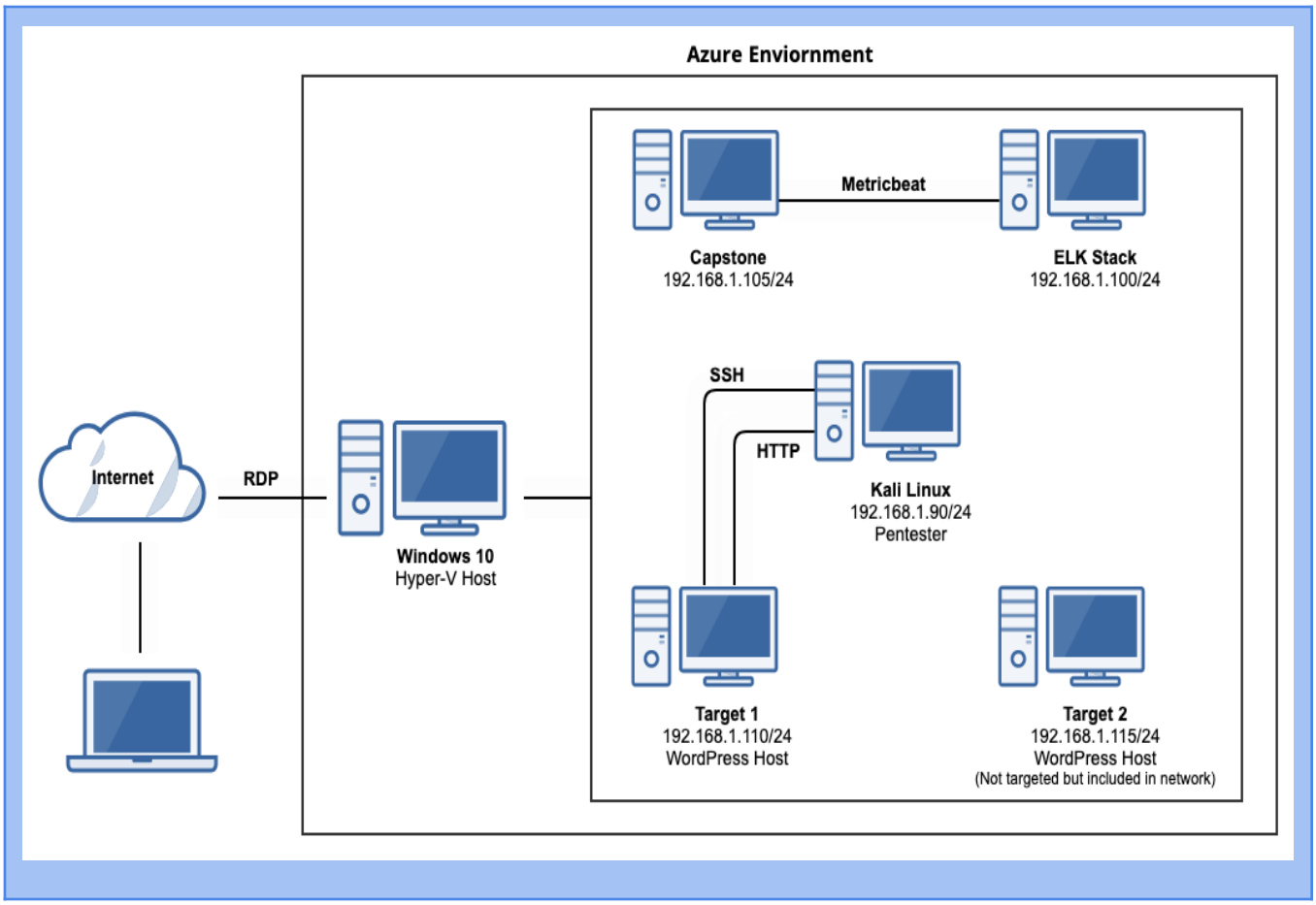
Blue Team (Defensive) Analysis Report - Summary of Operations

Table of Contents

- Network Topology
- Description of Targets
- Monitoring the Targets
- Patterns of Traffic & Behavior
- Suggestions for Going Further

Network Topology



The following machines were identified on the network:

- Name of VM 1 **Kali**
  - **Operating System**: Linux

    - **Purpose**: The Penetration Testing Machine attacking the vulnerable target.
    - **IP Address**: 192.168.1.90
- Name of VM 2 **Capstone**
    - **Operating System**: Linux
    - **Purpose**: A testing machine for enabling alerts
    - **IP Address**: 192.168.1.105
- Name of VM 3 **ELK**
    - **Operating System:** Linux
    - **Purpose:** Used for gathering information from the victim machine using Metricbeat, Filebeats, and Packetbeats
    - **IP Address:** 192.168.1.100
- Name of VM 4 **Target 1**
    - **Operating System:** Linux
    - **Purpose:** The vulnerable VM with WordPress
    - **IP Address:** 192.168.1.110
- Name of VM 5 **Hyper V Manager-Host**
    - **Operating System:** Windows 10
    - **Purpose:** Contains the vulnerable machines and the attacking machine
    - **IP Address:** 192.168.1.1

