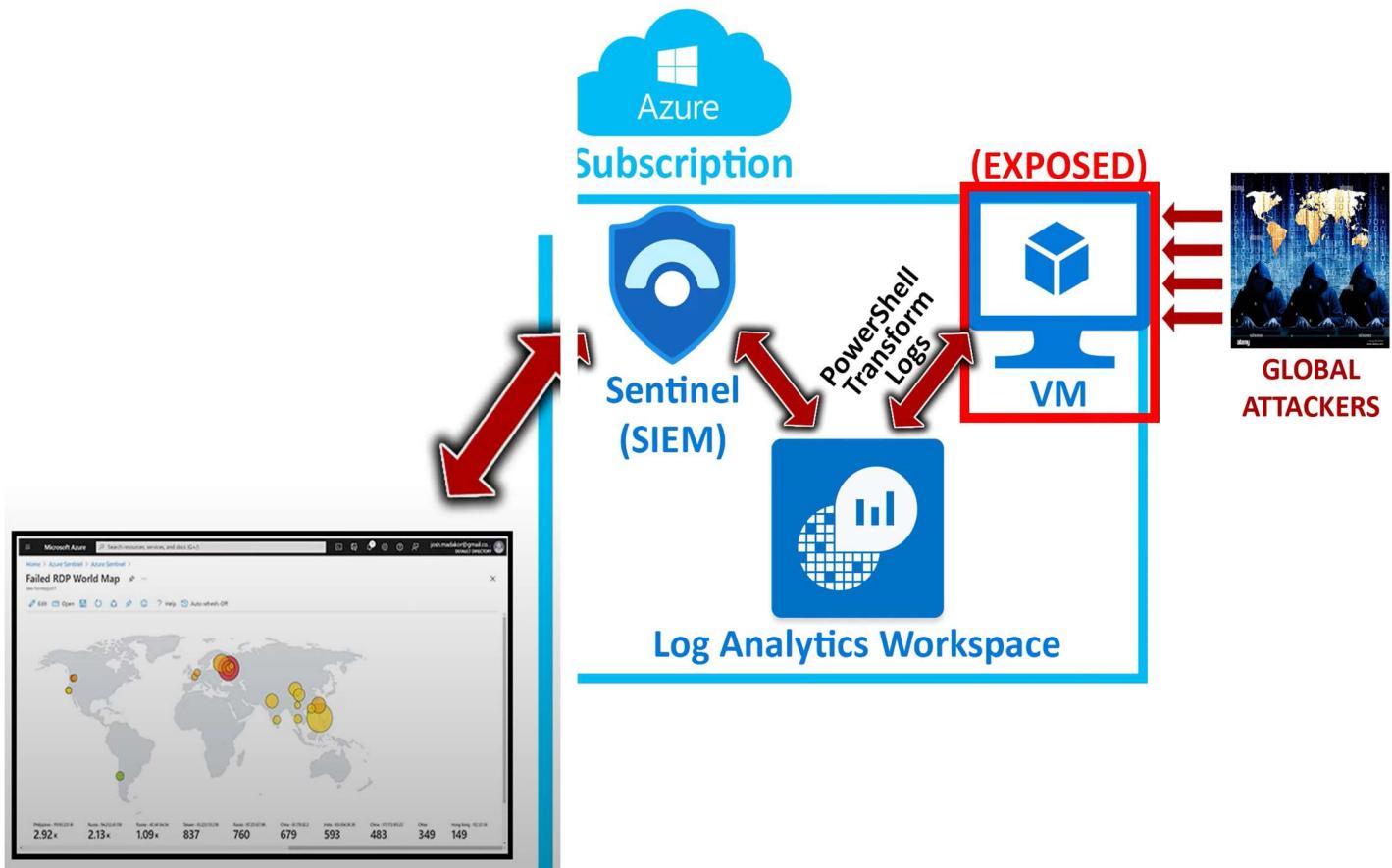


In this project I have set up the Microsoft Azure Sentinel (Cloud-based SIEM) and connected it to Honeypot created in Virtual Machine in Azure. This project has given me the opportunity to gain practical experience in the cybersecurity field.

Tasks completed in this project include:

- Used custom PowerShell script to extract metadata from Windows
- Event Viewer to be forwarded to third party API to derive the geolocation data.
- Configured Log Analytics workspace in azure to ingest custom logs from windows Virtual Machine containing geographic information (Latitude, Longitude, state/Province, and country) using third party API.
- Configured custom fields in Log Analytics workspace with the intent of mapping geo-data in Azure sentinel.
- Configure Azure Sentinel (Microsoft Cloud SIEM) workbook to display global attack data (RDP Brute Force) on world map based on physical location and magnitude of attacks.

Diagram of the project



Create a virtual machine using Virtual Machine Resource and set up the basics with the data as shown below

Microsoft Azure Search res...

Home > Virtual machines > Create a virtual machine ...

Basics Disks Networking Management Monitoring Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ✓

Resource group * ✓
[Create new](#)

Instance details

Virtual machine name * ✓

Region * ✓

Availability options ✓

Security type ✓

Image * ✓
[See all images](#) | [Configure VM generation](#)

VM architecture x64
 Arm64
⚠️ Arm64 is not supported with the selected image.

Run with Azure Spot discount

Size * ✓
[See all sizes](#)

Administrator account

Username * ✓

Password * ✓

Confirm password * ✓

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

[Review + create](#) < Previous Next : Disks >

Under the Networking tab configure the new security group and remove the existing rule and create new rule as shown below

The screenshot shows the Microsoft Azure portal interface for creating a virtual machine. On the left, there's a sidebar with 'Home', 'Virtual machines', 'Create a virtual machine', 'Create network security group', and other options like 'Inbound rules' and 'Outbound rules'. The main area is titled 'Add inbound security rule' for the 'HoneyPotVM-nsg' security group. The configuration fields include:

- Source:** Any
- Source port ranges:** *
- Destination:** Any
- Service:** Custom
- Destination port ranges:** *
- Protocol:** Any (selected)
- Action:** Allow (selected)
- Priority:** 100
- Name:** AllowAnyCustomAnyInbound
- Description:** (empty)

At the bottom, there are 'OK', 'Add', and 'Cancel' buttons.

Since this virtual machine is being used as Honeypot, it is made easily accessible to hackers which will attract them to conduct brute force attack With following inbound rules.

