

Kickstart 2: SIEM and Log Analysis – SELKS

This material is prepared for use in *SIEM and Log Analysis course* and was prepared by Henrik Kramselund Jereminsen, xhek@kea.dk hkj@zencurity.dk. It contains the very basic information to get started!

The course had some problems with Elastic stack version 8 – which is updated on multiple fronts, like HTTPS/TLS. This is giving us a lot of headache. The students meet with different obstacles, so this kickstart 2 document is a way out!

I would like for you to install Docker and try out SELKS <https://www.stamus-networks.com/selks>

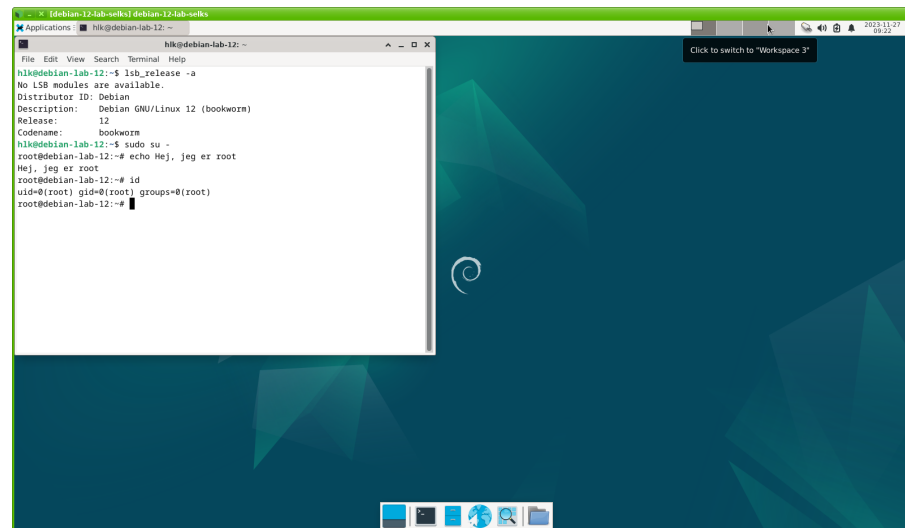
If you want to use the same as me with Debian VM, which installs in less than 30minutes:

- ☐ Install a basic Debian 12 Bookworm with Sudo configured
- ☐ Install git and Ansible, see our exercise:
`sudo apt install git ansible`
- ☐ Clone the Github repo: <https://github.com/kramse/kramse-labs>
`git clone https://github.com/kramse/kramse-labs`
- ☐ Go into this repository and install Docker, there is a small README.md too:
`cd kramse-labs/docker-install` and then `ansible-playbook 1-dependencies.yml`
- ☐ Enable Docker: `systemctl enable docker` and reboot the VM
- ☐ Check docker, `docker run hello-world`
- ☐ Clone the SELKS repository:
`git clone https://github.com/StamusNetworks/SELKS.git`
- ☐ Go into this and run docker-compose as described in the instructions:
<https://github.com/StamusNetworks/SELKS/wiki/Docker>
make sure to select the right network interface, so Suricata can sniff packets I did NOT install Portainer
- ☐ Use a browser to access the platform on `https://127.0.0.1`
- ☐ Relax

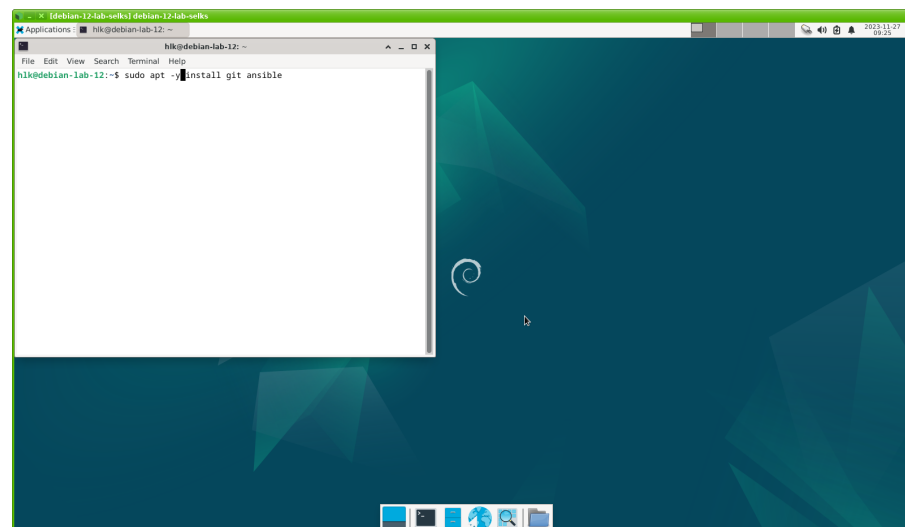
This will provide a basic Elasticsearch version 7, with Kibana and Suricata

