

Lab 10

Detection and Log Analysis

ITSC304: Operating Systems Exploitation

Table of Contents

[Lab Outcome(s) 3](#_Toc473929636)

[Reading 3](#_Toc473929637)

[Introduction 3](#_Toc473929638)

[1.0 Monitor Windows Events with Sysmon 4](#_Toc473929640)

[2.0 Windows Event Log Monitor and Analysis -ELK 7](#_Toc473929642)

L*abs must be submitted by the due date for full credit. After due date late submissions will be accepted for a period of one week (seven days) and the grade will be reduced by ten percent (10%) per day after due day.* ***Assignments that are submitted more than seven days late will receive a grade of zero (0).***

I certify that the work submitted in this assignment is my own and that it has not been taken in whole or in part from any other source. I understand that the penalty for plagiarism will include a grade of zero (0) for this assignment plus disciplinary action in accordance with SAIT policies.

**EVALUATION**

|  |  |  |
| --- | --- | --- |
| Monitor Windows Events with Sysmon | 40 |  |
| Windows Log Monitor and Analysis with ELK-Stack | 30 |  |
| TOTAL MARK | 70 |  |

Lab Outcome(s)

* Mitigate Windows threads
* Detection with Sysmon
* Log monitor and Analysis

Reading

* Textbook sections as defined in the Course Schedule.
* <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon#examples>
* <https://github.com/SwiftOnSecurity/sysmon-config>
* <https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/>
* <https://www.elastic.co/elk-stack>.
* <https://www.elastic.co/elk-stack>.

Introduction

There are many detection techniques. None of these techniques can guaranteed, but the chances of detection increases if you're understand the "normal" operations of a system such that you're able to compare it to suspect system.

Lab Requirements:

To complete this Lab you need the following:

1. Virtual-Box latest version
2. Ubuntu 18 or higher with 2 cores and at least 8 GB RAM
3. Windows 7 x64 virtual machine
4. Windows 10 x64 virtual machine
5. Monitor Windows Events with Sysmon \_\_\_/40

The objective of this exercise is to use Sysmon tool to monitor and log windows events

Microsoft developed Sysmon (System monitor) which is a service and a device driver to monitor and log system activity to the Windows event log which provides detailed information on process creation, suspicious access to other processes, network connections, changes in file creation time, suspicious file activity, registry changes, named pipes, changes to Sysmon itself. Sysmon generates events logs that can be collected for monitoring and analysis into a central system such as SIEMS.

1. Use Virtual machines Windows 10 or Windows 7 to install Sysmon.
2. **(2 marks)** Access the following web site <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon#examples> read it and follow instructions to install Sysmon. List 4 Sysmon Features

**NOTE:** You can also use this site <https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/> to find more sysmon event details, examples and the respective event xml. You can explore the fields for every Sysmon event ID.

1. How many events are supported by Sysmon?
2. **(4 marks)** Provide the Event ID and briefly explanation of the following Events generated by Sysmon?
   1. A process changed a file creation time
   2. Driver Loaded
   3. Image Loaded
   4. CreateRemoteThread
   5. Process Access
   6. Registry Event (Key and Value Rename)
   7. Pipe Event (Pipe creation)
   8. DNS Event(DNS Query)
3. Install Sysmon from System Internals <https://docs.microsoft.com/en-us/sysinternals/downloads/>
4. **(2 marks)** Open cmd as Administrator and access Sysmon64.exe and use /? Or –h switch to find out the features and switches that can be used with Sysmon. Install sysmon using -i, l and n switches. What is the purpose of these switches?
5. Run services to verify if sysmon64 service is running automatically.
6. Where are Sysmon Events stored?
7. **(2 marks)** Start Event Viewer Logs and access Sysmon events. Explore friendly and XML views details to identify the following event’s details:
   1. Event 1: Task Category, ProcessGUID, ProcessID, Hashes, CommandLine,
   2. Event 4: Task Category, TimeCreated. ProcesID, ThreadID, Channel, State