 Add inbound security rule

HoneyPotVM-nsg

Source ⓘ

Source port ranges * ⓘ

Destination ⓘ

Service ⓘ

Destination port ranges * ⓘ

Protocol
 Any
 TCP
 UDP
 ICMP

Action
 Allow
 Deny

Priority * ⓘ

Name *

Description

Add **Cancel**

Creating Log Analytics workspace:

It is created to ingest custom logs from virtual machine that contains geographic location

Home > Log Analytics workspaces >

Create Log Analytics workspace

Basics Tags Review + Create

i A Log Analytics workspace is the basic management unit of Azure Monitor Logs. There are specific considerations you should take when creating a new Log Analytics workspace. [Learn more](#) X

With Azure Monitor Logs you can easily store, retain, and query data collected from your monitored resources in Azure and other environments for valuable insights. A Log Analytics workspace is the logical storage unit where your log data is collected and stored.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Visual Studio Enterprise Subscription – MPN

Resource group * ⓘ

HoneypotLab

[Create new](#)

Instance details

Name * ⓘ

WorkspaceHoneyPot

Region * ⓘ

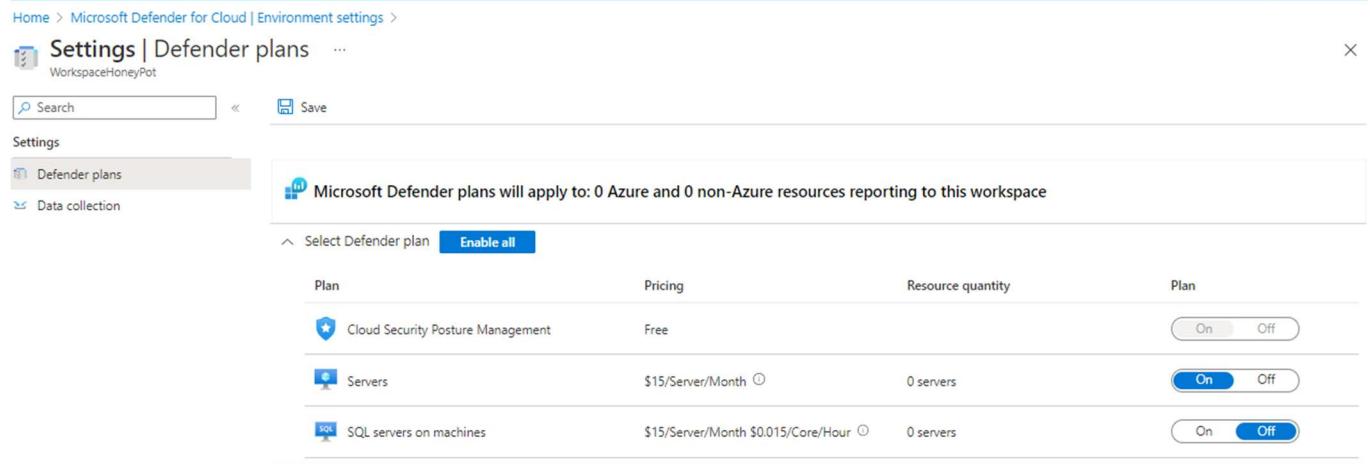
North Central US

[Review + Create](#)

[« Previous](#)

[Next : Tags >](#)

Turn on Microsoft Defender on for the virtual server by going into Microsoft Defender for cloud



Home > Microsoft Defender for Cloud | Environment settings >

Settings | Defender plans ...

WorkspaceHoneyPot

Search Save

Settings

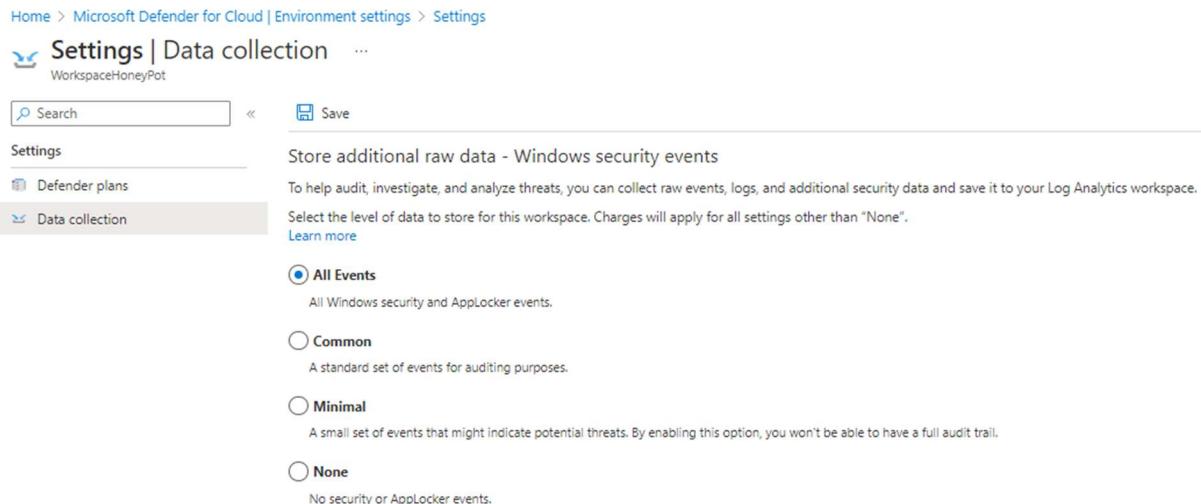
Defender plans Data collection

Microsoft Defender plans will apply to: 0 Azure and 0 non-Azure resources reporting to this workspace

Select Defender plan **Enable all**

Plan	Pricing	Resource quantity	Plan
Cloud Security Posture Management	Free	0 servers	<input type="button" value="On"/> <input type="button" value="Off"/>
Servers	\$15/Server/Month ⓘ	0 servers	<input type="button" value="On"/> <input type="button" value="Off"/>
SQL servers on machines	\$15/Server/Month \$0.015/Core/Hour ⓘ	0 servers	<input type="button" value="On"/> <input type="button" value="Off"/>

Click on All Events in Data collection to enable collection of all events which will eventually be sent to log Analytics



Home > Microsoft Defender for Cloud | Environment settings > Settings

Settings | Data collection ...

WorkspaceHoneyPot

Search Save

Settings

Defender plans Data collection

Store additional raw data - Windows security events

To help audit, investigate, and analyze threats, you can collect raw events, logs, and additional security data and save it to your Log Analytics workspace.

Select the level of data to store for this workspace. Charges will apply for all settings other than "None".

[Learn more](#)

All Events
All Windows security and AppLocker events.

Common
A standard set of events for auditing purposes.

Minimal
A small set of events that might indicate potential threats. By enabling this option, you won't be able to have a full audit trail.

None
No security or AppLocker events.

Connect Log Analytics Workspaces to the Virtual Machine. This will collect all logs from Virtual machine

Home > Log Analytics workspaces > WorkspaceHoneyPot

Log Analytics workspace | Virtual machines

snupripathi (techtwari.com)

+ Create Open recycle bin ...

Filter for any field... Name ↑

WorkspaceHoneyPot ...

