropriate permissions. These permissions are typically granted through Azure roles, which can be assigned to users, groups, or service principals.

D. IPGeolocation API

Creating an account for IPGeolocation and using the API key given to put it into the PowerShell script in order to return log data to Azure.



The screenshot displays the IPGeolocation website interface. At the top, there is a navigation bar with the logo and links for Products, What is my IP?, Pricing, Documentation, Blog, Sign Up, and Sign In. The main content area features a large heading "Free IP Geolocation API and Accurate IP Lookup Database" and a description of the API's capabilities. A search bar is prominently displayed, showing the IP address "72.238.211.18" and a search button. Below the search bar, a JSON response is shown, providing detailed location information for the entered IP address. A "Get Free API Access" button is located at the bottom left, and a "View More" button is at the bottom right of the JSON response area.

ipgeolocation

Products ▾ What is my IP? Pricing Documentation Blog Sign Up Sign In

Free IP Geolocation API and Accurate IP Lookup Database

Free IP API provides country, city, state, province, local currency, latitude and longitude, company detail, ISP lookup, language, zip code, country calling code, time zone, current time, sunset and sunrise time, moonset and moonrise time from any IPv4 and IPv6 address in REST, JSON and XML format over HTTPS.

[Get Free API Access](#)

Enter any IPv4, IPv6 address or domain name:

72.238.211.18

```
{
  "ip": "72.238.211.18",
  "country_name": "United States",
  "state_prov": "Florida",
  "city": "Orlando",
  "latitude": "28.38873",
  "longitude": "-81.47733",
  "time_zone": "America/New_York",
  "isp": "Charter Communications, Inc",
  "currency": "US Dollar",
  "country_flag": "🇺🇸"
}
```

[View More](#)

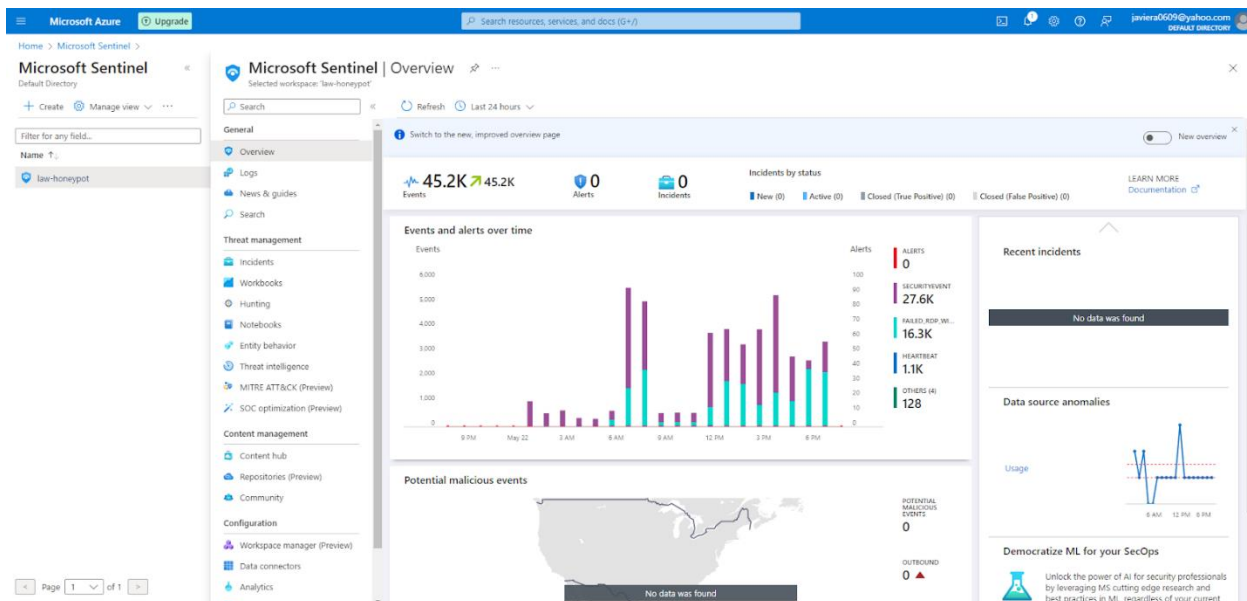
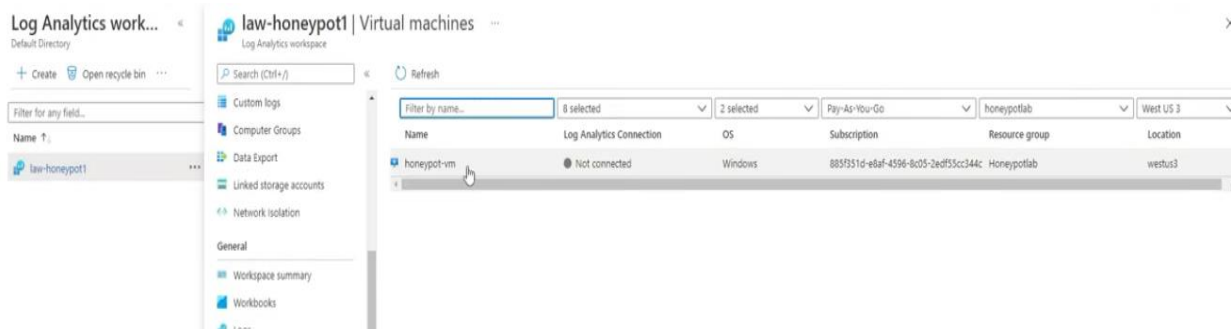
III. Setting Up Microsoft Sentinel (Azure Sentinel)

