300471606 subankamo

Lab 5 - Intrusion Detection Systems

Due Date – 04/06/2023

- Follow the instructions in this document,
- Answer the questions in the order they appear in this document and in the labs included in this document (See each lab's document for more instructions)
- Submit the file (.doc, .docx, .pdf) using the ecs submission system (i.e. Lab5). https://apps.ecs.vuw.ac.nz/submit/CYBR371

Instructions

Part A (2 Marks Total):

Login into netlab at netlab.ecs.vuw.ac.nz, complete the following labs and answer the questions accordingly:

- **Identifying and Analyzing NHIDS Alerts**

Task:

A. Write a 300 word summary of what you did and learned in this lab.

In this lab, I learned about tools (Zenmap, sguil, and squert) to scan networks and analyse security alerts.
We started by logging into the SecOnion virtual machine and checking the status of the Network Security Monitoring service. If everything was fine, we moved on; otherwise, we started or restarted the service.
Then we launched the Kali virtual machine and opened Zenmap through the Kali PC Viewer. We typed in the IP addresses of our targets and hit the scan button. After the scan, we checked out the results to see which ports were open on different systems.
Next, we switched back to the SecOnion system and fired up Sguil, which is a tool for security monitoring. We logged in and made sure all checkboxes were selected to start monitoring.
Going back to Kali, we returned to Zenmap and scanned another target IP address using a different scan profile. Once the scan was complete, we went back to Sguil.
In Sguil, we organised the events by date and selected an interesting event related to an NMAP OS Detection. We had a closer look at the packet data and the associated rule. We also exported a detailed report for that event, saved it, and checked it out using a terminal command.

Then we explore Squert, another tool for security monitoring. We launched it, logged in, and refreshed the dashboard to see the latest events. We played around with filters to focus on specific events, like those hitting the DVL Server or picked up by a particular sensor. Overall, it was an interesting lab where we got to experiment with scanning networks, analysing security alerts, and gaining insights into potential vulnerabilities and threats using Zenmap, sguil, and squert. These tools provided valuable information and helped us generate detailed reports for further analysis and reporting.

Part B (2 Marks Total):

Login into netlab at netlab.ecs.vuw.ac.nz, complete the following lab and answer the questions accordingly:

- **Tripwire Host Based Intrusion Detection System**

Questions

Write Tripwire rules for the following Windows 7/10 files and directories. The rules must contain: 1) Proper rule name, 2) Severity level, 3) Specific folder/subfolders with associated monitoring level, 4) A description of the rule and 5) Justification for the given severity and monitoring level for each file and directory included in the rule.

a) explorer.exe
b) Windows\Temp folder
c) Regedit.exe
d) msports.dll

You may look at examples of Tripwire rules for Linux systems for reference. Tripwire documentation also provides comprehensive guidelines on creating policies for various types of systems.

>The folder/subfolder paths mentioned in the rules are based on the assumption that the specified files and directories are located in their default locations on Windows 7/10.

a) Rule for explorer.exe:
Rule Name: Explorer.exe Integrity Monitoring
Severity Level: High
Folder/Subfolders: C:\Windows\explorer.exe
Monitoring Level: Critical
Description: This rule monitors the integrity of the explorer.exe file, which is responsible for the Windows graphical user interface. Any modifications to this file can indicate unauthorised changes or potential malware activity.
Justification: The severity level is set to high because any unauthorised modification to explorer.exe can have a significant impact on the system's stability and security. The monitoring level is critical because any change to this file should be thoroughly investigated and considered potentially malicious.

b) Rule for Windows\Temp folder:

300471606 subankamo

Rule Name: Windows Temp Folder Monitoring
Severity Level: Medium
Folder/Subfolders: C:\Windows\Temp
Monitoring Level: Log Only
Description: This rule monitors the Windows temporary folder, where various temporary files are stored during system operations. Changes to files in this folder may indicate suspicious activities or potentially unwanted programs.
Justification: The severity level is set to medium as modifications to files in the temporary folder may not directly compromise system integrity but can potentially lead to security vulnerabilities. The monitoring level is set to log only as not all changes in this folder may be malicious, but monitoring can provide valuable information for investigation purposes.

c) Rule for Regedit.exe:
Rule Name: Regedit.exe Execution Monitoring
Severity Level: Medium
Folder/Subfolders: C:\Windows\Regedit.exe
Monitoring Level: Log Only
Description: This rule monitors the execution of Regedit.exe, the Windows Registry Editor. Monitoring its execution can help detect potential unauthorised changes to the system registry, which can have a significant impact on system stability and security.
Justification: The severity level is set to medium as the execution of Regedit.exe can allow users to modify critical system settings, and monitoring its execution provides insight into potential malicious activities. The monitoring level is set to log only as legitimate use of Regedit.exe is common, and not all executions may be malicious.

d) Rule for msports.dll:
Rule Name: Msports.dll File Integrity Monitoring
Severity Level: Low
Folder/Subfolders: C:\Windows\System32\msports.dll
Monitoring Level: None
Description: This rule monitors the integrity of the msports.dll file, which is responsible for managing serial communications ports in Windows. Monitoring changes to this file can help detect potential modifications or corruption.
Justification: The severity level is set to low as modifications to msports.dll are less likely to directly impact system stability or security. The monitoring level is set to none because monitoring changes to this file may not provide significant value in terms of detecting malicious activities, and regular system updates may modify this file.