Kickstart 2: SIEM and Log Analysis – SELKS

Basic Debian with Sudo:

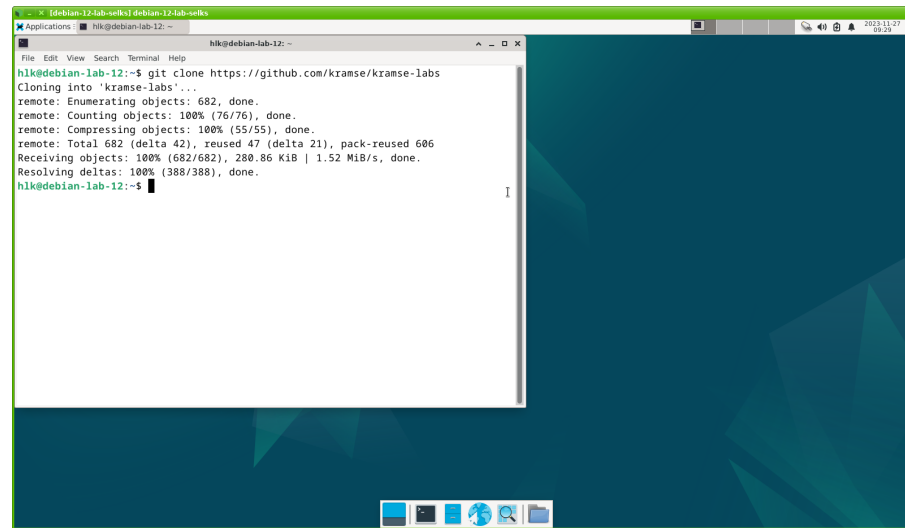


Install git and ansible:



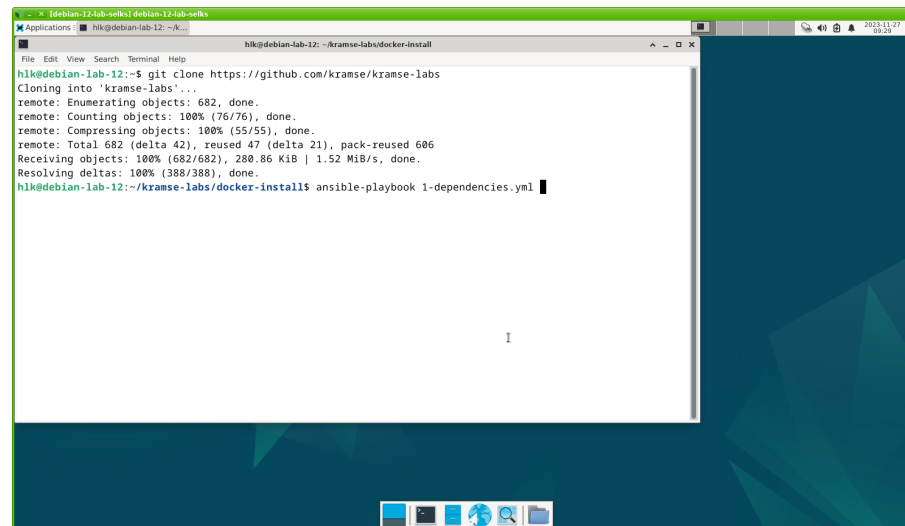
Kickstart 2: SIEM and Log Analysis – SELKS

Git clone kramse-labs:

A terminal window titled 'hik@debian-lab-12: ~' is open on a desktop with a blue geometric wallpaper. The terminal shows the output of a 'git clone' command. The output includes progress bars for enumerating, counting, and compressing objects, and a final message indicating the repository was cloned successfully. The prompt is 'hik@debian-lab-12:~\$'.

```
hik@debian-lab-12:~$ git clone https://github.com/kramse/kramse-labs
Cloning into 'kramse-labs'...
remote: Enumerating objects: 682, done.
remote: Counting objects: 100% (76/76), done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 682 (delta 42), reused 47 (delta 21), pack-reused 606
Receiving objects: 100% (682/682), 280.86 KiB | 1.52 MiB/s, done.
Resolving deltas: 100% (388/388), done.
hik@debian-lab-12:~$
```

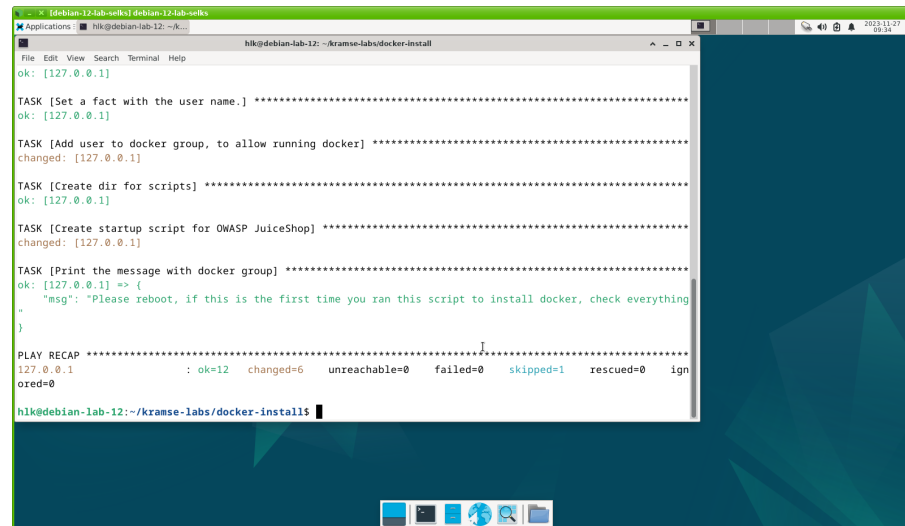
Use Ansible to install Docker:

A terminal window titled 'hik@debian-lab-12: ~' is open on a desktop with a blue geometric wallpaper. The terminal shows the output of a 'git clone' command, followed by the execution of an Ansible playbook. The output of the playbook is 'ansible-playbook 1-dependencies.yml'. The prompt is 'hik@debian-lab-12:~/kramse-labs/docker-install\$'.

```
hik@debian-lab-12:~$ git clone https://github.com/kramse/kramse-labs
Cloning into 'kramse-labs'...
remote: Enumerating objects: 682, done.
remote: Counting objects: 100% (76/76), done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 682 (delta 42), reused 47 (delta 21), pack-reused 606
Receiving objects: 100% (682/682), 280.86 KiB | 1.52 MiB/s, done.
Resolving deltas: 100% (388/388), done.
hik@debian-lab-12:~/kramse-labs/docker-install$ ansible-playbook 1-dependencies.yml
```