Search Refresh

Workspace summary

Workbooks

Logs

Solutions

Usage and estimated costs

Properties

Service Map

Workspace Data Sources

Virtual machines

- Storage accounts logs
- System Center
- Azure Activity log
- Scope Configurations (Preview)

Related Resources

Automation Account

Monitoring

- Insights
- Alerts
- Diagnostic settings

Automation

- Tasks (preview)
- Export template

Support + troubleshooting

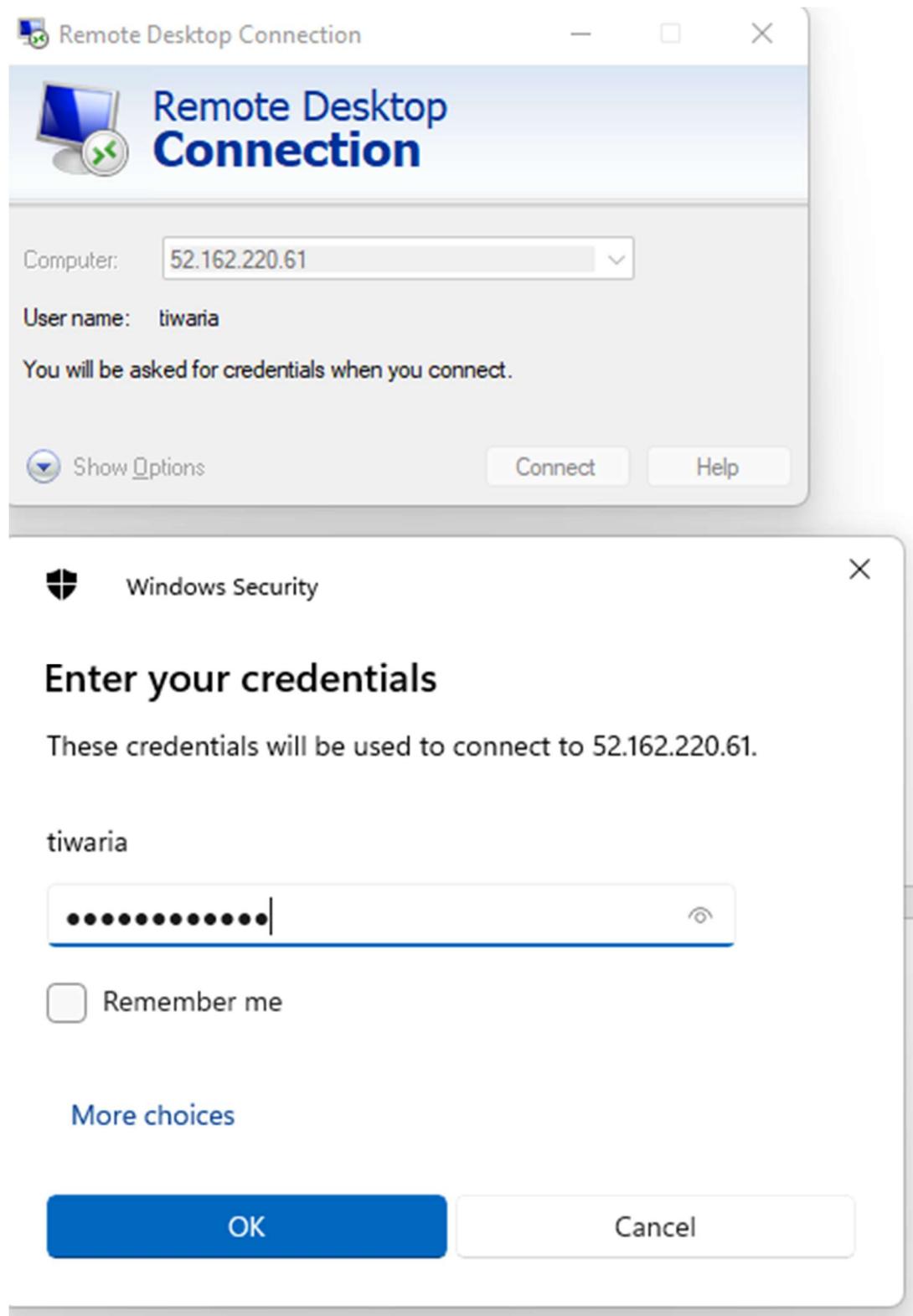
- Resource health
- New Support Request

< Page 1 >

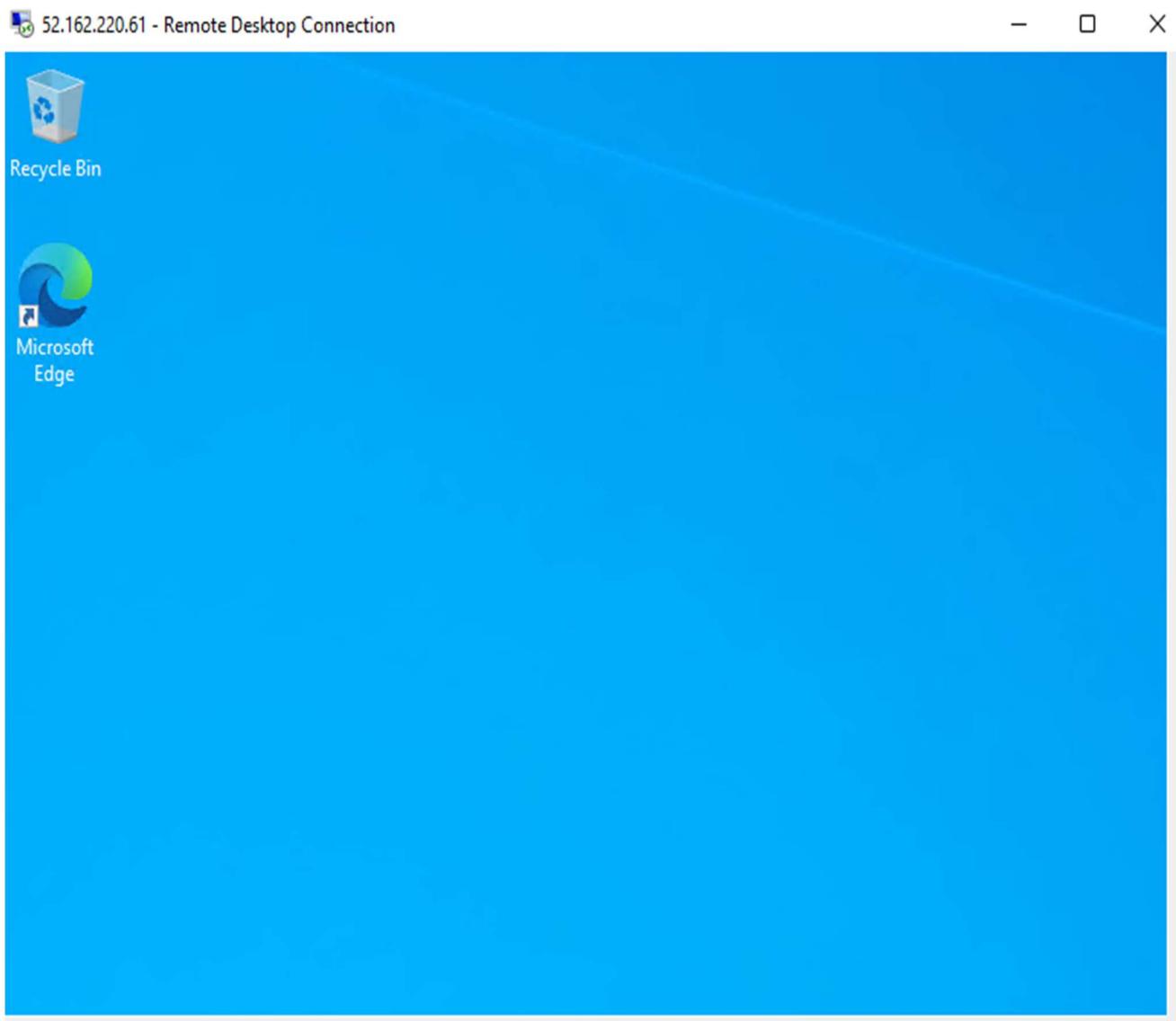
8 selected 2 sele... Visual Studio ... honeypotlab North ...

