

Kailash Parshad - HELK-Container-Security-DOCKER-Elastic-Search

GitHub: - <https://github.com/at0m-b0mb/>

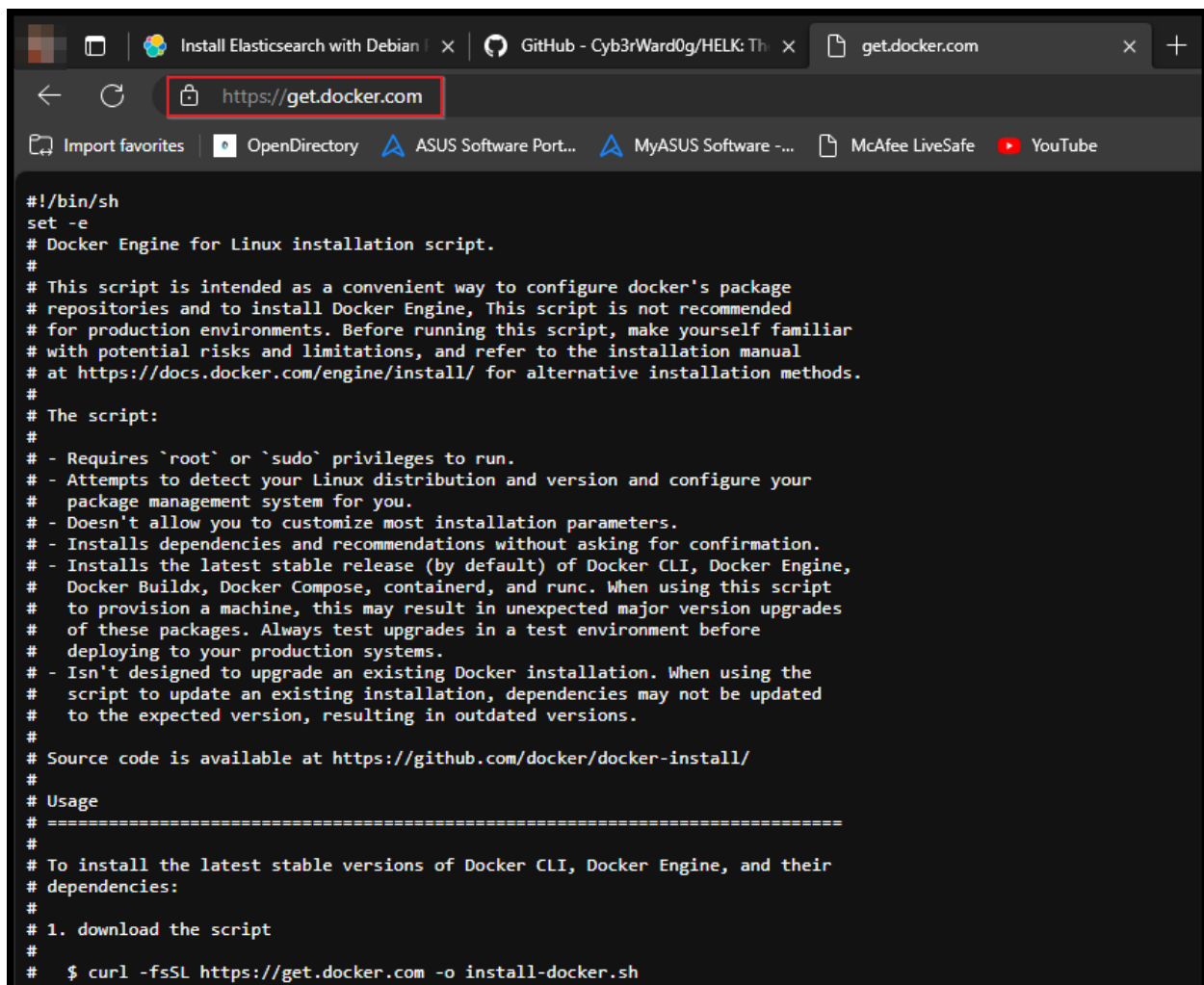
LinkedIn: - <https://www.linkedin.com/in/kailash-parshad/>

➤ Project HELK Container Security (DOCKER)

[Option 4: 8GB includes KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER + ELASTALERT]

• Installing Docker on Ubuntu: -

Link: - get.docker.com

A screenshot of a web browser window. The address bar shows 'https://get.docker.com' with a red box around it. The browser has several tabs open: 'Install Elasticsearch with Debian', 'GitHub - Cyb3rWard0g/HELK: Th', and 'get.docker.com'. The main content area displays the text of the Docker Engine for Linux installation script. The text includes a header, a disclaimer, a list of features and limitations, and usage instructions. The script is intended to be run in a terminal.

```
#!/bin/sh
set -e
# Docker Engine for Linux installation script.
#
# This script is intended as a convenient way to configure docker's package
# repositories and to install Docker Engine, This script is not recommended
# for production environments. Before running this script, make yourself familiar
# with potential risks and limitations, and refer to the installation manual
# at https://docs.docker.com/engine/install/ for alternative installation methods.
#
# The script:
#
# - Requires `root` or `sudo` privileges to run.
# - Attempts to detect your Linux distribution and version and configure your
#   package management system for you.
# - Doesn't allow you to customize most installation parameters.
# - Installs dependencies and recommendations without asking for confirmation.
# - Installs the latest stable release (by default) of Docker CLI, Docker Engine,
#   Docker Buildx, Docker Compose, containerd, and runc. When using this script
#   to provision a machine, this may result in unexpected major version upgrades
#   of these packages. Always test upgrades in a test environment before
#   deploying to your production systems.
# - Isn't designed to upgrade an existing Docker installation. When using the
#   script to update an existing installation, dependencies may not be updated
#   to the expected version, resulting in outdated versions.
#
# Source code is available at https://github.com/docker/docker-install/
#
# Usage
# =====
#
# To install the latest stable versions of Docker CLI, Docker Engine, and their
# dependencies:
#
# 1. download the script
#
# $ curl -fsSL https://get.docker.com -o install-docker.sh
```

- ***Copying the raw code and making a new Shell script: -***

```

root@b0mb: /home/at0m/HELK
GNU nano 6.2                                docker_Install.sh
#!/bin/sh
set -e
# Docker Engine for Linux installation script.
#
# This script is intended as a convenient way to configure docker's package
# repositories and to install Docker Engine, This script is not recommended
# for production environments. Before running this script, make yourself famili>
# with potential risks and limitations, and refer to the installation manual
# at https://docs.docker.com/engine/install/ for alternative installation metho>
#
# The script:
#
# - Requires `root` or `sudo` privileges to run.
# - Attempts to detect your Linux distribution and version and configure your
#   package management system for you.
# - Doesn't allow you to customize most installation parameters.
# - Installs dependencies and recommendations without asking for confirmation.
# - Installs the latest stable release (by default) of Docker CLI, Docker Engin>
#   Docker Buildx, Docker Compose, containerd, and runc. When using this script
#   to provision a machine, this may result in unexpected major version upgrades
#
[ Read 743 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line

