Semi-Optional: Create a VM and onboard it to MDE if you haven't already **Do not use labuser/Cyberlab123! for credentials (or any other easy password). Your VM will most certainly get breached by a bad actor if it's on long enough. This has already happened once.**

- Leave it running for 90 minutes so it will be found on the internet and attacked by bad actors/bots. Alternatively, the pre-existing honeypot VM can be used/observed: **windows-target-1**
- Choose your scope for the lab: your own vm or windows-target-1

1. Preparation

- **Goal**: Set up the hunt by defining what you're looking for.
  - During routine maintenance, the security team is tasked with investigating any VMs in the shared services cluster (handling DNS, Domain Services, DHCP, etc.) that have mistakenly been exposed to the public internet. The goal is to identify any misconfigured VMs and check for potential brute-force login attempts/successes from external sources.
- **Activity**: Develop a hypothesis based on threat intelligence and security gaps (e.g., "Could there be lateral movement in the network?").
  - During the time the devices were unknowingly exposed to the internet, it's possible that someone could have actually brute-force logged into some of them since some of the older devices do not have account lockout configured for excessive failed login attempts.

2. Data Collection

- **Goal**: Gather relevant data from logs, network traffic, and endpoints.
  - Consider inspecting the logs to see which devices have been exposed to the internet and have received excessive failed login attempts. Take note of the source IP addresses and number of failures, etc.
- **Activity**: Ensure data is available from all key sources for analysis.
  - Ensure the relevant tables contain recent logs:
    - DeviceInfo
    - DeviceLogonEvents

3. Data Analysis

- **Goal**: Analyze data to test your hypothesis.
- **Activity**: Look for anomalies, patterns, or indicators of compromise (IOCs) using various tools and techniques.
  - Is there any evidence of brute force success (many failed logins followed by a success?) on your VM or ANY VMs in the environment?
  - If so, what else happened on that machine around the same time? Were any bad actors able to log in?

4. Investigation

- **Goal**: Investigate any suspicious findings.
- **Activity**: Dig deeper into detected threats, determine their scope, and escalate if necessary. See if anything you find matches TTPs within the [MITRE ATT&CK Framework.]
  - You can use ChatGPT to figure this out by pasting/uploading the logs: Scenario 1: TTPs

5. Response

- **Goal**: Mitigate any confirmed threats.
- **Activity**: Work with security teams to contain, remove, and recover from the threat.
  - Can anything be done?

6. Documentation

- **Goal**: Record your findings and learn from them.
- **Activity**: Document what you found and use it to improve future hunts and defenses.
  - Document what you did

7. Improvement

- **Goal**: Improve your security posture or refine your methods for the next hunt.
- **Activity**: Adjust strategies and tools based on what worked or didn't.
  - Anything we could have done to prevent the thing we hunted for? Any way we could have improved our hunting process?

Notes / Findings:

Sample Queries (spoilers, highlight or copy/paste to reveal):

```
// Check most failed logons
```

```
// Take the top 10 IPs with the most logon failures and see if any succeeded to logon
```

```
// Look for any remote IP addresses who have had both successful and failed logons
```

Timeline Summary and Findings:
arklab has been internet facing for a few days:

```
DeviceInfo
| distinct DeviceName == "ArkLab"
| where IsInternetFacing == True
```

Last internet facing time:
2025-06-22T16:22:50.5210308Z

Several bad actors have been discovered attempting to log into target machine:

```
DeviceLogonEvents
| where DeviceName == 'arklab'
| where LogonType has_any("Network", "Interactive", "RemoteInteractive", "Unlock")
| where ActionType == "LogonFailed"
| where isnotempty(RemoteIP)
| summarize Attempts = count() by ActionType, RemoteIP, DeviceName
| order by Attempts
```

ActionType	RemoteIP	DeviceName	Attempts
LogonFailed	185.224.3.219	arklab	165
LogonFailed	80.249.131.239	arklab	98
LogonFailed	176.65.150.72	arklab	42
LogonFailed	94.26.249.208	arklab	27
LogonFailed	185.243.96.107	arklab	20
LogonFailed	94.26.229.189	arklab	17
LogonFailed	82.148.20.48	arklab	11
LogonFailed	92.53.90.248	arklab	11
LogonFailed	92.53.65.234	arklab	9
LogonFailed	192.42.116.210	arklab	1
LogonFailed	23.129.64.143	arklab	1
LogonFailed	46.22.223.204	arklab	1

Top 6 most failed attempts from IP Addresses into VM; all unsuccessful
Query:

```
// Take the top 6 IPs with the most logon failures and see if any succeeded to logon
let RemoteIPsInQuestion = dynamic(["94.26.249.208","80.249.131.239", "94.26.229.189",
"92.53.90.248", "82.148.20.48", "92.53.65.234"]);
DeviceLogonEvents
| where LogonType has_any("Network", "Interactive", "RemoteInteractive", "Unlock")
| where ActionType == "LogonSuccess"
| where RemoteIP has_any(RemoteIPsInQuestion)
```

Query <No Results>

The only **successful** account was “arklab” in the last 30 days (49 total)

```
DeviceLogonEvents
| where DeviceName == "arklab"
| where LogonType == "Network"
| where ActionType == "LogonSuccess"
| where Accountname == “arklab”
| summarize count()
```

There were (0) failed logins into “arklab” indicating a brute force is unlikely and and a 1-time password guess is unlikely

```
DeviceLogonEvents
| where DeviceName == "arklab"
| where LogonType == "Network"
```

```
| where ActionType == "LogonFailed"
| where Accountname == “arklab”
| summarize count()
```

We checked all successful logins from ‘arklab’ to see if there were any of them unusual or from unexpected locations. All were successful matches

```
DeviceLogonEvents
| where DeviceName == 'arklab'
| where LogonType == 'Network'
| where ActionType == 'LogonSuccess'
| where AccountName == 'arklab'
| summarize LoginCount =count() by DeviceName, ActionType, AccountName, RemoteIP
```

DeviceName	ActionType	AccountName	RemoteIP	LoginCount
arklab	LogonSuccess	arklab		1
arklab	LogonSuccess	arklab	10.0.8.7	1

Though the device was exposed to the internet and clear brute force attempts have taken place, there is no evidence of any brute force success or unauthorized access from the legitimate users

Relvent TTPs:
T1078 - Valid Accounts
- Attempted use of legitimate account ("arklab") observed in logs.
- However, only expected and legitimate logins occurred, with no signs of compromise.

T1110 - Brute Force
- Multiple failed login attempts from remote IPs targeting "ArkLab".
- No successful logins from attacker IPs, indicating failed brute-force attempts.

T1016 - System Network Configuration Discovery *(Inferred)*
- Since the machine was internet-facing, it may have been scanned prior to login attempts.
- While not explicitly logged, scanning and reconnaissance are typically precursors to brute force attacks.

T1046 - Network Service Scanning *(Inferred)*
- Brute force attempts suggest the system’s exposed services were likely discovered through scanning.

Recommended Mitigations (from MITRE ATT&CK)
M1036 – Account Use Policies: Enforce strong password policies and disable unused accounts. **M1026 – Privileged Account Management**: Use just-in-time access and monitor privileged accounts.
M1037 – Network Segmentation: Limit exposure of systems to the public internet when not required.
M1042 – Disable or Remove Feature or Program: Disable remote access methods when not in use.

Response Actions:

- Hardened the NSG attached to arklab to allow only RDP traffic from specific endpoints (no public internet access)

- Implemented account lockout policy
- Implemented MFA