Name	Log Analytics Conn...	OS	Subscription	Resource group	Loc
HoneyPotVM	This workspace	Windows	2a3a0251-7a15-4a7...	HoneypotLab	nc

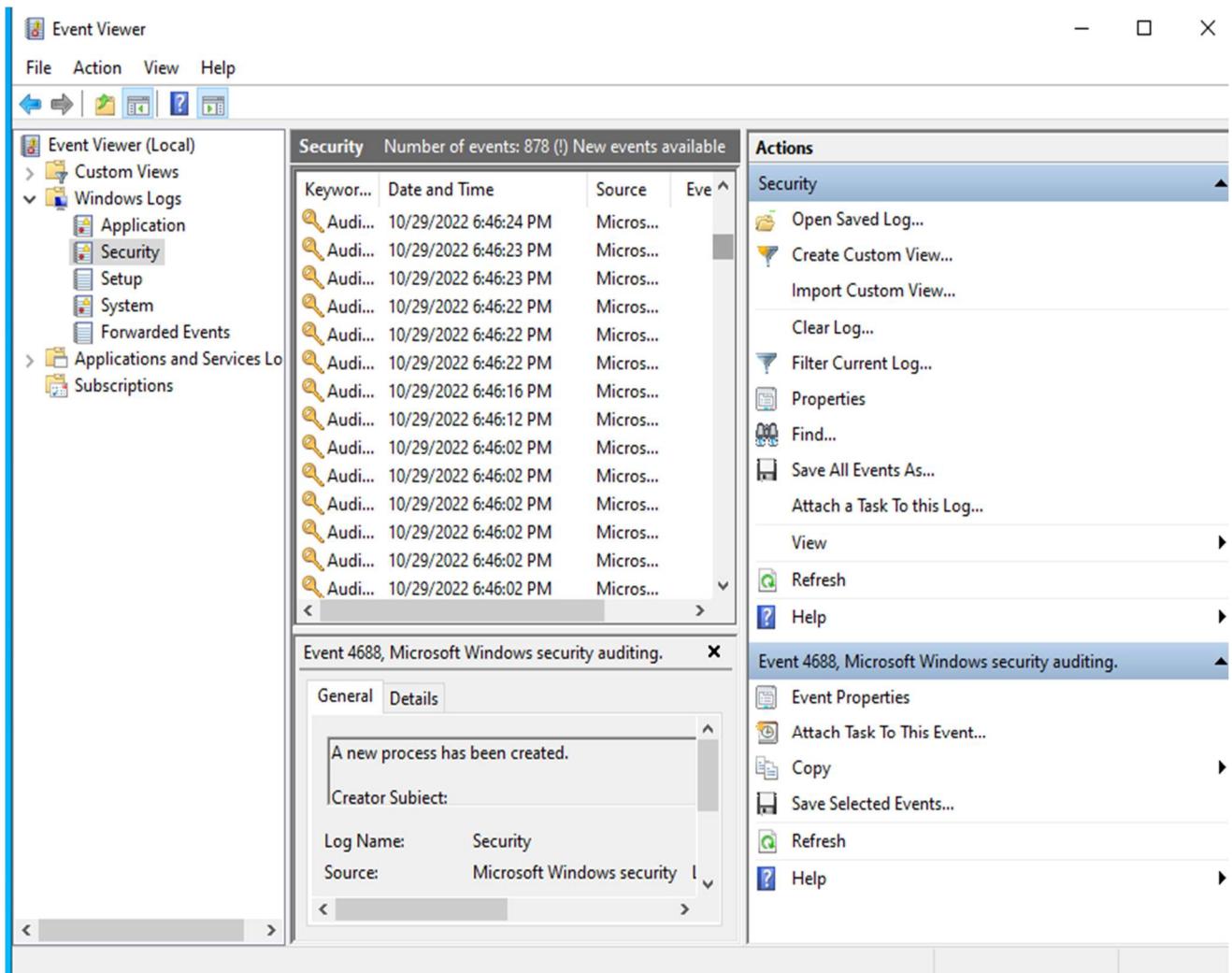
Login to the virtual Machine with the remote Desktop



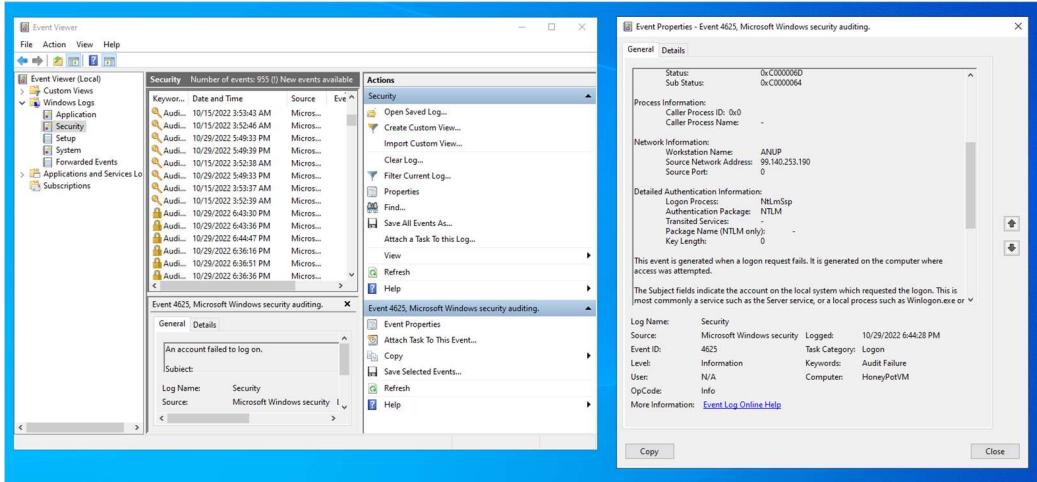
This is how the virtual machine looks like after Remote Desktop Connection