A. Creating a Log Analytics Workspace

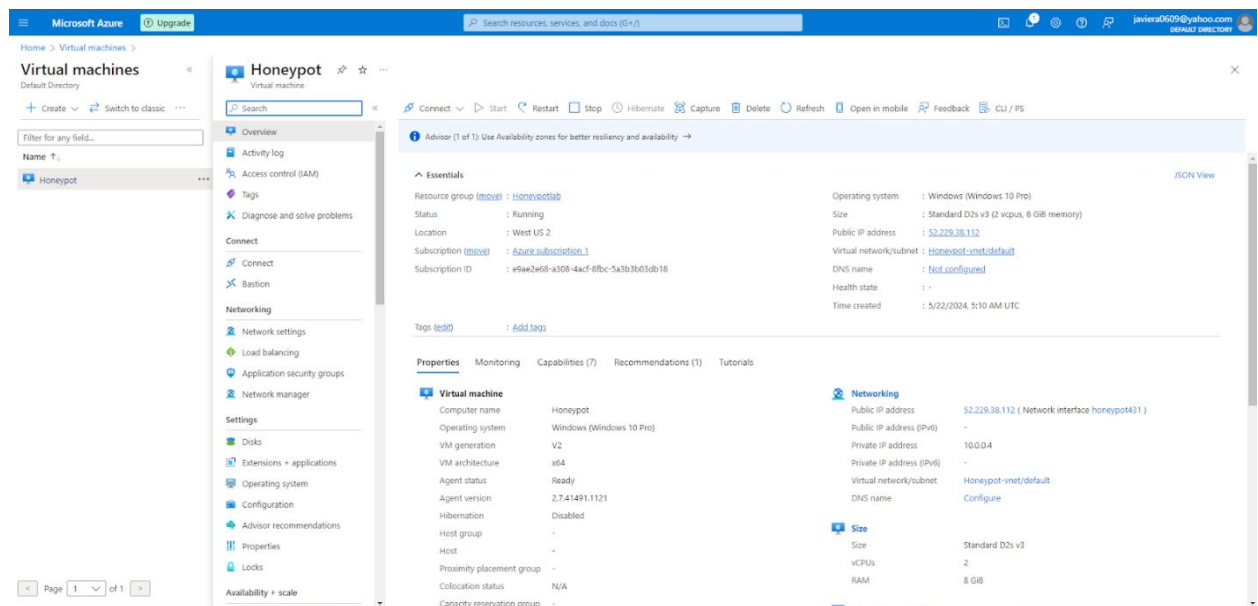
Here we will create a Log Analytics Workspace where we will later create a query script to parse the data we receive from the virtual machine's event logs and organize it to plot it on a geographical map using the latitude and longitude of the attacker's IP location.

B. Adding Microsoft Sentinel to the Log Analytics Workspace

After creating the workspace, we will add Microsoft Sentinel to the workspace so we can later manage the incoming data.