C. Attach screen captures to demo the details of Event 1 and 4

1. (**1 mark**) On the right side you will find Sysmon **Log Actions**. Click on **Properties** and verify if Logging is enabled. Write down the Maximum log size:\_\_\_\_\_\_\_\_
2. **(2 marks)** Under Actions use the filter to Logged: **Last hour the events: 1,3,6 and 11** and attach the screen captures to demo results
3. **(2 marks)** Open command line (cmd) as administrator and use net command to create a new user called hacker. Access Sysmon to detect and monitor this event. Under Actions you can use filter, sort or find to search the event(s). Find this event and identify the following:
   1. Event ID
   2. Parent GUI and Parent ID
   3. CommandLine
   4. Hashes
   5. ParentCommandLine
   6. Attach a screen capture that demo details of identified event
4. **( 4 marks)** Now use WMI to initialize powershell as follows:
   1. Start cmd as administrator

Type: wmic process call create “C:\Windows\system32\Windowspowershell\v1.0\powershell.exe”

* 1. Use Sysmon to detect this event log and explain what the risk is of initializing powershell via WMI. What information from this event log you will use for analysis? Explain.

1. Sysmon requires a configuration file .xml that allows to filter the events. Microsoft has a Sysmon configuration template called sysmonconfig-export.xml in: <https://github.com/SwiftOnSecurity/sysmon-config> This template is very large and it can be customized to include and exclude rules. It follows Mitres Att&ack framework. Access <https://github.com/SwiftOnSecurity/sysmon-config> and read instructions and prerequisites to edit and update sysmonconfig-export.xml file. To understand the structure and fields purpose of this .xml file read from <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon#examples> entries and conditions implemented in Sysmon configuration file.
2. Copy sysmonconfig-export.xml to your system under sysmon directory
3. **(2 marks)** Open sysmonconfig-export.xml file and explore this large file. Read the NOTES and Technical instructions.
   1. List three things Sysmon is unable to do (do not do)
   2. What is the purpose of “Image” field?
   3. What is the purpose of processGUI field?
   4. How are lines commented?
4. **(2 marks)** One feature of Sysmon is the generation of hashes for all binaries in the system.
   1. What hashes are supported by Sysmon?
   2. What is the purpose of IMPHASH?
5. **(2 marks)** Explore Sysmon Event ID conditions. Verify what is included and what is excluded in each Sysmon Event. Explore Sysmon Event 1 read the comment and check <ProcessCreate onmatch="exclude">. Explain why all these processes are excluded.
6. **(1 mark)** Which SYSMON EVENT ID will include thread injections into Winlogon and lsass?
7. Find SYSMON EVENT ID 7 and check tag <ImageLoad onmatch="include"> read the NOTE and explain the meaning of include in this case.
8. Edit the file and modify “include” with “exclude” for SYSMON EVENT ID 7, 9, 10, 17 and 18. Read the purpose of these events
9. After modifying sysmonconfig-export.xml file save it.
10. Once the configuration file is modified it has to be updated by Sysmon. Use the respective Sysmon command to update Sysmon configuration file (sysmonconfig-export.xml). Make sure the file is validated and the configuration is updated
11. After updating configuration file check Sysmon event logs and you should see more events including 7, 9 and 10. You can Use Sysmon filter to display only event 7 or/and 9

Explore one of the logs for event 7 and one log for event 9. Are there logged events 17 or/and 18?

1. Use Windows virtual machine where you installed and configure Sysmon to perfrom the following:
   1. **(5 marks)** Implement one of the exploits ( e.g. backdoor, inject dll,) used in previous Labs to exploit Windows machine and used Sysmon to detect the events of this exploit
   2. **(5 marks)** Implement one of the post-exploits such as escalate privileges or persistence used in previous Labs and use Sysmon to detect the events of this post-exploitation
   3. Attach screen captures that demo the details of the events generated by Sysmon to detect the exploit and post-exploitation activities you implemented
   4. **(4 marks)** Explain what fields you analyzed in each event for the exploit and post exploit and provide some conclusion.
2. Windows Logging Monitor and Analysis \_\_\_30

The objectives of this exercise are:

1. to implement HELK project based on ELK stack infrastructure
2. to use group policy to enable logging
3. to use HELK to detect and analyze logs

Antivirus cannot detect all attacks therefore it is necessary to use other tools to monitor malicious activities (events) in the system. As you learnt before Sysmon can generate and log events but cannot analyze it. Generated logs do not have structure (unstructured) and systems can generate too many (big data) which makes it difficult to analyse. There are more complex systems that can be used to monitor and automatically analyse these logs. The industry implements SIEM which is very expensive. ELK Stack is an open source platform that acts as SIEM.