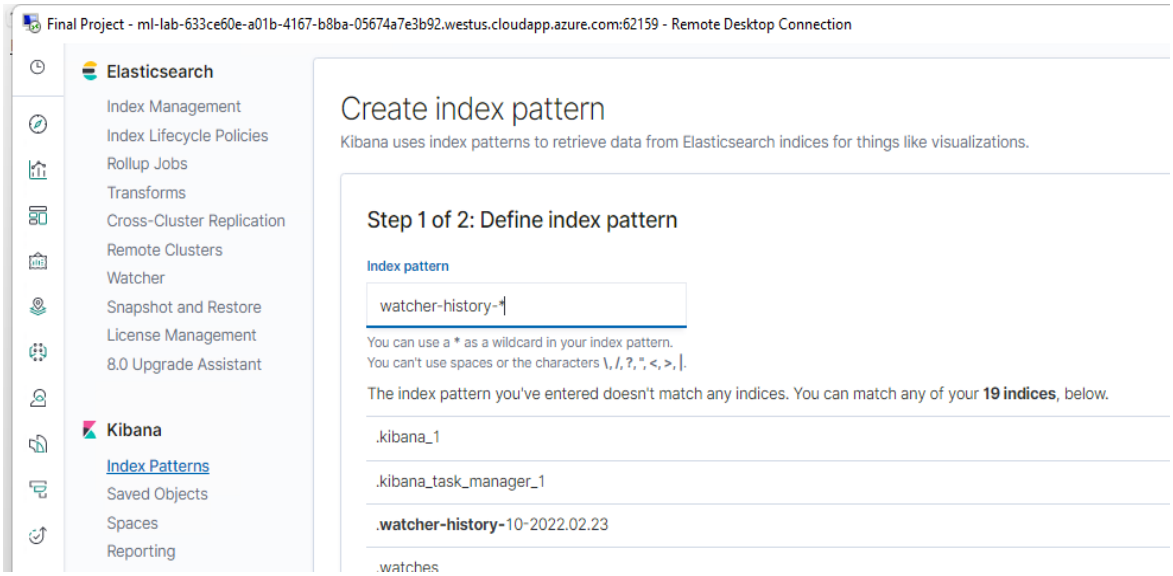
Description of Targets

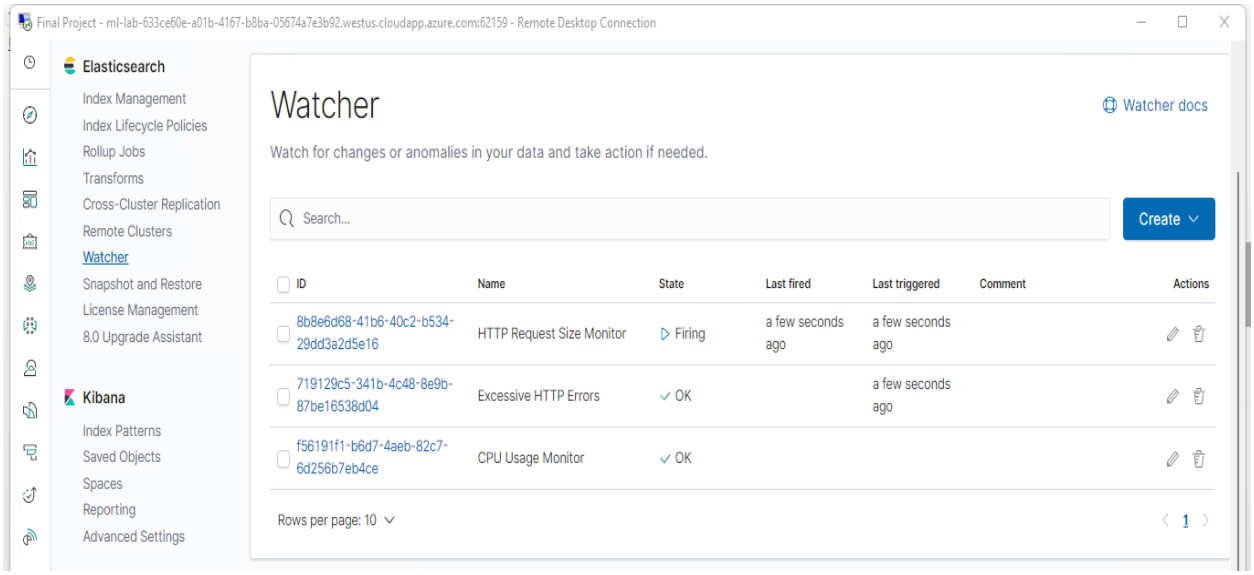
The target of this attack was: Target 1 (192.168.1.110).

Target 1 is an Apache web server and has SSH enabled, so ports 80 and 22 are possible ports of entry for attackers. As such, the following alerts have been implemented:

Monitoring the Targets

Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:





Alert 1: CPU Usage Monitor

CPU Usage Monitor is implemented as follows:

- **Metric**: Metricbeat: system.process.cpu.total.pct
  WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes
- **Threshold**: The maximum cpu total percentage is over .5 in 5 minutes.
- **Vulnerability Mitigated**:
  - Malicious software, programs (malware or viruses) running taking up resources can be detected.