```

- ***Giving Execution Rights and running the file (ROOT Privileges required): -***

```

root@b0mb:/home/at0m/HELK# chmod +x docker_Install.sh
root@b0mb:/home/at0m/HELK# ./docker_Install.sh
# Executing docker install script, commit: e5543d473431b782227f8908005543bb4389b
8de
+ sh -c apt-get update -qq >/dev/null
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq apt-transport-http
s ca-certificates curl >/dev/null
+ sh -c install -m 0755 -d /etc/apt/keyrings
+ sh -c curl -fsSL "https://download.docker.com/linux/ubuntu/gpg" | gpg --dearm
r --yes -o /etc/apt/keyrings/docker.gpg
+ sh -c chmod a+r /etc/apt/keyrings/docker.gpg
+ sh -c echo "deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.gpg] https://do
wnload.docker.com/linux/ubuntu jammy stable" > /etc/apt/sources.list.d/docker.li
st
+ sh -c apt-get update -qq >/dev/null
+ sh -c DEBIAN_FRONTEND=noninteractive apt-get install -y -qq docker-ce docker-c
e-cli containerd.io docker-compose-plugin docker-ce-rootless-extras docker-build
x-plugin >/dev/null

```

- ***Cloning the GIT repository: -***

Link: - git clone <https://github.com/Cyb3rWard0g/HELK.git>

```
root@b0mb:/home/at0m/HELK# git clone https://github.com/Cyb3rWard0g/HELK.git
Cloning into 'HELK'...
remote: Enumerating objects: 10109, done.
remote: Counting objects: 100% (49/49), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 10109 (delta 23), reused 37 (delta 16), pack-reused 10060
Receiving objects: 100% (10109/10109), 852.60 MiB | 2.01 MiB/s, done.
Resolving deltas: 100% (6948/6948), done.
root@b0mb:/home/at0m/HELK#
```

- **Running the install script: -**

```
root@b0mb:/home/at0m/HELK/HELK/docker# sudo ./helk_install.sh

*****
**          HELK - THE HUNTING ELK          **
**                                          **
** Author: Roberto Rodriguez (@Cyb3rWard0g) **
** HELK build version: v0.1.9-alpha10082020 **
** HELK ELK version: 7.6.2          **
** License: GPL-3.0                  **
*****

[HELK-INSTALLATION-INFO] HELK hosted on a Linux box
[HELK-INSTALLATION-INFO] Available Memory: 8479 MBs
[HELK-INSTALLATION-INFO] You're using ubuntu version jammy

*****
*          HELK - Docker Compose Build Choices          *
*****

1. KAFKA + KSQL + ELK + NGINX
2. KAFKA + KSQL + ELK + NGINX + ELASTALERT
3. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER
4. KAFKA + KSQL + ELK + NGINX + SPARK + JUPYTER + ELASTALERT
```

```
root@b0mb: /home/at0m/HELK/HELK/docker
6.175
[HELK-INSTALLATION-INFO] HELK IP set to 192.168.6.175
[HELK-INSTALLATION-INFO] Please make sure to create a custom Kibana password and
  store it securely for future use.
[HELK-INSTALLATION-INFO] Set HELK Kibana UI Password: hunting
[HELK-INSTALLATION-INFO] HELK Kibana UI password set to hunting
[HELK-INSTALLATION-INFO] Installing htpasswd..
[HELK-INSTALLATION-INFO] Docker already installed
[HELK-INSTALLATION-INFO] Checking if it is installed via snap..
[HELK-INSTALLATION-INFO] Snap v2.59.5 is available
[HELK-INSTALLATION-INFO] Docker not installed via snap
[HELK-INSTALLATION-INFO] Assesing if Docker is running..
[HELK-INSTALLATION-INFO] Docker is running
[HELK-INSTALLATION-INFO] Making sure you assigned enough disk space to the curre
nt Docker base directory
[HELK-INSTALLATION-INFO] Available Docker Disk: 125 GBs
[HELK-INSTALLATION-INFO] Installing docker-compose..
[HELK-INSTALLATION-INFO] Checking local vm.max_map_count variable and setting it
  to 4120294
[HELK-INSTALLATION-INFO] Setting local vm.swappiness variable to 25
[HELK-INSTALLATION-INFO] Building & running HELK from helk-kibana-notebook-analy
sis-alert-basic.yml file..
```

- Let's us check the installation logs: -

```
at0m@b0mb: ~
at0m@b0mb:~$ tail -f /var/log/helk-install.log
Pulling helk-elasticsearch (docker.elastic.co/elasticsearch/elasticsearch:7.6.2)
...
7.6.2: Pulling from elasticsearch/elasticsearch
Digest: sha256:59342c577e2b7082b819654d119f42514ddf47f0699c8b54dc1f0150250ce7aa
Status: Downloaded newer image for docker.elastic.co/elasticsearch/elasticsearch:7.6.2
Pulling helk-kibana (docker.elastic.co/kibana/kibana:7.6.2)...
7.6.2: Pulling from kibana/kibana
Digest: sha256:e8f3743e404462709663422056db2d5076a7a6bd6024f64aea1599b3014c63be
Status: Downloaded newer image for docker.elastic.co/kibana/kibana:7.6.2
Pulling helk-logstash (otrf/helk-logstash:7.6.2.1)...
7.6.2.1: Pulling from otrf/helk-logstash
Digest: sha256:b1135da506f40fc1d5861db7ba844486f3a08a57af3fdb8e301ab487f51a2ac1
Status: Downloaded newer image for otrf/helk-logstash:7.6.2.1
Pulling helk-nginx (otrf/helk-nginx:0.3.0)...
0.3.0: Pulling from otrf/helk-nginx
Digest: sha256:32eb6e39681849dc3bed36cfb95bd39b25f8c66d08965b6855f64eb2ee0668ba
Status: Downloaded newer image for otrf/helk-nginx:0.3.0
Pulling helk-zookeeper (otrf/helk-zookeeper:2.4.0)...
2.4.0: Pulling from otrf/helk-zookeeper
Digest: sha256:d8a7c57c03384f5ce2b6125505c1f8e2a020432de81bde3677fcc8009fc5cfd2
Status: Downloaded newer image for otrf/helk-zookeeper:2.4.0
Pulling helk-kafka-broker (otrf/helk-kafka-broker:2.4.0)...
```

```
2.4.5: Pulling from otrf/helk-spark-master
Digest: sha256:1c3589bf181e5302153480b38e4e675afd1a29ef5d3fc6e31d9a33a566b95f18
Status: Downloaded newer image for otrf/helk-spark-master:2.4.5
Pulling helm-spark-worker (otrf/helm-spark-worker:2.4.5)...
2.4.5: Pulling from otrf/helm-spark-worker
Digest: sha256:0c3e2f759d6f286dbf740dab6a74740eb1b173d41156d50c3e4a32ea7e5aa74c
Status: Downloaded newer image for otrf/helm-spark-worker:2.4.5
Pulling helm-elastalert (otrf/helm-elastalert:latest)...
latest: Pulling from otrf/helm-elastalert
Digest: sha256:689fba01b8b238c7a5a0e41b20f1990318c74c0102c6178189baa28037c5c8a7
Status: Downloaded newer image for otrf/helm-elastalert:latest
Creating helm-elasticsearch ... done
Creating helm-kibana ... done
Creating helm-nginx ... done
Creating helm-logstash ... done
Creating helm-zookeeper ... done
Creating helm-elastalert ... done
Creating helm-spark-master ... done
Creating helm-jupyter ... done
Creating helm-spark-worker ... done
Creating helm-kafka-broker ... done
Creating helm-ksql-server ... done
Creating helm-ksql-cli ... done
```