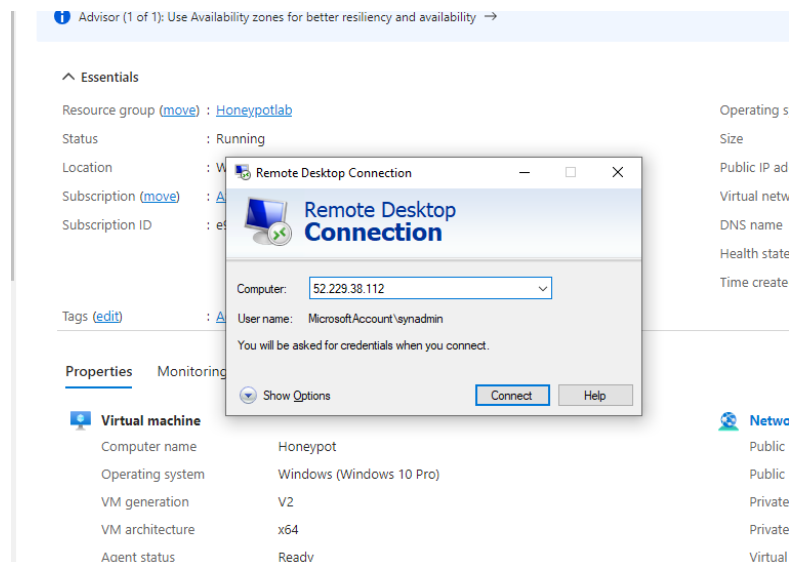


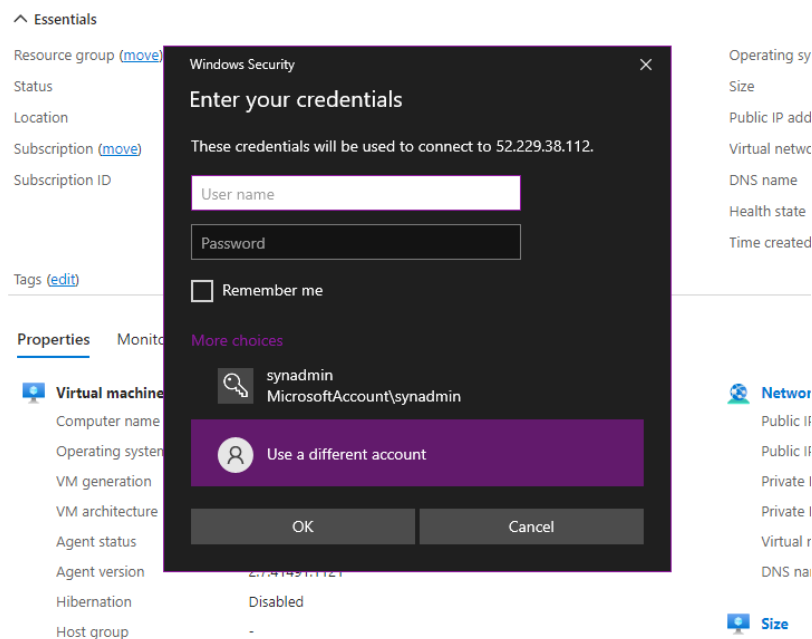
IV. Creating a Virtual Machine Exposed to the Internet



A. Creating the Virtual Machine

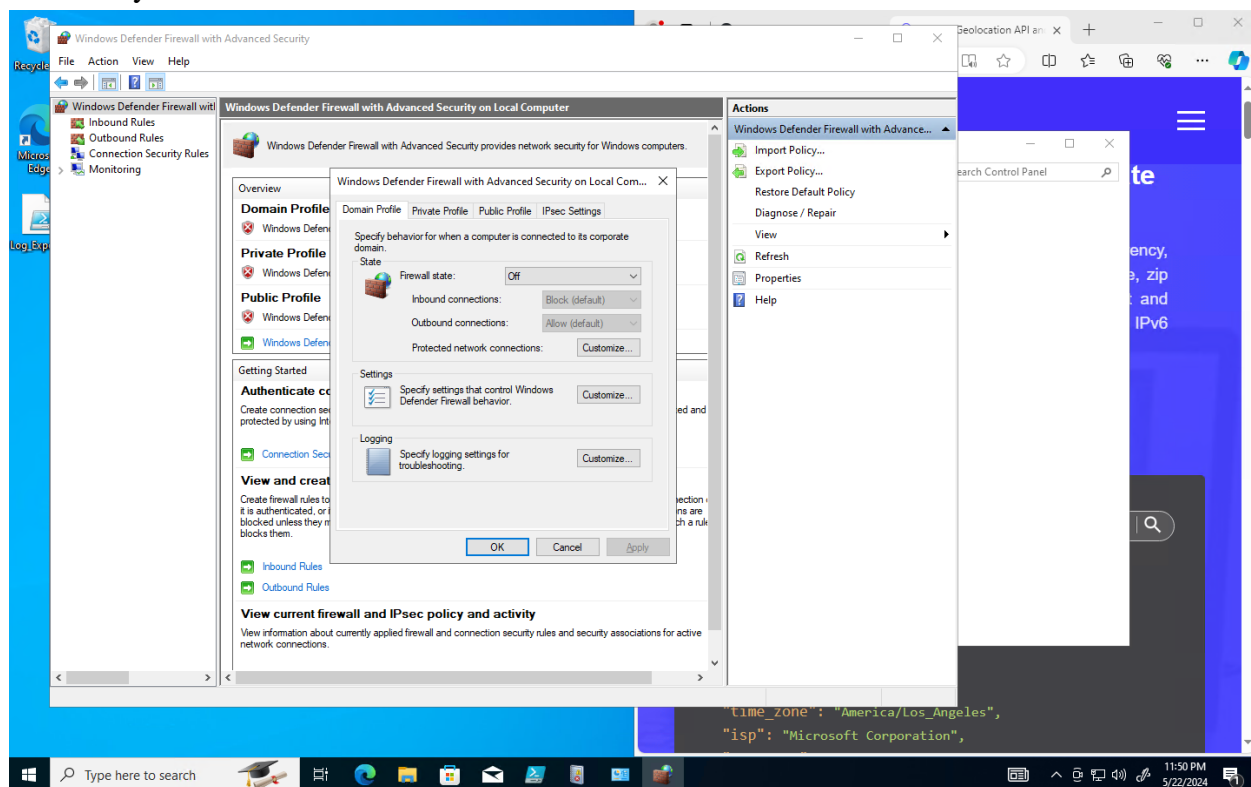
Illustrated above are the details related to the virtual machine we created on an Azure server. We can see here that the main details that are critical to the performance of the project are shown as the resource group with the name we created (Honeypotlab) along with the public IP address of 52.229.38.112 and the operating system of Windows 10 Pro.