ELK Stack is an open source infrastructure (set of projects) created to monitor and analyze data (logs) generated by different sources and different formats. The purpose of these projects is to collect, convert, normalize, parse and visualise the logs. Once the logs have structured format can be queried from the database and visualize it for analysis. The tool offers centralized logging. This feature is helpful when attempting to identify problems with servers or applications. For details read <https://www.elastic.co/elk-stack>.

The setup and configuration of ELK-Stack infrastructure can be complex and time consuming Developers and github contributors took advantage of virtualization (containers) and created some projects that simulated the infrastructure. Some of these projects are Hunting ELK (HELK) <https://github.com/Cyb3rWard0g/HELK> and SOF-ELK <https://www.sans.org/blog/the-new-version-of-sof-elk-is-here-download-turn-on-and-get-going-on-forensics-analysis/>. All these platforms are based on ELK-Stack. In this exercise we will implement HELK

1. **( 3 marks)** Access the source <https://github.com/Cyb3rWard0g/HELK> read introduction and learn the basics and briefly explain the main purpose of the following HELK components.
   1. Beats
   2. Kafka
   3. Logstash
   4. Elasticsearch
   5. Kibana
   6. Sigma - Elastalert
2. Read installation instructions and hardware requirements. Make sure virtual Ubuntu machine has required specifications to install and run HELK. It was tested on Ubuntu 18.04 bionic. It may not work on different versions. Make sure the machine is configured at least with >=8GB of RAM and storage space >= 20GB available and two CPUs.
3. Install HELK. Click on Installation and follow the instructions to install HELK
   1. Clone HELK repo
   2. To install HELK run the script

**sudo ./helk\_install.sh**

* 1. Open a second terminal and monitor the installation process (containers installation) with:

**tail -f /var/log/helk-install.log**

* 1. During installation provide the username and password to access kibana. Take notes of **IP address** (should be the IP address of machine running HELK), **user name and password** needed later to access kibana remotely.
  2. Once HELK installation is done( around 15-30 minutes) verify containers installed and their respective usage

1. Explore HELK infrastructure. Before with connect to end point let explore the most important docker images of this infrastructure using the command:

**sudo docker exec -ti helk-logstash bash**

* 1. Remember logstash is a data processing pipeline that collects data with different format from a variety of resources, transform it and send it to elasticsearch. To explore the pipelines change to directory pipeline and explore the rules in the conf files. Lower numbers get executed first. The first one should be kafka

**bash$ hostname To access the docker container**

**bash$ cd pipeline**

**bash$ ls**

**bash$ cat 002-kafka-input.conf Check the broker name and the beat files that will be put into the pipe**

* 1. Use cat to verify the pipe for sysmon-filter and powershell-output conf files. Pay attention to connections, sessions codec, port number ( that what pipe is)
  2. Exit the current container and explore elasticalert container that contains **sigma** rules

**bash$ exit**

**sudo docker exec -ti helk-elasticalert bash**

* 1. Access sigma rules under /opt/sigma/rules

**cd /opt/sigma/rules**

**ls**

* 1. Verify the rules for **apt** and explore some of the rules. Pay attention file format is .**yml**