- ❖ Controlling the CPU usage percentage at 50%, will trigger a memory alert only if the CPU remains at or above 50% consistently for 5 minutes.

- **Reliability**: **Medium**

  - ❖ This alert can generate a lot of false positives due to CPU spikes occurring when specific integrations are initiated at the start of processing.
  - ❖ More effective to be used for analysis, when it is used along with the other 2 alerts created.
  - ❖ One more advantage is - Even if there isn't a malicious program running this can still help determine where to improve on CPU usage.

Alert 2: HTTP Request Size Monitor

HTTP Request Size Monitor is implemented as follows:

- **Metric**: Packetbeat: http.request.bytes
  - **When sum() of http.request.bytes OVER all documents is ABOVE 3500 for the LAST 1 minute**
- **Threshold**: The sum of the requested bytes is over 3500 in 1 minute
- **Vulnerability Mitigated**:
  - Identification of DoS attacks.
  - Code injection in HTTP requests (XSS and CRLF) or DDOS.
  - protection is enabled to detect or prevent DDOS attacks for IPS/IDS by controlling the number of http request sizes through a filter.
- **Reliability**: **Medium**

  - There is a possibility for a large non malicious HTTP request or legitimate HTTP traffic getting captured too.
- This alert doesn't generate an excessive amount of false positives because DDOS attacks submit requests within seconds, not within minutes.

Alert 3:Excessive HTTP Errors

Excessive HTTP Errors is implemented as follows:

- **Metric**: Packetbeat: http.response.status_code

- **When count() GROUPED OVER top5 'http.response.status_code' is above 400 for the last 5 minutes**

- **Threshold**: grouped http response status codes above 400 every 5 minutes
- **Vulnerability Mitigated**:
  - Identification of brute force attacks.
  - Could help by using intrusion detection/prevention for attacks - IPS would block any suspicious IP's
  - Utilize Account Management to lock or request user accounts to change the passwords every 60 days
- **Reliability**: **High**.

  - Measuring by error codes 400 and above will filter out any normal or successful responses. 400+ codes are client and server errors which are of more concern. Especially when taking into account these error codes going off at a high rate.

- This alert will not generate an excessive amount of false positives as it's more targeted at identifying brute force attacks.

Security Recommendations

Alerts only detect malicious behavior, but do not stop it. Each alert above pertains to a specific vulnerability/exploit. The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats, identified by the alerts above. In addition to watching for occurrences of such threats, the network should be hardened against them.

Hence, as Blue Team, we suggest that IT implement the fixes below to protect the network:

Vulnerability 1:Enumeration and Brute Force Attacks (**Excessive HTTP Errors**)

Patch: WordPress Hardening

- Implement regular updates to WordPress including WordPress Core, PHP version, Plugins.
- Install security plugin(s) for Ex. Wordfence (adds security functionality)
- Disable unused WordPress features and settings like:
  - WordPress XML-RPC (on by default)
  - WordPress REST API (on by default)
- Block requests to /?author=<number> by configuring web server settings
- Remove WordPress logins from being publicly accessible specifically: /wp-admin and /wp-login.php

Why It Works:

- Regular updates to WordPress, the PHP version and plugins is an easy way to implement patches or fixes to exploits/vulnerabilities.
- Depending on the WordPress security plugin it can provide things like:
  - Malware scans
  - Firewall
  - IP options (to monitor/block suspicious traffic)
- REST API is used by WPScan to enumerate users
  - Disabling it will help mitigate WPScan or enumeration in general
- XML-RPC uses HTTP as it's method of data transport
- WordPress links (permalinks) can include authors (users)
  - Blocking request to view all users helps mitigate against user enumeration attacks
- Removal of public access to WordPress login helps reduce the attack surface

Vulnerability 2:Code Injection in HTTP Requests (XSS and CRLF) and DDOS (**HTTP Request Size Monitor**)

Patch: Code Injection/DDOS Hardening

- Implementation of HTTP Request Limit on the web server
  - Limits can include a number of things:
    - Maximum URL Length
    - Maximum length of a query string
    - Maximum size of a request

- Implementation of input validation on forms

Why It Works:

- If an HTTP request URL length, query string and oversize limit of the request occurs, a 404 range of errors will occur. This will help reject these requests that are too large.

o Input validation can help protect against malicious data anyone attempts to send to the server via the website or application in/across a HTTP request.

Vulnerability 3:Malicious Code (Malware and Viruses) and Resource Utilization (**CPU Usage Monitor**)

Patch: Virus or Malware hardening

- Add or update antivirus software.
- Implement and configure Host Based Intrusion Detection System (HIDS)
  - Ex. SNORT (HIDS)

Why It Works:

- Antiviruses specialize in removal, detection and overall prevention of malicious threats against computers.
  - Any modern antivirus covers more than viruses and is a very robust solution to protecting a computer in general.
- HIDS monitors and analyzes internals of computing systems. They also monitor and analyze network packets.