- **HELK is successfully installed: -**

HELK KIBANA URL: <https://192.168.6.175>

HELK KIBANA USER: helm

HELK KIBANA PASSWORD: hunting

HELK SPARK MASTER UI: <https://192.168.6.175:8080>

HELK JUPYTER SERVER URL: <https://192.168.6.175/jupyter>

HELK JUPYTER CURRENT TOKEN:

a560e916917dd9d3a4a7414fa12a20b53a3aa85021a104e3

HELK ZOOKEEPER: 192.168.6.175:2181

HELK KSQL SERVER: 192.168.6.175:8088


```
*****
****
** [HELK-INSTALLATION-INFO] HELK WAS INSTALLED SUCCESSFULLY
**
** [HELK-INSTALLATION-INFO] USE THE FOLLOWING SETTINGS TO INTERACT WITH THE HELK
**
*****
****
```

```
HELK KIBANA URL: https://192.168.6.175
HELK KIBANA USER: helk
HELK KIBANA PASSWORD: hunting
HELK SPARK MASTER UI: https://192.168.6.175:8080
HELK JUPYTER SERVER URL: https://192.168.6.175/jupyter
HELK JUPYTER CURRENT TOKEN: a560e916917dd9d3a4a7414fa12a20b53a3aa85021a104e3
HELK ZOOKEEPER: 192.168.6.175:2181
HELK KSQL SERVER: 192.168.6.175:8088
```

IT IS HUNTING SEASON!!!!

You can stop all the HELK docker containers by running the following command:
 [+] sudo docker-compose -f helk-kibana-notebook-analysis-alert-basic.yml stop

root@b0mb:/home/at0m/HELK/HELK/docker#

- **Checking all the Running Container: -**

```
root@b0mb:/home/at0m/HELK/HELK/docker# sudo docker ps
```

| CONTAINER ID | IMAGE | COMMAND | CREATED |
|---------------|---|--------------------------|----------------|
| STATUS | PORTS | | |
| d15938e325c8 | confluentinc/ksqldb-cli:latest | "/bin/sh" | 53 minutes ago |
| Up 53 minutes | | | |
| fdf7c1418dc8 | confluentinc/ksqldb-server:latest | "/usr/bin/docker/run" | 53 minutes ago |
| Up 53 minutes | 0.0.0.0:8088->8088/tcp, :::8088->8088/tcp | | |
| 66e2ba3a89be | otrf/helk-kafka-broker:2.4.0 | "/kafka-entrypoint..." | 53 minutes ago |
| Up 53 minutes | 0.0.0.0:9092->9092/tcp, :::9092->9092/tcp | | |
| 14ff365cb065 | otrf/helk-spark-worker:2.4.5 | "/spark-worker-entr..." | 53 minutes ago |
| Up 53 minutes | | | |
| 42cd8605bc4a | otrf/helk-elastalert:latest | "/elastalert-entryp..." | 53 minutes ago |
| Up 53 minutes | | | |
| 8d017305e737 | docker_helk-jupyter | "/opt/jupyter/script..." | 53 minutes ago |
| Up 53 minutes | 8000/tcp, 8888/tcp | | |