Kickstart 2: SIEM and Log Analysis – SELKS

Wait for docker to be installed:



```
hlk@debian-lab-12: ~/kramse-labs/docker-install
ok: [127.0.0.1]

TASK [Set a fact with the user name.] *****
ok: [127.0.0.1]

TASK [Add user to docker group, to allow running docker] *****
changed: [127.0.0.1]

TASK [Create dir for scripts] *****
ok: [127.0.0.1]

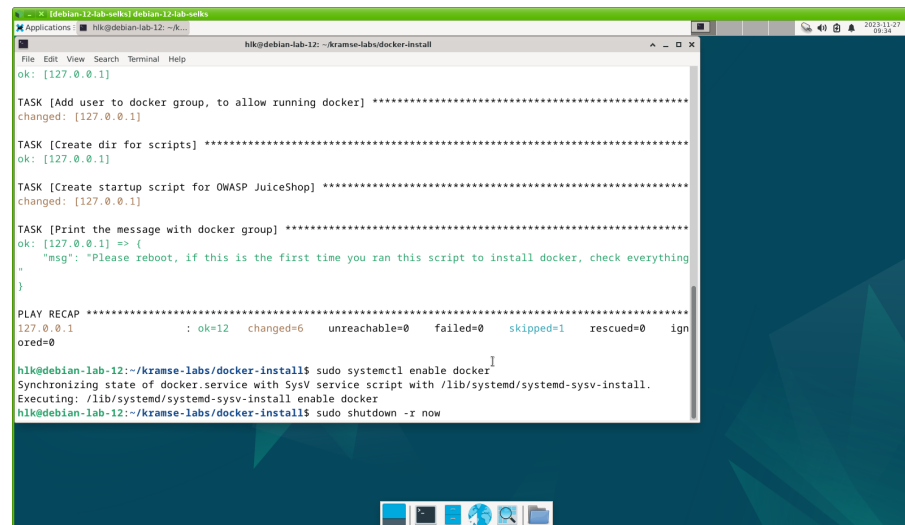
TASK [Create startup script for OWASP JuiceShop] *****
changed: [127.0.0.1]

TASK [Print the message with docker group] *****
ok: [127.0.0.1] => {
  "msg": "Please reboot, if this is the first time you ran this script to install docker, check everything"
}

PLAY RECAP *****
127.0.0.1      : ok=12  changed=6  unreachable=0  failed=0  skipped=1  rescued=0  ignored=0

hlk@debian-lab-12:~/kramse-labs/docker-install$
```

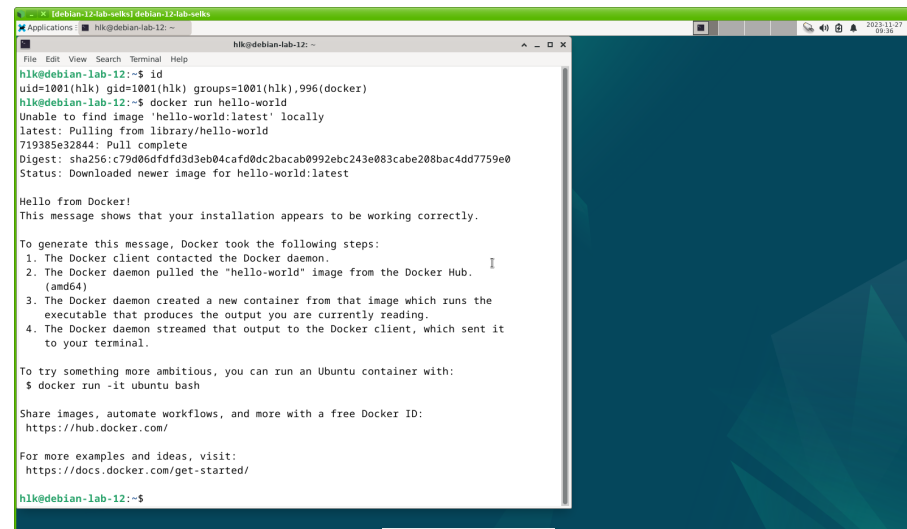
Enable it for reboot and reboot:



```
hlk@debian-lab-12:~/kramse-labs/docker-install$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable docker
hlk@debian-lab-12:~/kramse-labs/docker-install$ sudo shutdown -r now
```