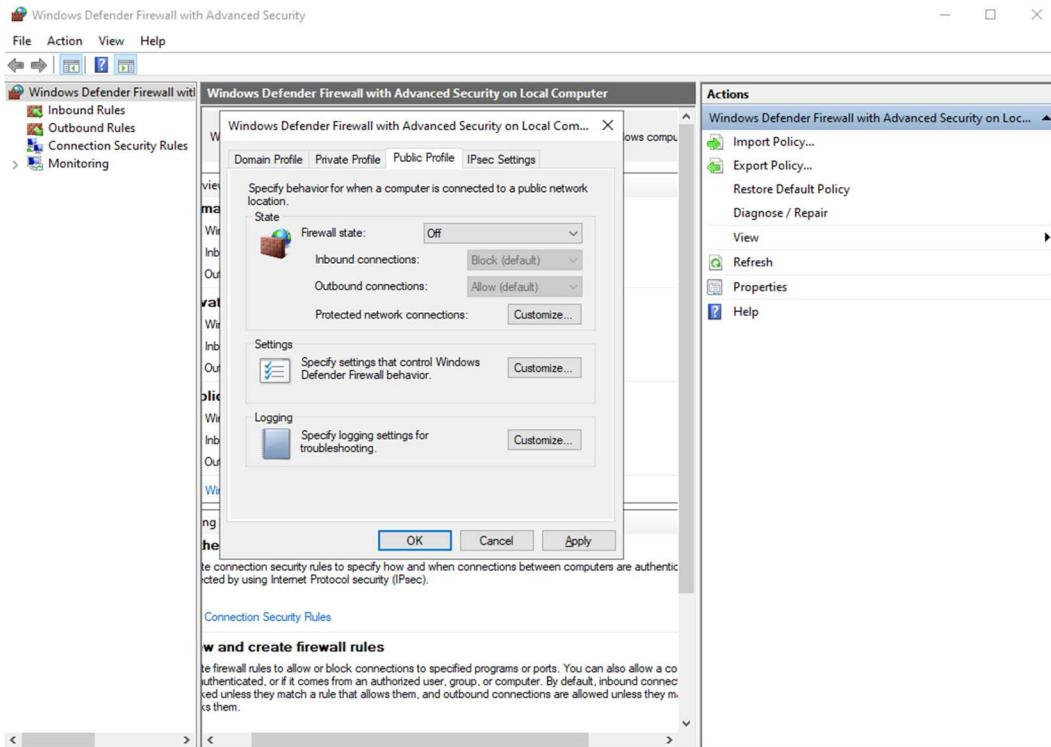
Go to Event Viewer > Windows Logs > Security. It takes some time for event logs to be visible. I am going to focus on Event ID 4625 which is Audit failure



If we click on one event id, it will show details about that failed login including Ip address



Open Windows Defender Firewall in VM and change the firewall state to off for all Domain Profile, Private Profile and Public Profile.



Ping the Ip address of virtual machine. It will start receiving ICMP packets after turning off the firewall

```
C:\ Command Prompt
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\Users\tiwar>ping 52.162.220.61

Pinging 52.162.220.61 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 52.162.220.61:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Users\tiwar>ping 52.162.220.61

Pinging 52.162.220.61 with 32 bytes of data:
Reply from 52.162.220.61: bytes=32 time=29ms TTL=111
Reply from 52.162.220.61: bytes=32 time=30ms TTL=111
Reply from 52.162.220.61: bytes=32 time=28ms TTL=111
Reply from 52.162.220.61: bytes=32 time=27ms TTL=111

Ping statistics for 52.162.220.61:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 27ms, Maximum = 30ms, Average = 28ms

C:\Users\tiwar>
```

Run the Following PowerShell Script in Virtual machine which exports logs from Virtual machine to Log Analytics

<https://github.com/axt3023/Powerrshell-Script-to-extract-geolocation--from-event-viewer>

Copy the PowerShell scripts in the virtual machine and save it in the VM desktop

```
Administrator: Windows PowerShell ISE

File Edit View Tools Debug Add-ons Help
Log_Exporter.ps1 X
1 # Get API key from here: https://ipgeolocation.io/
2 $API_KEY = "d4600b4efdef42b39828f515041a457"
3 $LOGFILE_NAME = "failed_rdp.log"
4 $LOGFILE_PATH = "C:\ProgramData\$($LOGFILE_NAME)"
5
6 # This filter will be used to filter failed RDP events from Windows Event Viewer
7 $XMLFilter = @'
8 <QueryList>
9   <Query Id="0" Path="Security">
10     <Select Path="Security">
11       *[System[(EventID='4625')]]
12     </Select>
13   </Query>
14 </QueryList>
15 '@
16
17 <#
18 This function creates a bunch of sample log files that will be used to train the
19 Extract feature in Log Analytics workspace. If you don't have enough log files to
20 "train" it, it will fail to extract certain fields for some reason _-.
21 We can avoid including these fake records on our map by filtering out all logs with
22 a destination host of "samplehost"
23 #>
24 Function write-Sample-Log() {
25   "latitude:47.91542,longitude:-120.60306,destinationhost:samplehost,username:fakeuser,sourcehost:2
26   "latitude:-22.90906,longitude:-47.06455,destinationhost:samplehost,username:lnwbaq,sourcehost:20
27   "latitude:52.37022,longitude:4.89517,destinationhost:samplehost,username:CSNYDER,sourcehost:89.24
28   "latitude:40.71455,longitude:-74.00714,destinationhost:samplehost,username:ADMINISTRATOR,sourceh
29   "latitude:33.99762,longitude:-6.84737,destinationhost:samplehost,username:AZUREUSER,sourcehost:10
30   "latitude:-5.32558,longitude:100.28595,destinationhost:samplehost,username:Test,sourcehost:42.1.6
31   "latitude:41.05722,longitude:28.84926,destinationhost:samplehost,username:AZUREUSER,sourcehost:17
32   "latitude:55.87925,longitude:37.54691,destinationhost:samplehost,username:Test,sourcehost:87.251.
33   "latitude:52.37018,longitude:4.87324,destinationhost:samplehost,username:AZUREUSER,sourcehost:20
34   "latitude:17.49163,longitude:-88.18704,destinationhost:samplehost,username:Test,sourcehost:45.227
35   "latitude:-55.88802,longitude:37.65136,destinationhost:samplehost,username:Test,sourcehost:94.232
36 }
37
38 # This block of code will create the log file if it doesn't already exist
39 if ((Test-Path $LOGFILE_PATH) -eq $false) {
40   New-Item -ItemType File -Path $LOGFILE_PATH
41   write-Sample-Log
42 }
43
44 # Infinite Loop that keeps checking the Event Viewer logs.
45 while ($true)
```

PS C:\Users\tiwaria>

Commands X

Modules: All Refresh

Name:

A:

- Add-AppvClientConnectionGroup
- Add-AppvClientPackage
- Add-AppvPublishingServer
- Add-AppxPackage
- Add-AppxProvisionedPackage
- Add-AppxVolume
- Add-BCDataCacheExtension
- Add-BitLockerKeyProtector
- Add-BitsFile
- Add-CertificateEnrollmentPolicyServer
- Add-Computer
- Add-Content
- Add-DnsClientNrrRule
- Add-DtcClusterTMMapping
- Add-EtwTraceProvider
- Add-History
- Add-InitiatorIdToMaskingSet
- Add-JobTrigger
- Add-KdsRootKey
- Add-LocalGroupMember
- Add-Member
- Add-MpPreference
- Add-NetEventNetworkAdapter
- Add-NetEventPacketCaptureProvider
- Add-NetEventProvider
- Add-NetEventVFPProvider
- Add-NetEventVmNetworkAdapter
- Add-NetEventVmSwitch
- Add-NetEventVmSwitchProvider
- Add-NetEventWFCaptureProvider
- Add-NetIphHttpsCertBinding
- Add-NetLbfoTeamMember
- Add-NetLbfoTeamNic