- **Checking all the Resources Utilized: -**

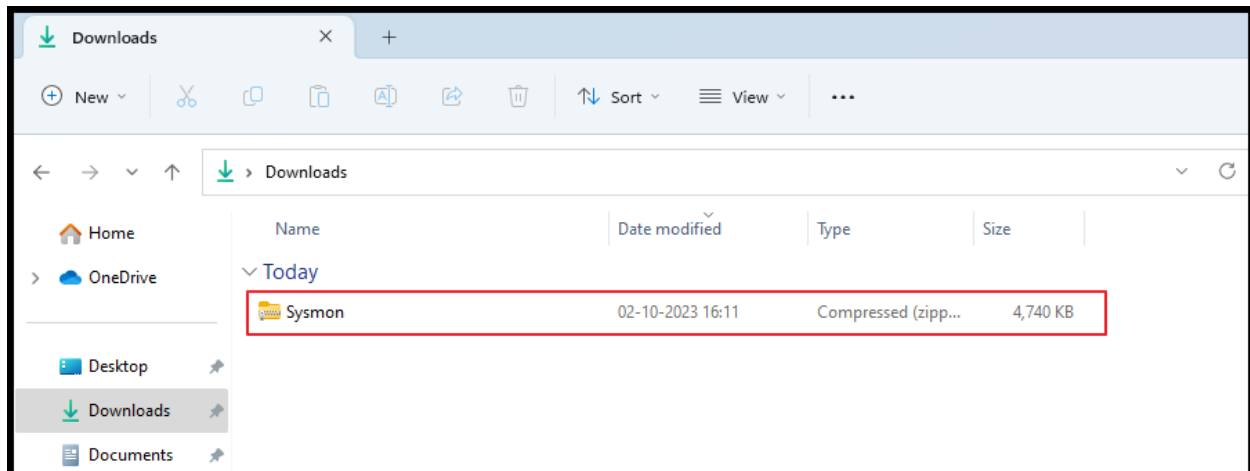
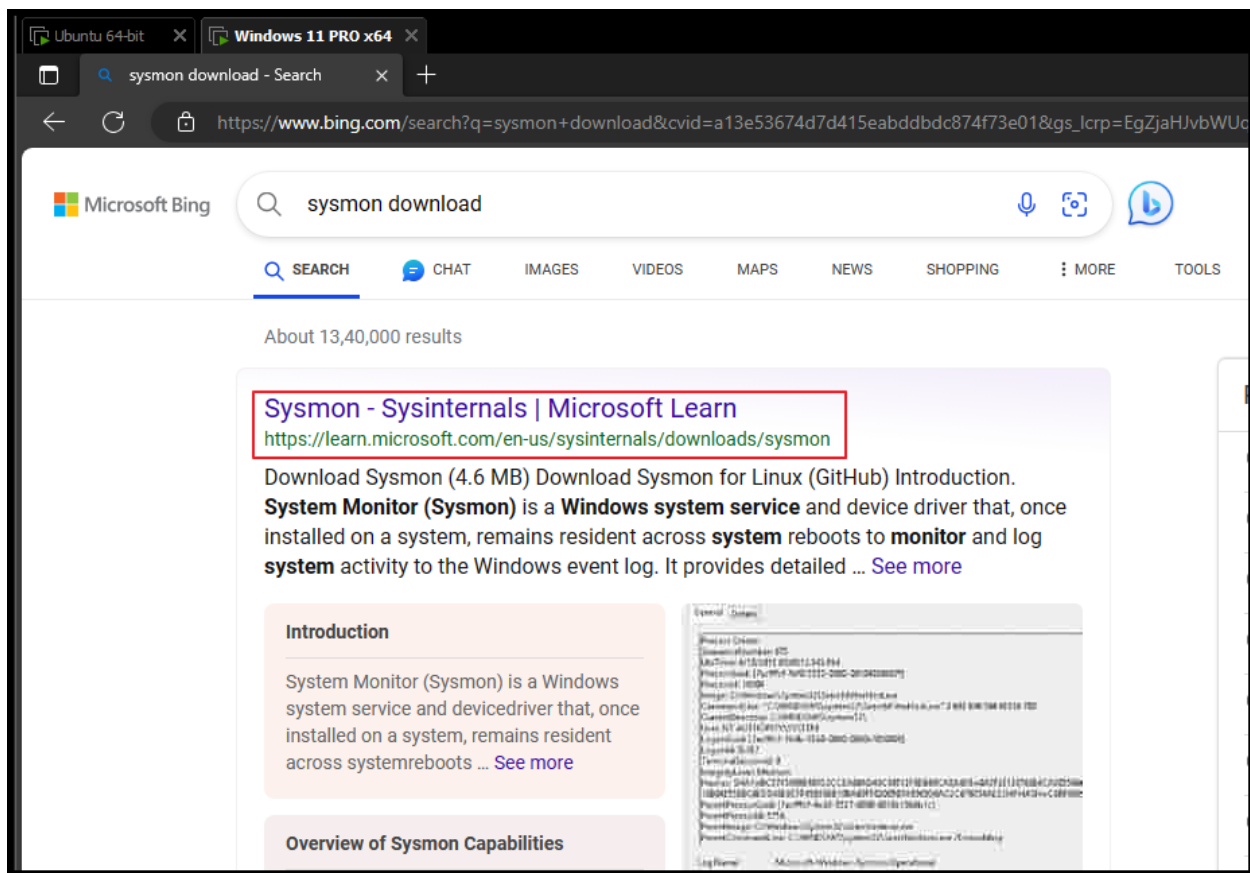
```
at0m@b0mb:~$ sudo docker stats --all
[sudo] password for at0m:

CONTAINER ID   NAME                CPU %     MEM USAGE / LIMIT   MEM %     NET I/O       BLOCK I/O     PIDS
d15938e325c8   helm-ksql-cli       0.00%     940KiB / 9.623GiB   0.01%     5.62kB / 0B   49.2kB / 0B   1
fdf7c1418dc8   helm-ksql-server    0.72%     406.5MiB / 9.623GiB 4.13%     1.33MB / 1.12MB 180MB / 5.67MB 39
66e2ba3a89be   helm-kafka-broker   1.55%     350.3MiB / 9.623GiB 3.55%     5.76MB / 5.58MB 26MB / 13.4MB  76
14ff365cb065   helm-spark-worker    0.22%     179.8MiB / 9.623GiB 1.82%     42.7kB / 318kB  1.74MB / 7.16MB 33
42cd8605bc4a   helm-elastalert      13.85%    81.23MiB / 9.623GiB 0.82%     42.4MB / 68.5MB 15.7MB / 7.91MB 12
8d017305e737   helm-jupyter         0.12%     80.29MiB / 9.623GiB 0.81%     5.77kB / 0B    75.8MB / 62.8MB 9
5f0be8c4b73d   helm-zookeeper       0.24%     91.22MiB / 9.623GiB 0.93%     355kB / 262kB   40.3MB / 9.73MB 48
ac5e087e9d89   helm-spark-master    0.24%     197.5MiB / 9.623GiB 2.00%     390kB / 543kB   10.6MB / 5.19MB 33
7ec3afb4ad94   helm-logstash        11.16%    1.161GiB / 9.623GiB 12.07%    3.14MB / 189MB  168MB / 4.88MB 105
e2c92f97dc8c   helm-nginx           0.00%     5.488MiB / 9.623GiB 0.06%     30.9MB / 31.3MB 19.5MB / 143kB  4
fd07d46ffed7   helm-kibana          0.69%     338.4MiB / 9.623GiB 3.43%     3.56MB / 35.7MB 250MB / 136MB  13
2e5c1aa14c2e   helm-elasticsearch   6.10%     3.593GiB / 9.623GiB 37.34%    260MB / 45.6MB 262MB / 748MB  81
```

