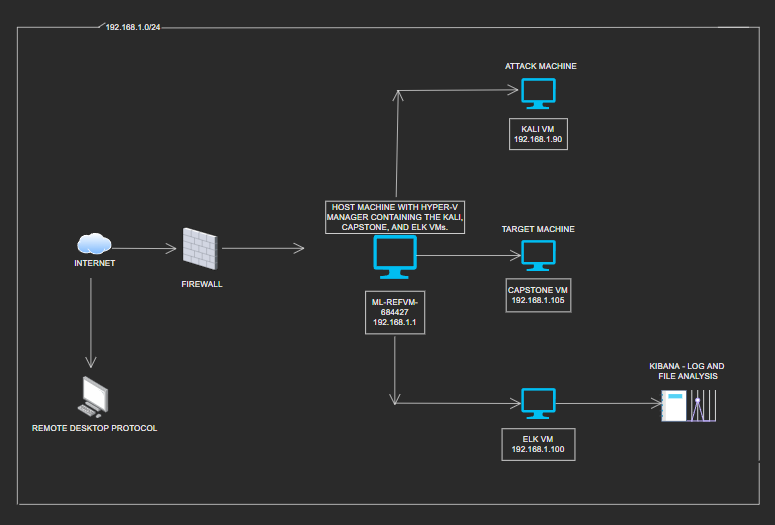
#### **Question 1: Setting Alerts in a New Monitoring System**

"How do you determine which alerts to set in a new monitoring system?"

Determining which alerts to set in a new monitoring system is an important aspect in ensuring that alert fatigue does not create a hazardous environment. According to a conference paper published in Lecture Notes in Computer Science Volume 11953, “The threat-alert fatigue problem, which is the inability of security operators to genuinely investigate each alert coming from network-based intrusion detection systems, causes many unexplored alerts and hence a deterioration of the quality of service” (p. 1). Companies need the alerts to be valid and necessary without inundating analysts with overwhelming data to investigate.

In this project, we set up a network to mirror our ability to identify critical vulnerabilities on a client’s network. As you can see in the diagram below, My network began with connecting my Remote Desktop Protocol to the internet, which then allowed me to connect to the host machine, ML-REFVM-684427. The host machine supported the Hyper-V Manager, which contained the attack machine (Kali), the target machine (Capstone), and the ELK VM (which sent logs to Kibana for analysis).



Malicious traffic such as brute force attacks and unapproved traffic on open Port 80 are some examples of what are likely to appear on the network. The brute force attack that was used in this project involved correctly determining a password for one of the client’s users, which ultimately allowed me to gain access to their server and confidential information. It should be noted that the usernames and passwords used by the client were all very simple and did not conform to necessary rules such as minimum 10 characters, inclusion of numbers and special characters, and usernames not mirroring the employee’s name.

For alerts pertaining to brute force attacks:

* An alert should be sent to the SOC analyst any time a user has 5 unsuccessful login attempts in a 1 minute timeframe
  + There are very few instances in which a user would have that volume of unsuccessful login attempts within a 1 minute timeframe; this would point to a brute force attack, especially if there were 3 different accounts being tested at the same time, which is what I did while performing the brute force attack on the client network.

For alerts pertaining to open Port 80:

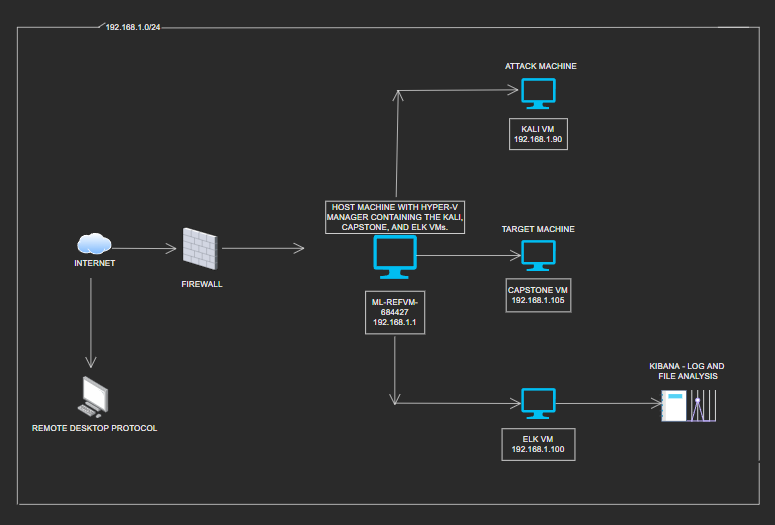
* Port 80 is commonly used for web communication, and in this project, the client left it open for public access. The vulnerability allowed the attacker machine access into the target machine’s web servers, where the secret files were accessible and found.
* An alert should be sent to the SOC analyst any time an IP address sends out more than 10 requests per second.
  + This volume of requests indicates a potential port scan and if Port 80 is going to be left open for public access, then there needs to be an alert in place any time there is potential for malicious port scanning.