**cd apt**

**ls**

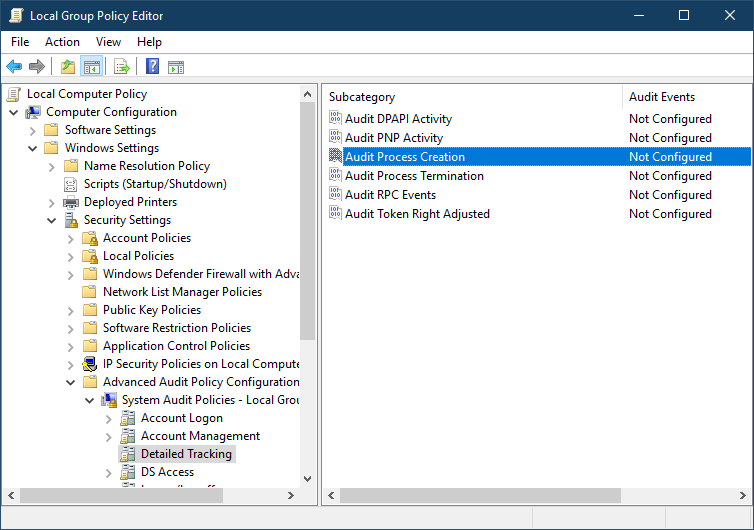
**cat apt\_cloudhopper.yml**

Pay attention to the attack type, image, Event ID, CommanLine, service

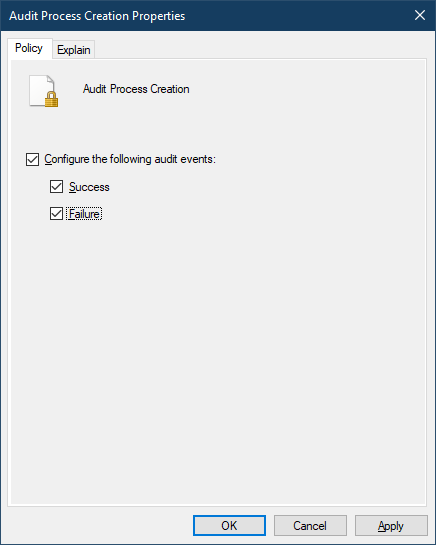
* 1. **(3 marks)** Attach screen captures of explored files and rules
  2. Verify windows rules such as powershell and sysmon
     1. **(2 marks)** Edit or cat powershell\_suspicoius\_download.yml .
        1. What is the attack type?
        2. What are the detection keywords?
     2. **(2 marks)** Edit powershell\_xor\_commanline.yml and explain briefly this rule
     3. **( 3 marks)** Now access windows/sysmon rules. Edit, explore and briefly explain the following rules:
        1. Sysmon\_mimikatz\_detection\_lsass.yml
        2. Sysmon\_mimikatz\_inmemory\_detection.yml
        3. Select two rules from windows/sysmon and explain them

1. In order to connect to **docker.sock** remotely we need to change the permission for this socket and add the current user to the group docker as follows:
   1. **cd /var/run**
   2. **ls** **docker.sock** To verify the permission of this file
   3. chmod 666 docker.sock Change permission to allow rw to user, group and other
   4. **sudo usermod –aG docker ${USER}** This will add the current user to docker group
2. **Connect and configured endpoint** (Windows client machine). We will use the Windows machine where Sysmon was installed.
   1. Make sure Ubuntu (HELK) machine can communicate with Windows 10 (endpoint).
   2. On endpoint (Windows client) we need to install and run the following services:
      1. **Sysmon**
      2. **Winlogbeat**.
   3. If you completed the previous exercise of this Lab, the virtual Windows machine should have Sysmon installed and configured. Run services command to verify if Sysmon service is running. Before configuring Winlogbeat we will enable logging using group policy object.
   4. It is important to enable Powershell scripting, command line, WMI logging because these are the Windows tools that hackers use. Most of the time hackers used command line, this is why is important to enable logging and detect command line for process creation. Windows group policy (GP) object can be used to enable these logs. Use group policy to enable command line for process creation and powershell as follows:
      1. Edit group policy: **run gpedit.msc** and click on:

Computer Configuration> Windows Settings> Security Settings > Advanced Audit Policy configuration> System Audit Policies > Detailed Tracking > Audit Process Creation