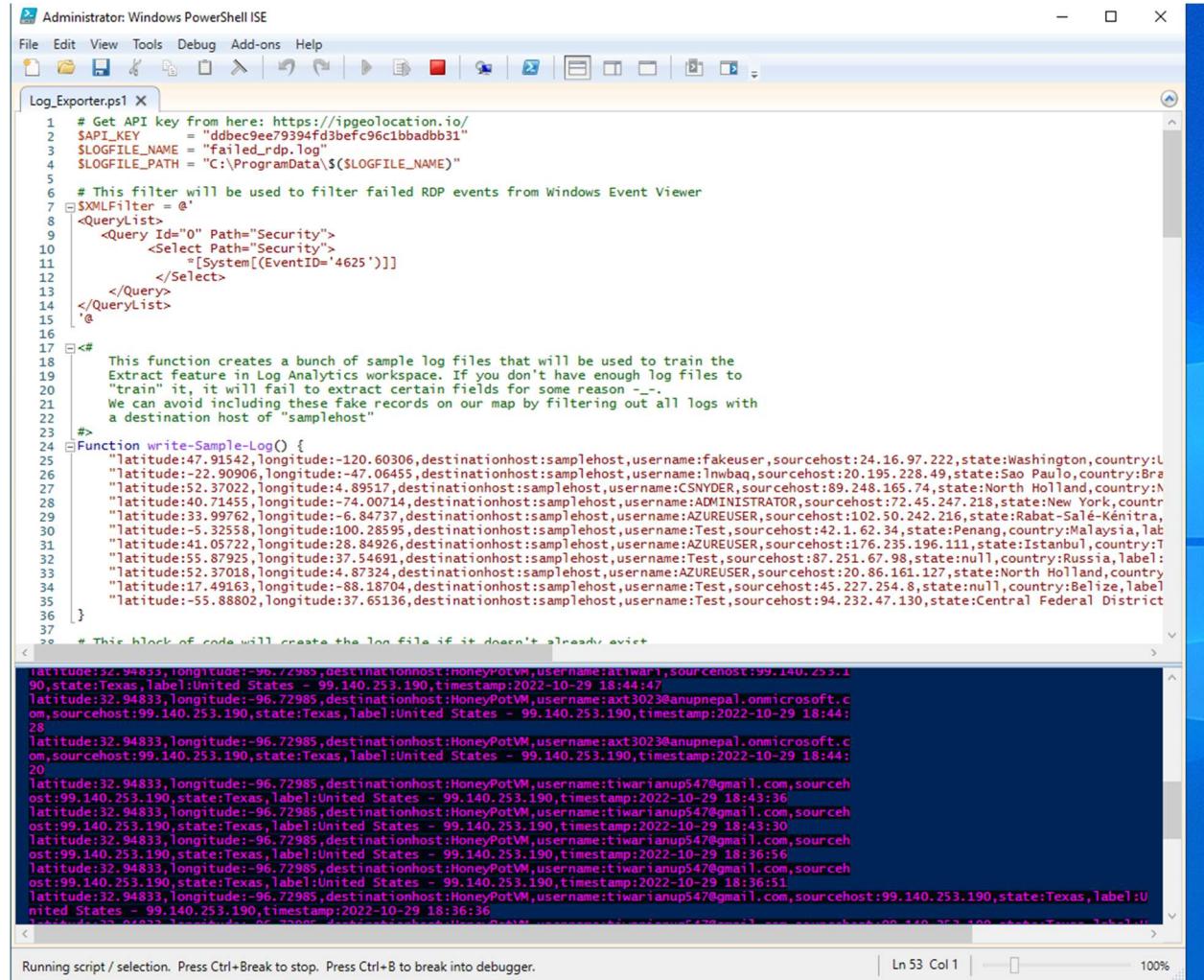
Run Insert Copy

Get the own API_Key from ipgeolocation.io and update in the PowerShell script

The screenshot shows a web browser window with the URL <https://ipgeolocation.io>. The page has a blue header with the logo "ipgeolocation". Below the header, there is a search bar with the placeholder "Enter any IPv4, IPv6 address or domain name:" and a text input field containing "52.162.220.61". To the right of the input field is a magnifying glass icon. A large block of JSON data is displayed below the search bar, representing the geolocation information for the given IP address. The data includes fields such as "ip", "country_name", "state_prov", "city", "latitude", "longitude", "time_zone", "isp", "currency", and "country". A "View More" button is located at the bottom right of the data block. At the bottom left of the page, there is a button labeled "Get Free API Access".

```
"ip": "52.162.220.61",
"country_name": "United States",
"state_prov": "Illinois",
"city": "Chicago",
"latitude": "41.88425",
"longitude": "-87.63245",
"time_zone": "America/Chicago",
"isp": "MICROSOFT-CORP-MSN-AS-BLOCK",
"currency": "US Dollar",
"country": "US"
```