Kickstart 2: SIEM and Log Analysis – SELKS

Check docker – if it only works for root it is also OK to use that:



```
hik@debian-lab-12:~$ id
uid=1001(hik) gid=1001(hik) groups=1001(hik),996(docker)
hik@debian-lab-12:~$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
719385e32844: Pull complete
Digest: sha256:c79d06dffd3d3eb04caf0dc2bacab0992ebc243e083cabe208bac4dd7759e0
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

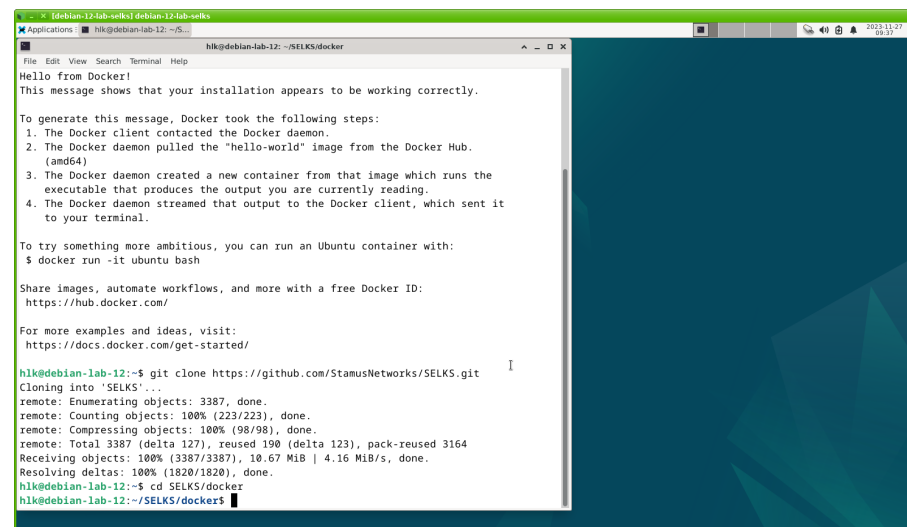
To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

hik@debian-lab-12:~$
```

Git clone SELKS repository:



```
hik@debian-lab-12:~/SELKS/docker$
Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

hik@debian-lab-12:~$ git clone https://github.com/StamusNetworks/SELKS.git
Cloning into 'SELKS'...
remote: Enumerating objects: 3387, done.
remote: Counting objects: 100% (223/223), done.
remote: Compressing objects: 100% (98/98), done.
remote: Total 3387 (delta 127), reused 190 (delta 123), pack-reused 3164
Receiving objects: 100% (3387/3387), 10.67 MiB | 4.16 MiB/s, done.
Resolving deltas: 100% (1820/1820), done.
hik@debian-lab-12:~$ cd SELKS/docker
hik@debian-lab-12:~/SELKS/docker$
```

Kickstart 2: SIEM and Log Analysis – SELKS

Run the `./easy-setup` script:

```

root@debian-12-lab-selks# hiki@debian-lab-12: ~/SELKS
Applications hiki@debian-lab-12: ~/SELKS docker
File Edit View Search Terminal Help
remote: Compressing objects: 100% (98/98), done.
remote: Total 3387 (delta 127), reused 190 (delta 123), pack-reused 3164
Receiving objects: 100% (3387/3387), 10.67 MiB | 4.16 MiB/s, done.
Resolving deltas: 100% (1820/1820), done.
hiki@debian-lab-12:~$ cd SELKS/docker
hiki@debian-lab-12:~/SELKS/docker$ ./easy-setup.sh
DISCLAIMER : This script comes with absolutely no warranty. It provides a quick and eas
y way to install SELKS on your system

Although this script should run properly on major linux distribution, it has only been t
ested on Debian 10, Debian 11, Ubuntu 20.04 and Centos 8

Press any key to continue or ^c to exit

This version of SELKS relies on docker containers. We will now check if docker is alr
eady installed

#####
# INSTALLATION #
#####

+ Docker installation found: Docker version 24.0.7, build afdd53b
+ Docker seems to be installed properly
+ docker-compose installation found

Portainer is a web interface for managing docker containers. It is recommended if you
are not experienced with docker.
Do you want to install Portainer ? [y/n]

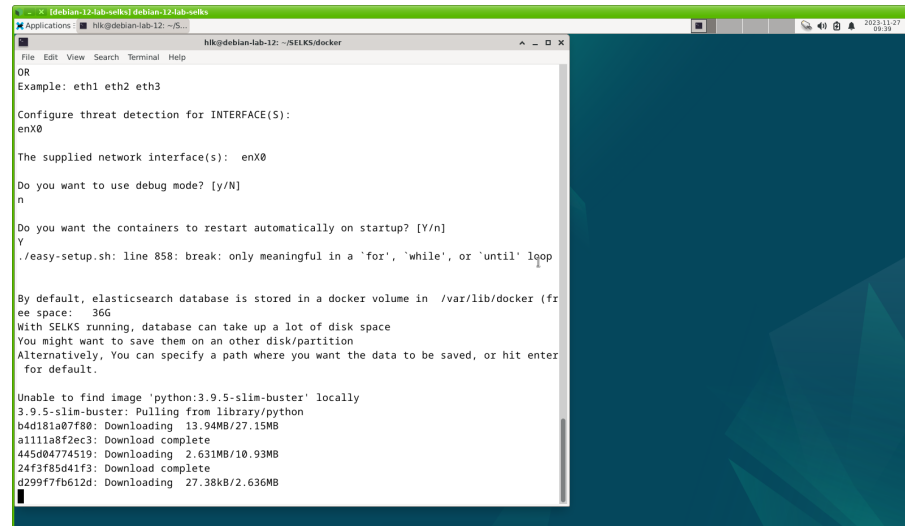
```