- **Checking the Docker Containers' Logs**

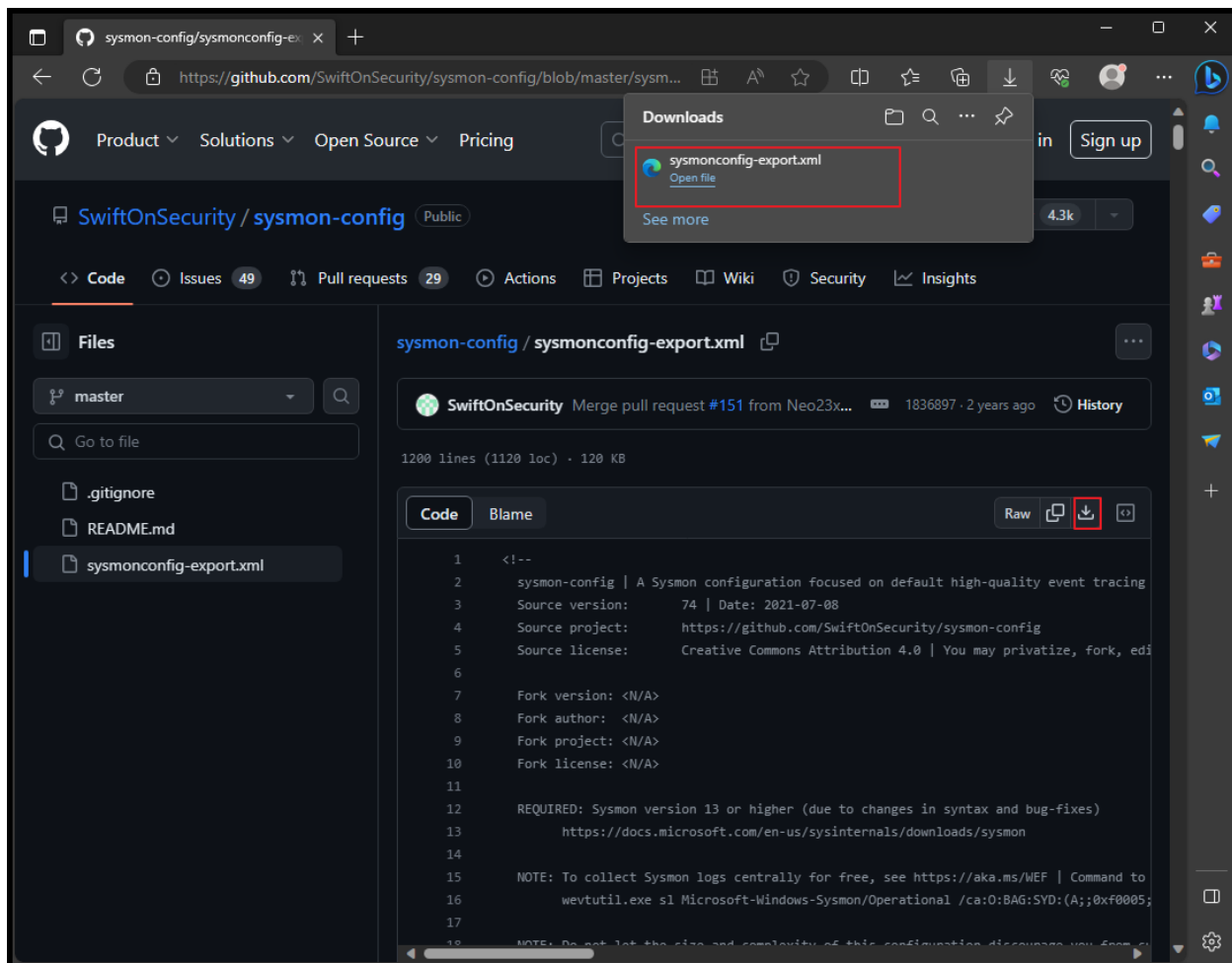
```
at0m@b0mb:~$ sudo docker logs --follow --tail 20 helm-elasticsearch
"at org.elasticsearch.index.mapper.DocumentMapper.merge(DocumentMapper.java:321) ~[elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.metadata.MetadataMappingService$PutMappingExecutor.applyRequest(MetadataMappingService.java:282) ~[elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.metadata.MetadataMappingService$PutMappingExecutor.execute(MetadataMappingService.java:238) ~[elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.MasterService.executeTasks(MasterService.java:702) ~[elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.MasterService.calculateTaskOutputs(MasterService.java:324) ~[elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.MasterService.runTasks(MasterService.java:219) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.MasterService.access$000(MasterService.java:73) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.MasterService$Batcher.run(MasterService.java:151) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.TaskBatcher.runIfNotProcessed(TaskBatcher.java:150) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.cluster.service.TaskBatcher$BatchedTask.run(TaskBatcher.java:188) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.common.util.concurrent.ThreadContext$ContextPreservingRunnable.run(ThreadContext.java:633) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.common.util.concurrent.PrioritizedEsThreadPoolExecutor$TieBreakingPrioritizedRunnable.runAndClean(PrioritizedEsThreadPoolExecutor.java:252) [elasticsearch-7.6.2.jar:7.6.2]",
"at org.elasticsearch.common.util.concurrent.PrioritizedEsThreadPoolExecutor$TieBreakingPrioritizedRunnable.run(PrioritizedEsThreadPoolExecutor.java:215) [elasticsearch-7.6.2.jar:7.6.2]",
"at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1128) [?:?]",
"at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:628) [?:?]",
"at java.lang.Thread.run(Thread.java:830) [?:?]" }
{"type": "server", "timestamp": "2023-10-02T09:41:48,163Z", "level": "INFO", "component": "o.e.c.m.MetadataMappingService", "cluster.name": "helm-cluster", "node.name": "helm-1", "message": "[elastalert status_status/5X9HgQHxTICeU99q5MH-XA] update_mapping [doc]", "cluster.uuid": "24wAai_CTQSLSCAg0xcXFQ", "node.id": "wUtMA2BGRsKNFsiLYaEaqw" }
{"type": "deprecation", "timestamp": "2023-10-02T09:41:48,590Z", "level": "WARN", "component": "o.e.d.r.a.s.RestSearchAction", "cluster.name": "helm-cluster", "node.name": "helm-1", "message": "[types removal] Specifying types in search requests is deprecated.", "cluster.uuid": "24wAai_CTQSLSCAg0xcXFQ", "node.id": "wUtMA2BGRsKNFsiLYaEaqw" }
{"type": "deprecation", "timestamp": "2023-10-02T10:00:08,480Z", "level": "WARN", "component": "o.e.d.i.q.QueryShardContext", "cluster.name": "helm-cluster", "node.name": "helm-1", "message": "[types removal] Using the _type field in queries and aggregations is deprecated, prefer to use a field instead.", "cluster.uuid": "24wAai_CTQSLSCAg0xcXFQ", "node.id": "wUtMA2BGRsKNFsiLYaEaqw" }
{"type": "deprecation", "timestamp": "2023-10-02T10:17:32,526Z", "level": "WARN", "component": "o.e.d.r.a.a.i.RestGetMappingAction", "cluster.name": "helm-cluster", "node.name": "helm-1", "message": "[types removal] Using include_type_name in get mapping requests is deprecated. The parameter will be removed in the next major version.", "cluster.uuid": "24wAai_CTQSLSCAg0xcXFQ", "node.id": "wUtMA2BGRsKNFsiLYaEaqw" }
```