#### **Question 1: Planning an Engagement**

"How do you plan and execute an effective offensive engagement?"

In Project 2, we planned and executed an offensive engagement to assist the client in assessing any vulnerabilities on their network.

In this project, there were multiple VMs on the network.



The ML-REFVM-684427 hosted the Hyper-V manager, which included access to the Kali VM, which was the attack machine; the Capstone VM, which was the target machine; and the ELK VM, which transferred log data to Kibana for analysis. I utilized the Kali VM to infiltrate the Capstone VM to expose the vulnerabilities and exploitations.

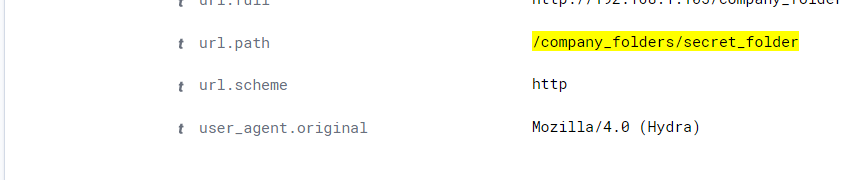
To perform the infiltration, I used tools such as nmap, which allowed me to scan for open ports on the target machine; unhashing a password within a secret folder; performing a brute force attack via Hydra in order to discover the correct password for users (with one successful attack); and successfully implementing a reverse php shell payload to the target machine.

The client’s network had some basic security measures in place: they had their secret file relatively hidden, they had a password hashed, and they only had one user who was allowed access to the server.  
The nmap command allowed me to gain access to the Index of /company\_folders through the webserver IP address 192.168.1.105. Reading through these folders lead to my knowledge of a secret\_folder.

Using the Hydra tool on Kali Linux allowed me to perform brute force attacks on the 3 users simultaneously, with the successful attack happening against the username “ashton.” This, in turn, lead to Ryan’s password hash as well as instructions on how to connect to the company’s webDAV server using Ryan’s password. The website crackstation.net allows anyone to type in a hash and receive the unhashed output in seconds. This is how I was able to obtain Ryan’s password, “linux4u”.  
After obtaining Ryan’s password, I was able to utilize msfvenom to create a multi/user exploit and meterpreter shell (reverse php) and execute it on the target machine’s Apache server (essentially uploading and opening the php payload onto the target machine’s port 4444 and allowing remote connection). Once this payload was executed, it provided me, the attacker, access to the server at 192.168.1.105. This is how I found the flag, “b1ng0w@5h1sn@m0”.

The methods of this engagement could have been detectable if there were certain security measures and alerts put in place by the client which would notify someone that potentially malicious activity was occurring.

As I found working on this project, the methods used to attack and infiltrate the target machine were also detectable through log analysis on Kibana. This would be the same for other providers, such as SPLUNK, as long as the analyst was looking for the right information in the log data. An example of this can be seen in the screenshot below, where we see that the user\_agent includes “(Hydra)” - the tool used to implement the brute force attack.



## RESOURCES:

## A Machine Learning Approach to Detection of Critical Alerts from Imbalanced Multi-Appliance Threat Alert Logs

Samuel Ndichu, Tao Ban, Takeshi Takahashi and Daisuke Inoue

Conference: 2021 IEEE International Conference on Big Data (Big Data), Year: 2021, Page 2119

DOI: [10.1109/BigData52589.2021.9671956](http://dx.doi.org/10.1109/BigData52589.2021.9671956)