Run the PowerShell script which will save the logs of failed data in the mentioned location in the scripts. The purple lines come as output every time there is failed login attempt



```
# Get API key from here: https://ipgeolocation.io/
$API_KEY = "ddbec9ee79394fd3befc96c1bbadb31"
$LOGFILE_NAME = "failed_rdp.log"
$LOGFILE_PATH = "C:\ProgramData\$($LOGFILE_NAME)"

# This filter will be used to filter failed RDP events from Windows Event Viewer
$XMLFilter = @'
<Query Id="0" Path="Security">
  <Select Path="Security">
    <![System[(EventID='4625')]]>
  </Select>
</Query>
</QueryList>
'@

<#
This function creates a bunch of sample log files that will be used to train the Extract feature in Log Analytics workspace. If you don't have enough log files to "train" it, it will fail to extract certain fields for some reason _-_. We can avoid including these fake records on our map by filtering out all logs with a destination host of "samplehost"
#>
Function write-Sample-Log() {
  "latitude:47.91542,longitude:-120.60306,destinationhost:samplehost,username:fakeuser,sourcehost:24.16.97.222,state:Washington,country:US
  "latitude:-22.90906,longitude:-47.06455,destinationhost:samplehost,username:lnwbaq,sourcehost:20.195.228.49,state:Sao Paulo,country:Brazil
  "latitude:52.37022,longitude:4.89517,destinationhost:samplehost,username:cSYNDER,sourcehost:89.248.165.74,state:North Holland,country:Netherlands
  "latitude:40.71455,longitude:-74.00714,destinationhost:samplehost,username:ADMINISTRATOR,sourcehost:72.45.247.218,state>New York,country:US
  "latitude:33.99762,longitude:-6.84737,destinationhost:samplehost,username:AZUREUSER,sourcehost:102.50.242.216,state:Rabat-Sale-Kénitra,Morocco
  "latitude:-5.32558,longitude:100.28595,destinationhost:samplehost,username:Test,sourcehost:42.1.62.34,state:Penang,country:Malaysia,lab
  "latitude:41.05722,longitude:28.84926,destinationhost:samplehost,username:AZUREUSER,sourcehost:176.235.196.111,state:Istanbul,country:Turkey
  "latitude:55.87925,longitude:37.54691,destinationhost:samplehost,username:Test,sourcehost:87.251.67.98,state:null,country:Russia,label:Russia
  "latitude:52.37018,longitude:4.87324,destinationhost:samplehost,username:AZUREUSER,sourcehost:20.86.161.127,state:North Holland,country:Netherlands
  "latitude:17.49163,longitude:-88.18704,destinationhost:samplehost,username:Test,sourcehost:45.227.254.8,state:null,country:Belize,label:Belize
  "latitude:-55.88802,longitude:37.65136,destinationhost:samplehost,username:Test,sourcehost:94.232.47.130,state:Central Federal District,Brazil
}

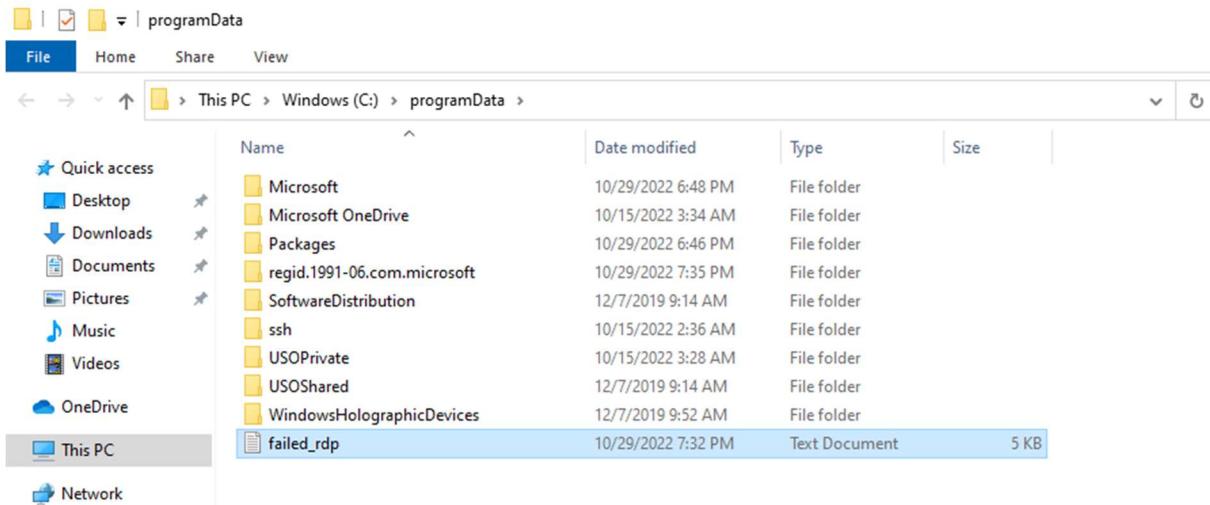
# This block of code will create the log file if it doesn't already exist
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:atiwarai,sourcenost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:44:47
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:axt3023@anupnepal.onmicrosoft.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:44:48
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:axt3023@anupnepal.onmicrosoft.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:44:49
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup54@gmail.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:43:36
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup54@gmail.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:43:30
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup54@gmail.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:36:56
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup54@gmail.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:36:51
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup54@gmail.com,sourcehost:99.140.253.190,state:Texas,label:United States - 99.140.253.190,timestamp:2022-10-29 18:36:36

```

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

Ln 53 Col 1 100%

There is a failed_rdp text documents in the programData location which is usually hidden. It should be accessed manually



Copy the data of failed_rdp and paste it into the device save it as failed_rdp.log file

Which is used in creating a custom log in log Analytics in Azure

The screenshot shows the Microsoft Azure Log Analytics workspace interface:

- Header:** Microsoft Azure, Search resources, services, and docs (G+/-)
- Breadcrumbs:** Home > Log Analytics workspaces > WorkspaceHoneyPot | Custom logs >
- Title:** Create a custom log
- Wizard Steps:** Sample (selected), Record delimiter, Collection paths, Details, Review + Create
- Description:** Upload a sample of the custom log. The wizard will parse and display the entries in this file. [Learn more](#)
- Sample log:** Select a sample log * (with a red asterisk)
- File Input:** Select a file (input field with a browse icon)

Specify the location C:\ProgramData\failed_Rdp.log so that it collects the log from t

Virtual machine

The screenshot shows the Microsoft Azure Log Analytics workspace interface. On the left, there's a sidebar with options like 'Create', 'Open recycle bin', 'Tags', 'Diagnose and solve problems', 'Locks', 'Agents management', 'Legacy agents management', 'Custom logs' (which is selected), 'Computer Groups', 'Data Export', 'Linked storage accounts', 'Network Isolation', and 'Tables'. The main area is titled 'WorkspaceHoneyPot | Custom logs' and shows a table with one result. The table has columns 'Name' and 'Type'. The single row is 'Failed_RDP_With_GEO_Location_CL' and 'File Based'. There are also tabs for 'Custom tables' and 'Custom fields'.

One failed login attempt from Thailand can be seen immediately in failed_rdp log file

in VM

```
latitude:47.91542,longitude:-120.60306,destinationhost:samplehost,username:fakeuser,sourcehost:24.16.97.222,state:Washington,country:United States,  
latitude:-22.90906,longitude:-47.06455,destinationhost:samplehost,username:lnwbaq,sourcehost:20.195.228.49,state:Sao Paulo,country:Brazil,label:Bra  
latitude:52.37022,longitude:4.89517,destinationhost:samplehost,username:CSNYDER,sourcehost:89.248.165.74,state:North Holland,country:Netherlands,la  
latitude:40.71455,longitude:-74.00714,destinationhost:samplehost,username:ADMINISTRATOR,sourcehost:72.45.247.218,state:New York,country:United Stat  
latitude:33.99762,longitude:-6.84737,destinationhost:samplehost,username:AZUREUSER,sourcehost:102.50.242.216,state:Rabat-Salé-Kénitra,country:Moroc  
latitude:-5.32558,longitude:100.28595,destinationhost:samplehost,username:Test,sourcehost:42.1.62.34,state:Penang,country:Malaysia,label:Malaysia -  
latitude:41.05722,longitude:28.84926,destinationhost:samplehost,username:AZUREUSER,sourcehost:176.235.196.111,state:Istanbul,country:Turkey,label:T  
latitude:55.87925,longitude:37.54691,destinationhost:samplehost,username:Test,sourcehost:87.251.67.98,state:null,country:Russia,label:Russia - 87.2  
latitude:52.37018,longitude:4.87324,destinationhost:samplehost,username:AZUREUSER,sourcehost:20.86.161.127,state:North Holland,country:Netherlands,  
latitude:17.49163,longitude:-88.18704,destinationhost:samplehost,username:Test,sourcehost:45.227.254.8,state:null,country:Belize,label:Belize - 45.  
latitude:-55.88802,longitude:37.65136,destinationhost:samplehost,username:Test,sourcehost:94.232.47.130,state:Central Federal District,country:Russ  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:atiwari,sourcehost:99.140.253.190,state:Texas,country:United States,labe  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:axt3023@anupnepal.onmicrosoft.com,sourcehost:99.140.253.190,state:Texas,  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:axt3023@anupnepal.onmicrosoft.com,sourcehost:99.140.253.190,state:Texas,  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup547@gmail.com,sourcehost:99.140.253.190,state:Texas,country:Ur  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup547@gmail.com,sourcehost:99.140.253.190,state:Texas,country:Ur  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup547@gmail.com,sourcehost:99.140.253.190,state:Texas,country:Ur  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup547@gmail.com,sourcehost:99.140.253.190,state:Texas,country:Ur  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup547@gmail.com,sourcehost:99.140.253.190,state:Texas,country:Ur  
latitude:32.94833,longitude:-96.72985,destinationhost:HoneyPotVM,username:tiwarianup547@gmail.com,sourcehost:99.140.253.190,state:Texas,country:Ur  
latitude:13.78445,longitude:100.57659,destinationhost:HoneyPotVM,username:azureuser,sourcehost:210.1.57.98,state:null,country:Thailand,label:Thail
```