- **Now install Sysmon on a Windows computer: -**



- ***Now configure Sysmon with Github's xml config: -***

Link: [-Github - SwiftOnSecurity/sysmon-config: Sysmon configuration file template with default high-quality event tracing](#)

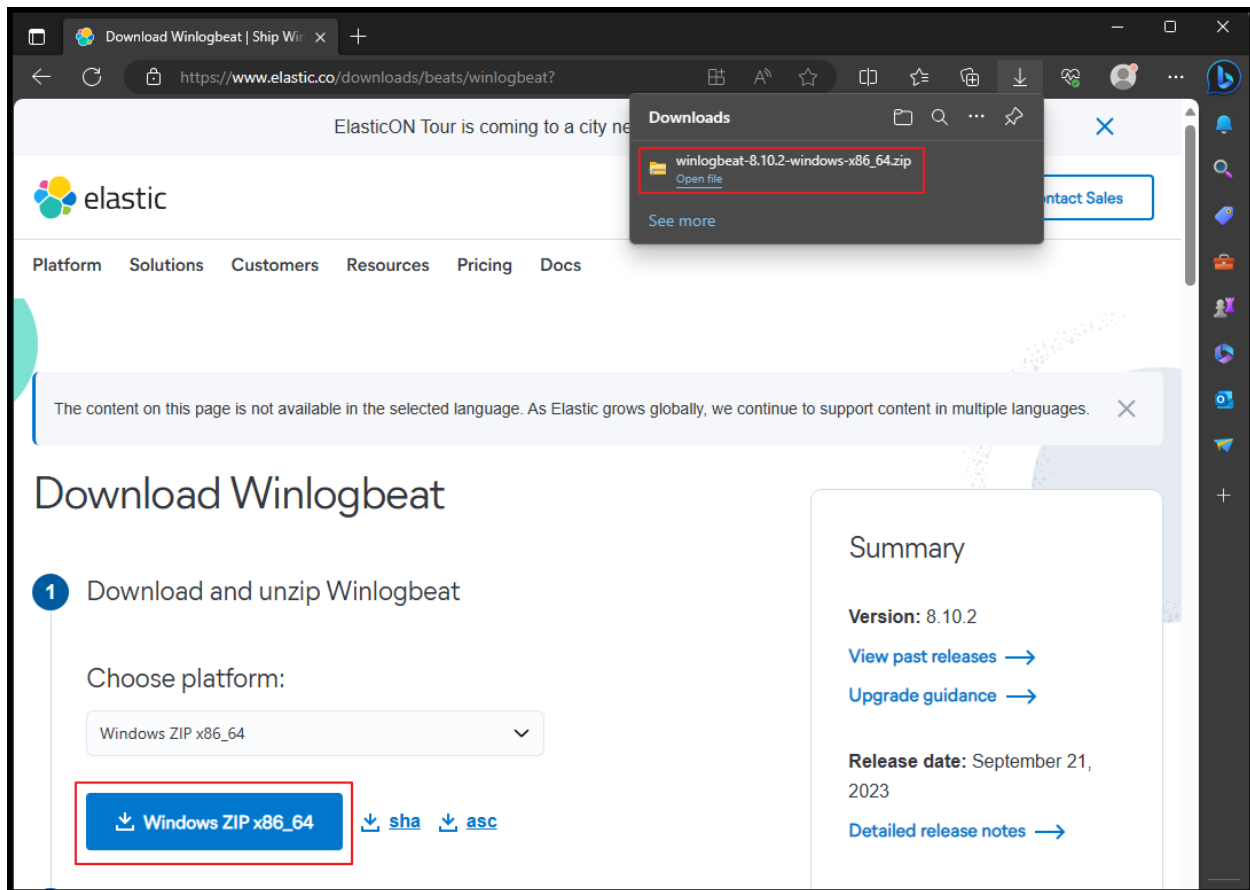


```
C:\Tools\Sysmon>.\Sysmon.exe -i .\sysmonconfig-export.xml

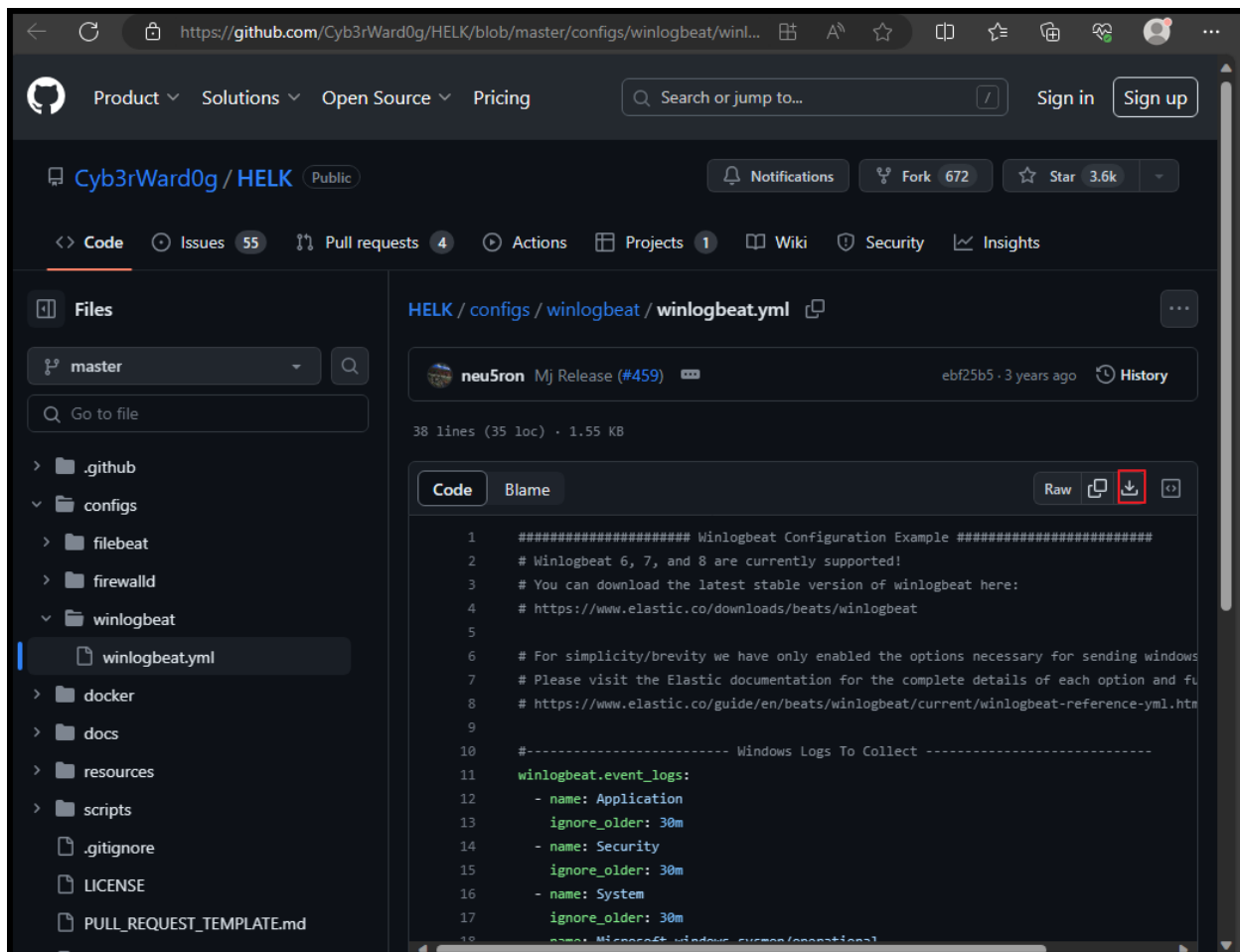
System Monitor v15.0 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2023 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.50
Sysmon schema version: 4.90
Configuration file validated.
Sysmon installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon..
Sysmon started.
```