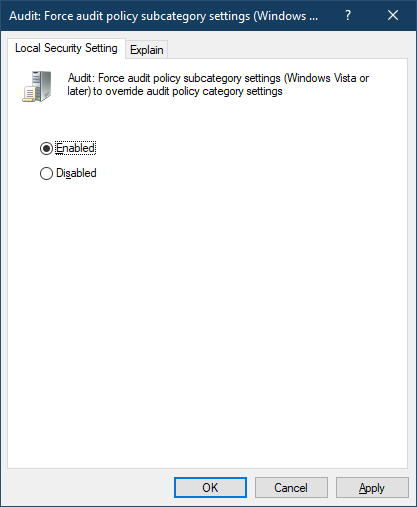


Enable Audit Process Creation. You can read the policy under “explain”



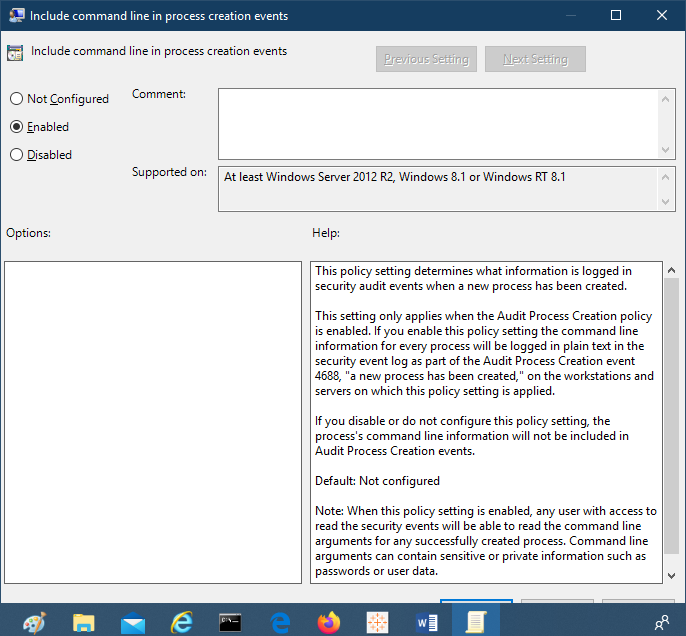
* + 1. Now enable the following Local policy:

Computer Configuration> Windows Settings> Security Settings > Local Policies> Security Options > Audit: Force Audit Policy Subcategory Settings



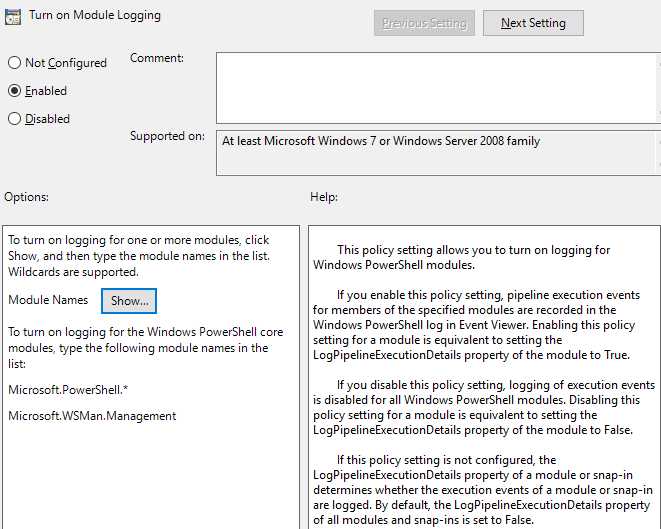
* + 1. Enable the following policy:

Computer Configuration> Administrative Templates > System > Audit Process Creation > include command line in process creation events



iv Now we can enable powershell. Click on:

Computer Configuration> Administrative Templates > Windows Components > Windows Powershell> Turn on Module Logging



Click on show to specify Module names. To include all modules type \* under Value field

* + 1. Enable the policy “ Turn on the Powershell Script Block Logging” Read the policies to understand the policy purpose
    2. Now enable Task Scheduler for logging. Access Task Scheduler and under actions select “Enable All Tasks History” if is not enabled.