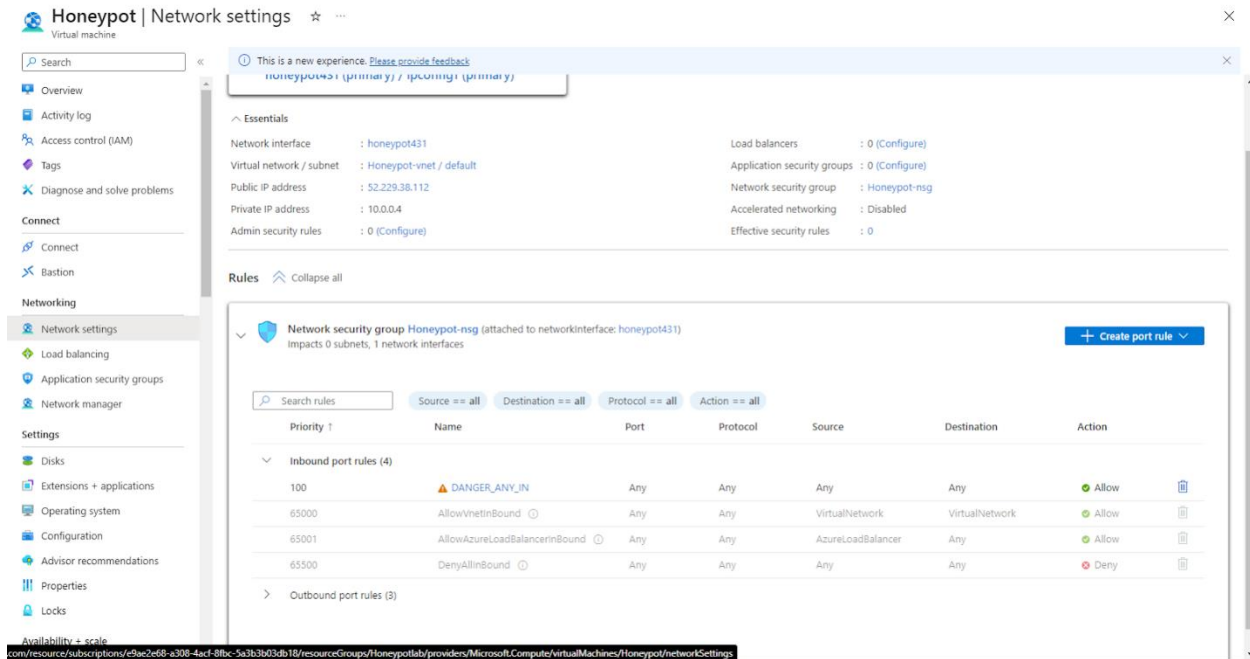




B. Logging into the Virtual Machine and creating a firewall rule to allow any inbound connections

After having logged into the Virtual Machine successfully, we proceed with disabling the firewall state for the domain, private, and public profile along with creating an inbound rule to allow any remote connections to the Virtual Machine.





V. Installing Log Collection Agents on the Virtual Machine

A. Using a PowerShell script in Windows PowerShell ISE to collect and organize data

After having opened up access to the machine we create a PowerShell script in Windows PowerShell ISE to filter failed RDP events from Windows Event Viewer. This function creates a bunch of sample log files that will be used to train the extract feature in the Log Analytics workspace. We can avoid including these fake records on our map by filtering out all logs with a destination host of "samplehost". We then save the file with the results of the script to our desktop as a .txt file. Note that this .txt file will constantly be updated as more and more failed attempts to log in to the Virtual Machine have been performed.