- **Downloading and Installing Winlogbeat:-**



- **Installing Winlogbeat.yml file from GitHub: -**



```
#----- 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: ["192.168.6.175:9092"]
  topic: "winlogbeat"
##### HELK Optimizing Latency #####
max_retries: 2
max_message_bytes: 1000000
```

- ***Now installing Winlogbeat as a Service: -***

```
Administrator: Command Prompt - powershell
PS C:\Tools\winlogbeat-8.10.2-windows-x86_64> Set-ExecutionPolicy Unrestricted
PS C:\Tools\winlogbeat-8.10.2-windows-x86_64> .\install-service-winlogbeat.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful, this script can potentially harm your
computer. If you trust this script, use the Unblock-File cmdlet to allow the script to run without this warning
message. Do you want to run C:\Tools\winlogbeat-8.10.2-windows-x86_64\install-service-winlogbeat.ps1?
[D] Do not run [R] Run once [S] Suspend [?] Help (default is "D"): R

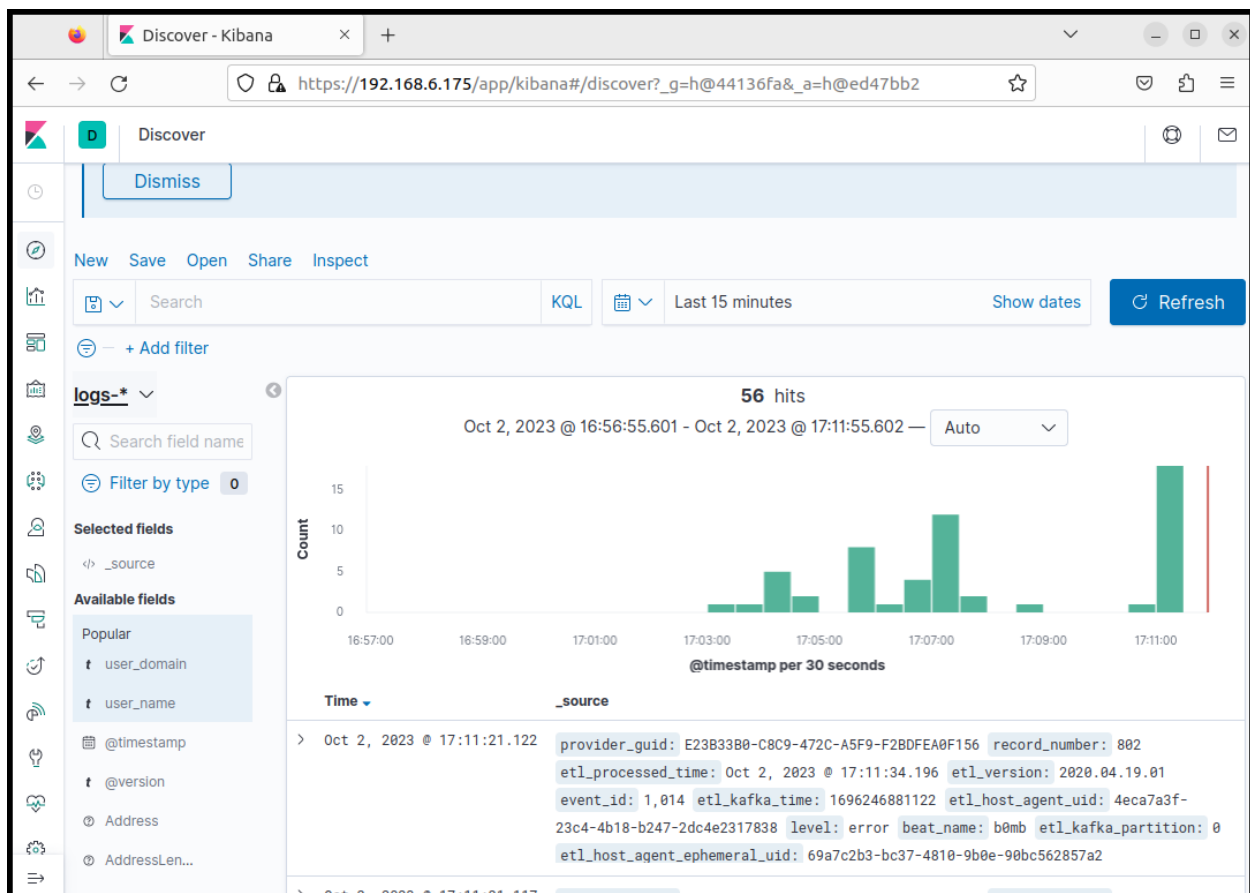
Status      Name      DisplayName
-----
Stopped     winlogbeat  winlogbeat

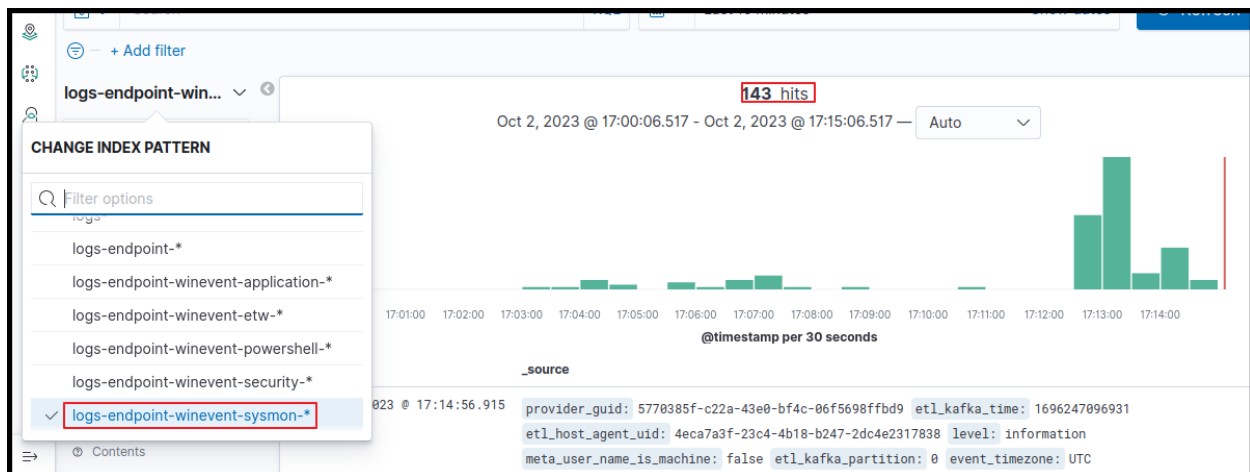
PS C:\Tools\winlogbeat-8.10.2-windows-x86_64>
```