1. **Download and configure Winlogbeat** on Windows client machine
   1. Download winlogbeat from: <https://www.elastic.co/downloads/beats/winlogbeat> and select Windows ZIP 64-BIT
   2. **(2 marks)** Read <https://www.elastic.co/blog/monitoring-windows-logons-with-winlogbeat> and briefly explain the purpose of this program
   3. Unzip Winlogbeat.
   4. Winlogbeat requires a configuration file (winlogbeat.yml). You can explore the default configuration file.
   5. HELK has a link to download a recommended configuration file winlogbeat.yml. Access <https://thehelk.com/installation.html> and copy Winlogbeat config file by clicking on Raw and copy it onto notepad. Before save it we need to modify this file. Browse the file at the bottom under kafka output in the line that define **hosts:** type the **IP address of your Ubuntu machine (HELK) machine**. In my case my machine is 10.0.2.15. Do not change the port. It should be 9092

#----------------------------- Kafka output --------------------------------

output.kafka:

# initial brokers for reading cluster metadata

# Place your HELK IP(s) here (keep the port).

# If you only have one Kafka instance (default for HELK) then remove the 2nd

IP that has port 9093

**hosts: ["10.0.2.15:9092"]**

topic: "winlogbeat"

############################# HELK Optimizing Latency

######################

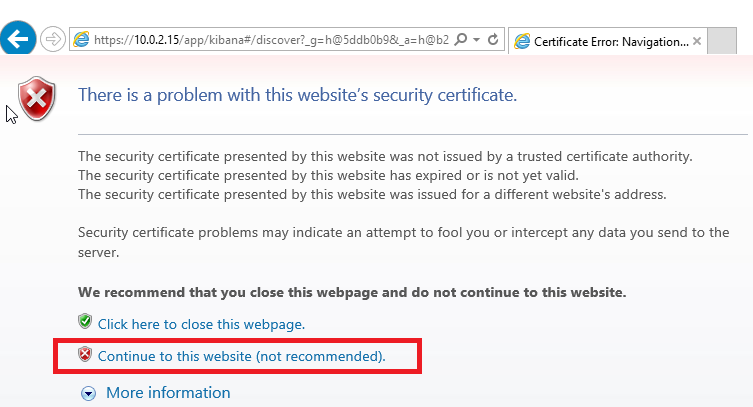
max\_retries: 2

max\_message\_bytes: 1000000

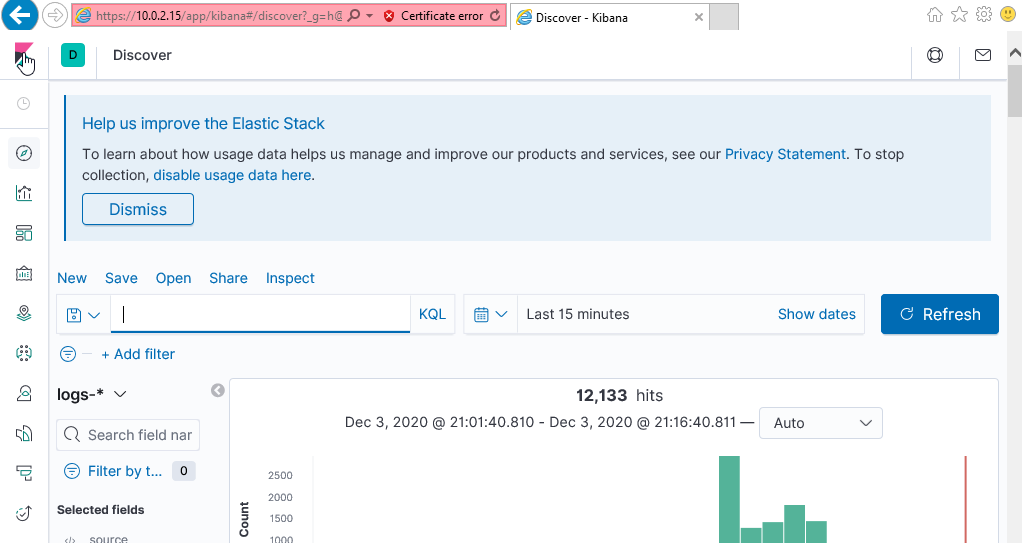
* 1. Save the file as winlogbeat.yml under the folder that has winlogbeat files
  2. Now that the configuration file is correct we can install winlogbeat service by executing **install-service-winlogbeat.ps1**. This is a powershell script then you need to run powershell first
     + - 1. Change to the directory winlogbeat and display the files in this directory. You should see installation and yml files
         2. Start poerwhell by typing: **powershell**
         3. Type: **Set-ExecutionPolicy bypass**
         4. Install the service by running the powershell:

**.\install-service-winlogbeat.ps1**

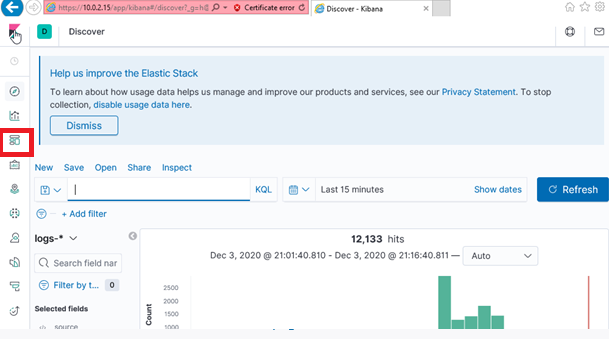
* 1. You will see **stop** but the service is installed. Run services and start the service. Make sure the service will start automatically when rebooting the machine
  2. Reboot Windows and make sure Sysmon and Winlogbeat services are running automatically before accessing Kibana.
  3. Access Kibana by typing **https://10.0.2.15**. This is the address of HELK machine
  4. It will display an error with security certificate. Ignore the error and click on continue with this web site. It should display the Login screen to access kibana



* 1. In order to access Kibana you need to provide credentials you wrote before in 4.d by default username is: helk password: hunting. After login you should have access to Kibana.



* 1. Explore Kibana start by clicking on Dashboards. Verify Sysmon Dashboard information and fields.



* 1. Click on the first icon called Discover. Identify the Search field and change it to search for **logs-endpoint-winevent-sysmon-\*** to search for command line. Now you can search for commands used before. You can search for **net or wmic** and find the event ID and the log details for the **process\_command\_line: net**
  2. You can also search or query by field’s name. Under options turn on query features then type one of the field’s name e.g event\_id: 10. This will filter it and displays only sysmon logs for event 10
  3. You can filter the logs by using AND. If you know the field name and values you want to analyse in a log you can AND it e.g query **even\_id:1 and process\_command\_line: net**. The more fields you provide the better filter and output for analysis
  4. To perform more analysis click on the date under Time column for one of the events to see all fields. You can toggle the column in a table. Select important fields for analysis such as: process name and process command line. You will see the details of these fields presented as columns in a table. You can add/remove the fields to/from the table
  5. You can also search by time. Explore this feature and display the logs generated: Today
  6. Use the respective search and query to find the logs with the details of the processes you created using command line such as: net and wmic
  7. Winlogbeat service was installed using powershell (.ps1) script. Use the respective search and query the fields that will display details of winlogbeat service running
  8. **(5 marks)** Attach screen captures that demo the logs and fields used to detect net, wmic and powershell
  9. **(5 marks) I**nject a malicious library into Windows machine and use HELK infrastructure to detect processes involved in the attack and malicious library. You can try a different attack. Attach screen captures that demos the logs you used to detect and analyze the attack
  10. **(5 marks)** Implement one of the post-exploits such as escalate privileges or persistence used in previous Labs and use HELK infrastructure to monitor and detect the events of this post-exploitation. Attach screen captures that demos the logs you used to detect and analyze the post-exploitation.

Testing one two three four five six seven eight nine ten eleven twelve thirteen fourteen fifteen sixteen seventeen eighteen nineteen twenty twenty-one twenty-two twenty-three twenty-four twenty-five twenty-six twenty-seven twenty-eight twenty-nine thirty.

.