Administrator: Windows PowerShell ISE

```
File Edit View Tools Debug Add-ons Help
Log_Exporter.ps1 X
1 Get API key from here: https://ipgeo.location.io/
2 $PL_KEY = "0e2e1a6f1025427d8a2df27475d68c6f"
3 $LOGFILE_PATH = "C:\ProgramData\S\SLOGFILE_NAME"
4
5 This filter will be used to filter failed RDP events from Windows Event Viewer
6 $MLFilter = @"
7 {
8   <Query Id="0" Path="Security">
9     <Select Path="Security">
10       <System [EventID= 4625]">
11     </Select>
12   </Query>
13 }
14 {
15   <Query Id="0" Path="Security">
16     <Select Path="Security">
17       <System [EventID= 4625]">
18     </Select>
19   </Query>
20 }
21 }
22
23 This function creates a bunch of sample log files that will be used to train the
24 Extract feature in Log Analytics workspace. If you don't have enough log files to
25 "train" it, it will fail to extract certain fields for some reason --.
26 We can avoid including these fake records on our map by filtering out all logs with
27 a destination host of "samplehost"
28
29 function write-Sample-Log {
30   [latitude:47.91542, longitude:-120.60306, destinationhost:samplehost, username:fakex
31   "latitue:22.90906, longitude:-47.06455, destinationhost:samplehost, username:ADMINI
32   "latitue:52.37022, longitude:4.89517, destinationhost:samplehost, username:CNVDER,
33   "latitue:40.71455, longitude:-74.00714, destinationhost:samplehost, username:ADMINI
34   "latitue:33.99762, longitude:6.84737, destinationhost:samplehost, username:AZUREUS
35   "latitue:5.32558, longitude:100.28595, destinationhost:samplehost, username:Test,
36   "latitue:11.95722, longitude:126.84926, destinationhost:samplehost, username:AZUREUS
37   "latitue:55.87925, longitude:37.54691, destinationhost:samplehost, username:Test,
38   "latitue:52.37018, longitude:4.87234, destinationhost:samplehost, username:AZUREUS
39 }
40
41 2:19:27:459
42 Invoke-WebRequest: The remote server returned an error: (429).
43 At C:\Users\syndamin\Desktop\Log_Exporter.ps1:115 Char:29
44 + ... $Response = Invoke-WebRequest -Uri $SAP1_ENDPOINT
45 + ~~~~~
46 + CategoryInfo          : InvalidOperation: (System.Net.HttpWebResponse:HttpWebResponse)
47 + FullyQualifiedErrorId : WebCmdletWebResponseException,Microsoft.PowerShell.Commands
48
49 [latitude:21.01941, longitude:105.80903, destinationhost:Honeypot, username:administrator, sou
50 2:19:27:460
```

Running script / selection. Press Ctrl+Break to stop. Press Ctrl+B to break into debugger.

Sentinel-Lab/Custom_Sec...Free IP Geolocation API an...+Free IP Geolocation API an...x

<>https://ipgeolocation.ioA🔍🌐🏠📄⋮👤

**ipgeolocation**

☰

Free IP Geolocation API and Accurate IP Lookup Database

Free IP API provides country, city, state, province, local currency, latitude and longitude, company detail, ISP lookup, language, zip code, country calling code, time zone, current time, sunset and sunrise time, moonset and moonrise time from any IPv4 and IPv6 address in REST, JSON and XML format over HTTPS.

[Get Free API Access](#)