- Starting the winlogbeat service: -

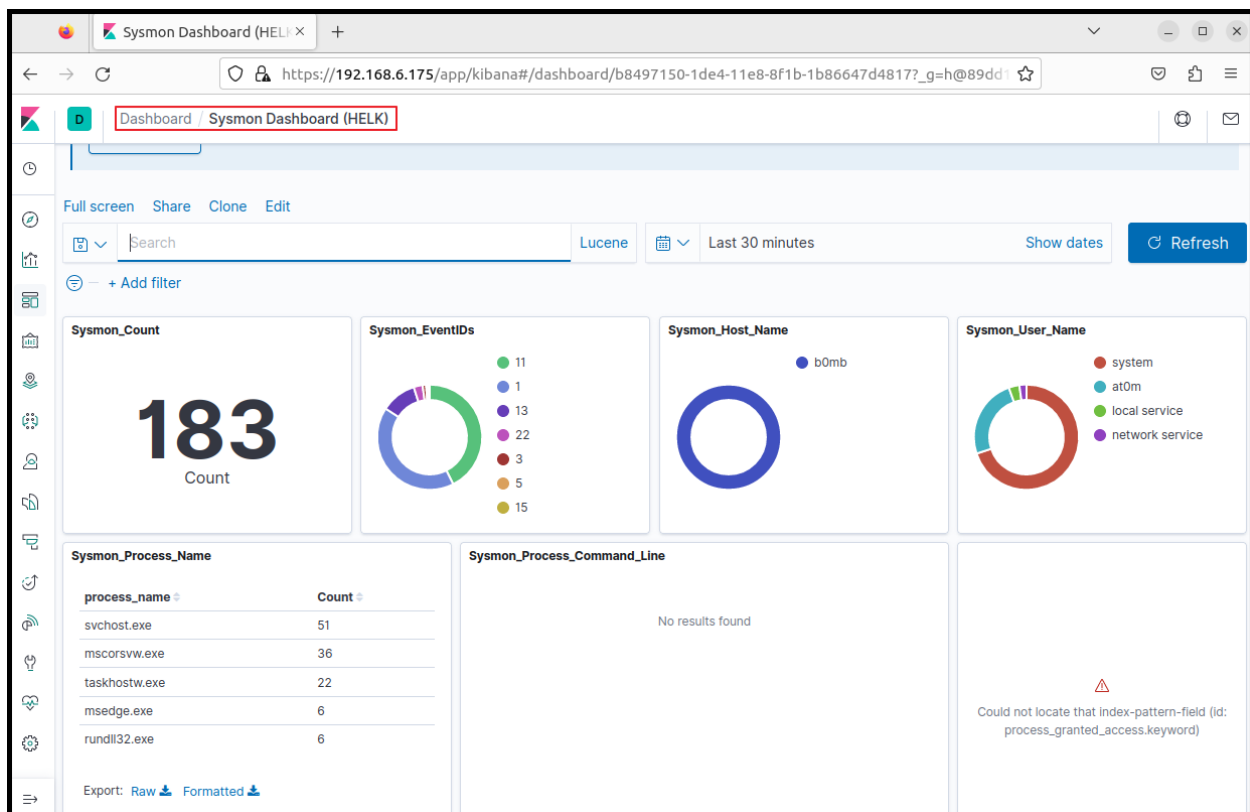
```
Administrator: Command Prompt - powershell
PS C:\Windows\System32> start-service winlogbeat
PS C:\Windows\System32>
```

- Now we will check Kibana for checking the logs: -





- **Checking Custom HELK Sysmon Dashboard :-**



Setup was complete and successful with ELK stack!

- **Now we will check the SPARK MASTER UI: -**

The screenshot shows the Spark Master web interface at `spark://ac5e087e9d89:7077`. The interface displays the following information:

- URL:** `spark://ac5e087e9d89:7077`
- Alive Workers:** 1
- Cores in use:** 4 Total, 0 Used
- Memory in use:** 1024.0 MB Total, 0.0 B Used
- Applications:** 0 Running, 0 Completed
- Drivers:** 0 Running, 0 Completed
- Status:** ALIVE

Under the **Workers (1)** section, a table lists the worker details:

| Worker Id | Address | State | Cores | Memory |
|---|-------------------|-------|------------|------------------------|
| worker-20231002092525-172.18.0.10-42950 | 172.18.0.10:42950 | ALIVE | 4 (0 Used) | 1024.0 MB (0.0 B Used) |

Below the workers section, there are two empty tables for **Running Applications (0)** and **Completed Applications (0)**, both with columns: Application ID, Name, Cores, Memory per Executor, Submitted Time, User, State, and Duration.

- **Checking Jupiter Notebook: -**

The screenshot shows the Jupyter Notebook web interface at `https://192.168.6.175/jupyter/tree/`. The interface displays the following information:

- Files** (selected), **Running**, **Clusters**
- Select items to perform actions on them.**
- Upload**, **New**, **Refresh**
- File Browser:**
 - Name:** datasets, demos, sigma, tutorials
 - Last Modified:** 3 hours ago
 - File size:**

THANKYOU