All failed login attempt Logs can be seen in workspace too now

Click on one of the results and extract raw data to customize it

Extract Latitude, Longitude, and other Parameters and name them properly. This is how we train the extraction to be accurate.

Failed_RDP_With_GEO_Location_CL

FILTER	FIELD NAME	VALUE
☐	TenantId	: e4c202a8-2df7-46e0-a4ca-8e8098a8abab
☐	SourceSystem	: OpsManager
☐	ManagementGroupName	: AOI-e4c202a8-2df7-46e0-a4ca-8e8098a8abab
☐	TimeGenerated	: 2022-10-29T20:06:40.745750Z
☐	Computer	: HoneyPotVM
☐	RawData	: latitude : 47.91542 Field value : d Sta : Field Title : Latitude_CF Field Type : Numeric

Close Extract

Set up Map in Azure Sentinel by adding new workbook

Microsoft Azure Search resources, services, and docs (G+/-) ANUPNEPAL (TECHTIWARI.COM)

Home > Microsoft Sentinel > Microsoft Sentinel | Workbooks >

New workbook ... X

workspace:honeypot

Edit Open ⟳ ✖ ? Help ⌚ Auto refresh: Off

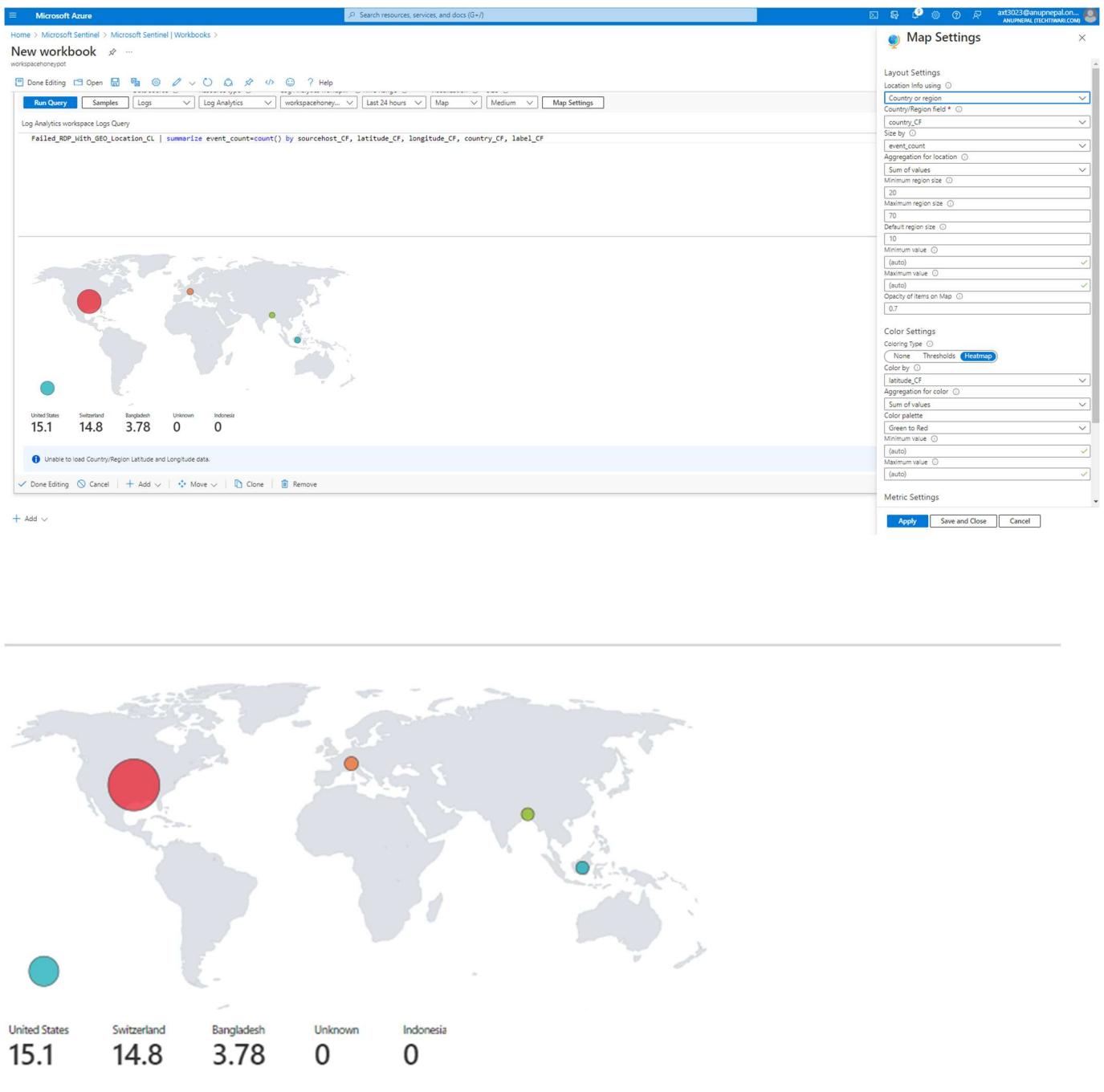
New workbook

Welcome to your new workbook. This area will display text formatted as markdown.

We've included a basic analytics query to get you started. Use the Edit button below each section to configure it or add more sections.

Category	Value
SecurityEvent	14.5 k
Failed_RDP_With_GEO_Location_CL	1.34 k
Heartbeat	214
Update	32
Usage	20
UpdateSummary	4
ProtectionStatus	4

Add new query and use map as a visualization option



After the setup is complete, the attacks can be seen in the world map.