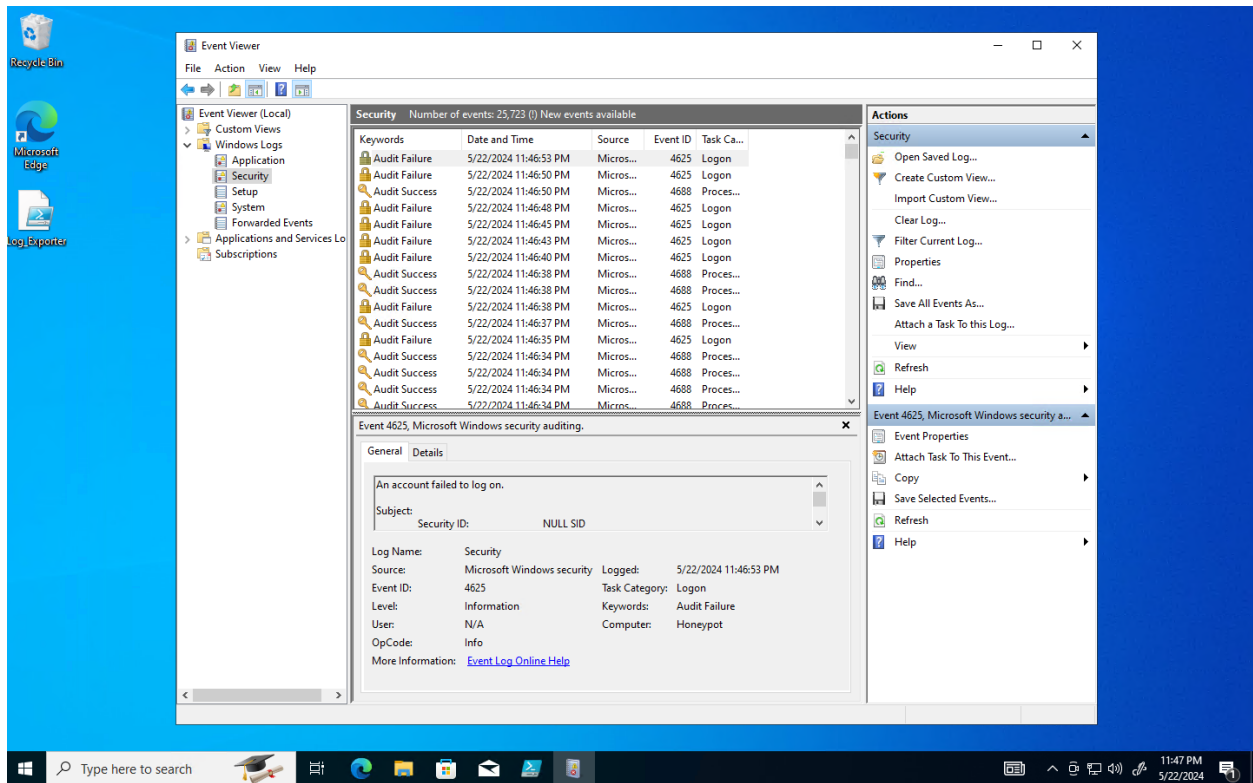
Enter any IPv4, IPv6 address or domain name:

🔍

```
"ip": "52.229.38.112",  
"country_name": "United States",  
"state_prov": "Washington",  
"city": "Quincy",  
"latitude": "47.23430",  
"longitude": "-119.85255",  
"time_zone": "America/Los_Angeles",  
"isp": "Microsoft Corporation",
```


B. Reviewing Event Viewer to see the data that we are going to be collecting from the logs

After having created the PowerShell Script and ran it, we can take a look at the Event Viewer to see what kind of logs the script is going to process. For this project, we are only going to look at events with the Event ID of 4625 which correlates to an account that has failed to log on.



C. Creating a custom log in Log Analytics Workspace to utilize the .txt file we created in the Virtual Machine

The screenshot displays the Microsoft Azure portal interface for a Log Analytics workspace named 'law-honeypot'. The left sidebar shows the navigation menu with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Logs, Settings, Tables, Agents, Usage and estimated costs, Data export, Network isolation, Linked storage accounts, Properties, Locks, Classic, Legacy agents management, Legacy activity log connector, Legacy storage account logs, Legacy computer groups, Legacy solutions, System center, and Workspace summary. The main pane shows a 'New Query' editor with a Kusto query designed to parse raw data and project specific fields. The query is as follows:

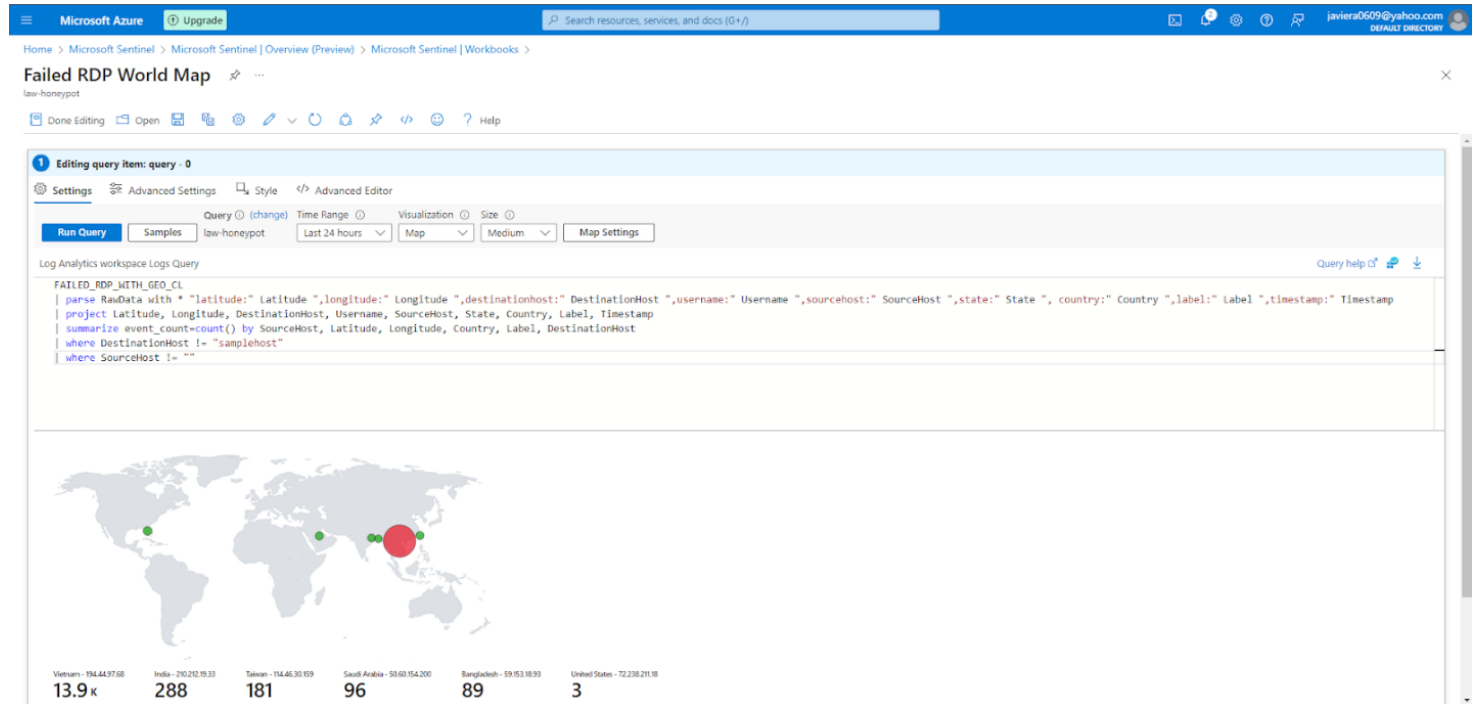
```
1 FAILED_RDP_WITH_GEO_CL
2 | parse RawData with " "latitude:" Latitude ","longitude:" Longitude ","destinationhost:" DestinationHost ","username:" Username ","sourcehost:" SourceHost ","state:" State ","country:" Country ","label:" Label ","timestamp:" Timestamp
3 | project Latitude, Longitude, DestinationHost, Username, SourceHost, State, Country, Label, Timestamp
```

Below the query editor, the 'Query history' section shows several previous queries and their results. The first query is the same as the one in the editor, with 12,986 results. The second query extends the first by adding fields for username, timestamp, latitude, and longitude, with 12,781 results. The third query summarizes the event count by source host, latitude, and longitude, with 19 results. The fourth query projects the same fields as the first, with 12,437 results. The fifth query is identical to the first, with 12,437 results.

Above is the log used to parse the raw data we collect from that .txt file and extract only the necessary information needed to plot the log-in attempts on a map. From the rawdata we are going to extract the following:

- Latitude
- Longitude
- Destination Host
- Username
- Source Host
- State
- Country
- Label
- Timestamp

VI. Plotting the data from the log into a geographical map



Map Settings

Layout Settings

Location Info using ⓘ

Latitude/Longitude

Latitude * ⓘ

Latitude

Longitude * ⓘ

Longitude

Size by ⓘ

Latitude

Aggregation for location ⓘ

Sum of values

Minimum region size ⓘ

20

Maximum region size ⓘ

70

Default region size ⓘ

10

Minimum value ⓘ

(auto) ✓

Maximum value ⓘ

(auto) ✓

Opacity of items on Map ⓘ

0.7

Color Settings

Coloring Type ⓘ

None Thresholds **Heatmap**

Color by ⓘ

Latitude

Aggregation for color ⓘ

Sum of values

Color palette

Green to Red

Apply

Save and Close

Cancel

Map Settings

Default region size ⓘ

10

Minimum value ⓘ

(auto) ✓

Maximum value ⓘ

(auto) ✓

Opacity of items on Map ⓘ

0.7

Color Settings

Coloring Type ⓘ

None Thresholds **Heatmap**

Color by ⓘ

Latitude

Aggregation for color ⓘ

Sum of values

Color palette

Green to Red

Minimum value ⓘ

(auto) ✓

Maximum value ⓘ

(auto) ✓

Metric Settings

Metric Label ⓘ

Label

Metric Value ⓘ

event_count

Create 'Others' group after ⓘ

10

Aggregate 'Others' metrics by ⓘ

Sum of values

☐ Custom formatting ⓘ

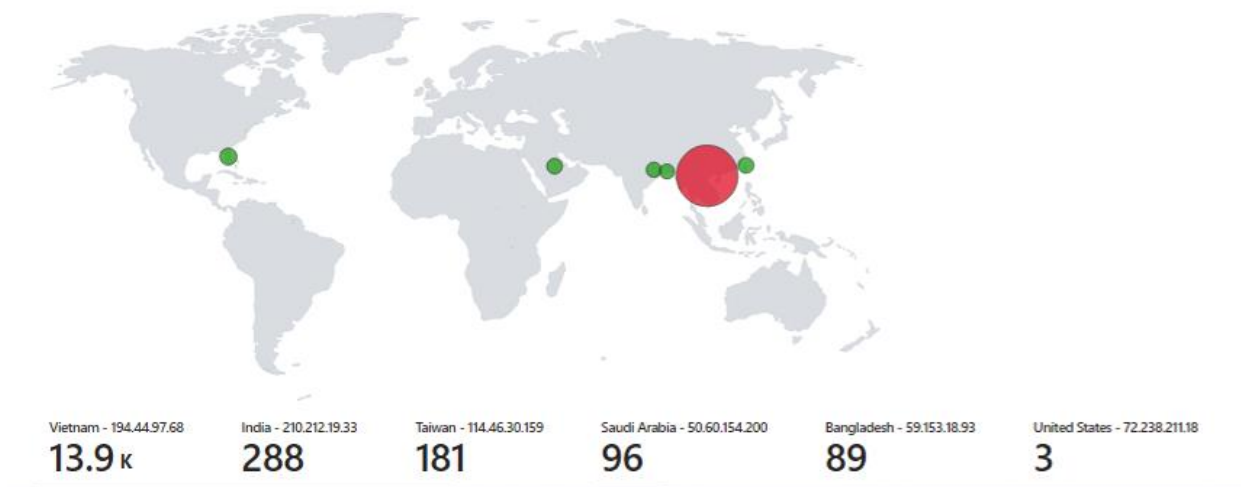
Apply

Save and Close

Cancel

A. Using the output data to configure a visual map of attempts

We can now adjust the configuration of the map to create a shaded circle around the general area of the intruder's IP and size it according to the number of events that come from that geographical location.



B. Final results of log collecting

After having opened up the virtual machine to the internet where anyone can access it due to the unrestricted firewall, the PowerShell script was able to process over 14,000 separate failed login attempts to the virtual machine and mapped their IP's geographic location using Microsoft Sentinel.

VII. Conclusion

A. Summary of Steps

Here is a listed summary of the steps taken in this project.

- Create Azure Subscription
- Create Virtual Machine
- Create Log Analytics Workspace
- Create an allow all rule in the Windows Firewall
- Enable gathering virtual machine logs in security center

- Connect Log Analytics to virtual machine
- Setup Microsoft Sentinel (Azure Sentinel)
- Log into virtual machine
- Observe Event Viewer logs in virtual machine
- Turn off Windows Firewall
- Download and input PowerShell script using the API key from IPGeolocation.io
- Run script to gather geo data from attackers
- Create custom log in Log Analytics Workspace to introduce the custom log
- Extract data
- Set up map in Microsoft Sentinel using latitude and longitude along with the associated country
- Adjust the map configuration

B. Final Thoughts

Implementing SIEM (Security Information and Event Management) monitoring is crucial, especially after setting up a project where a virtual machine is intentionally exposed to the network. By gathering data from potential attackers and plotting their IP locations on a map, organizations gain valuable insights into attack patterns and sources. This proactive approach allows for real-time threat detection, comprehensive incident analysis, and effective response strategies. Ultimately, SIEM monitoring enhances an organization's security posture, ensuring robust protection against evolving cyber threats.

VII. References

A. Documentation References

“Free IP Geolocation API and Accurate IP Geolocation Database.” *IPGeolocation API*, ipgeolocation.io/. Accessed 23 May 2024.

“Siem Tutorial for Beginners | Azure Sentinel Tutorial Map with Live Cyber Attacks!” *YouTube*, YouTube, 1 Nov. 2021, www.youtube.com/watch?v=RoZeVbbZ0o0.

joshmadakor1. “Sentinel-Lab/Custom_security_log_exporter.PS1 at Main · Joshmadakor1/Sentinel-Lab.” *GitHub*, github.com/joshmadakor1/Sentinel-Lab/blob/main/Custom_Security_Log_Exporter.ps1. Accessed 22 May 2024.

Azure, Microsoft. “Use Azure Monitor to Integrate with Siem Tools.” *Microsoft Azure Blog*, 28 July 2023, azure.microsoft.com/en-us/blog/use-azure-monitor-to-integrate-with-siem-tools/.

Akindehin, Ibitola. “A Beginner’s Guide to Setting up a Siem with Microsoft Azure.” *Medium*, Medium, 6 Nov. 2023, medium.com/@ibitolabif/a-beginners-guide-to-setting-up-a-siem-with-microsoft-azure-567276c35f81.