Answer questions about network interface

[illegible]

Kickstart 2: SIEM and Log Analysis – SELKS

Docker will start fetching the images – took about 5 minutes on 4G router:



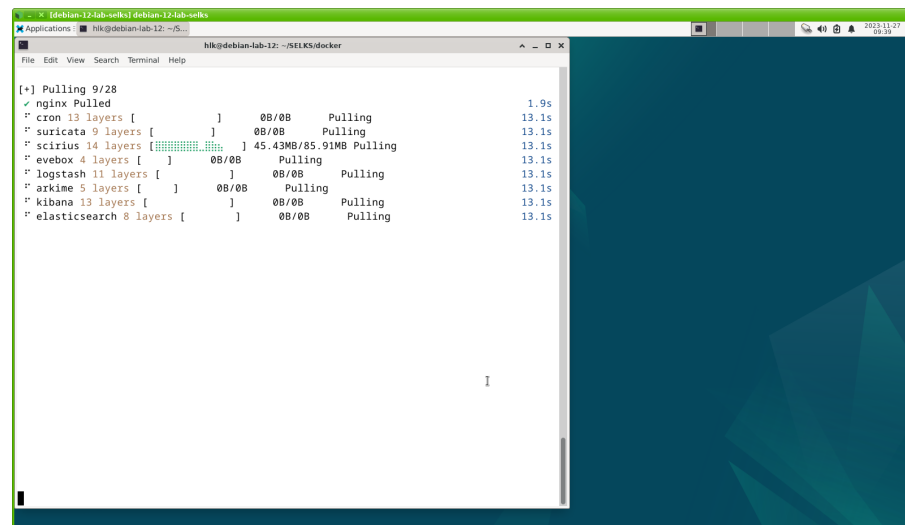
A terminal window titled 'hik@debian-lab-12: ~/SELKS/docker' showing the configuration of SELKS. The user has entered 'n' for debug mode and 'y' for automatic restarts. The script is now downloading Docker images for Elasticsearch and Python. The output shows the progress of downloading several images, including 'python:3.9.5-slim-buster' and 'elasticsearch:7.10.2'.

```
hik@debian-lab-12: ~/SELKS/docker
OR
Example: eth1 eth2 eth3
Configure threat detection for INTERFACE(S):
enX0

The supplied network interface(s): enX0
Do you want to use debug mode? [y/N]
n
Do you want the containers to restart automatically on startup? [Y/n]
y
./easy-setup.sh: line 858: break: only meaningful in a 'for', 'while', or 'until' loop

By default, elasticsearch database is stored in a docker volume in /var/lib/docker (fr
ee space: 36G
With SELKS running, database can take up a lot of disk space
You might want to save them on an other disk/partition
Alternatively, You can specify a path where you want the data to be saved, or hit enter
for default.

Unable to find image 'python:3.9.5-slim-buster' locally
3.9.5-slim-buster: Pulling from library/python
b4d181a07f80: Downloading 13.94MB/27.15MB
a1111a8f2ec3: Download complete
445d04774519: Downloading 2.631MB/10.93MB
24f3f85d41f3: Download complete
d299f7fb612d: Downloading 27.38kB/2.636MB
```

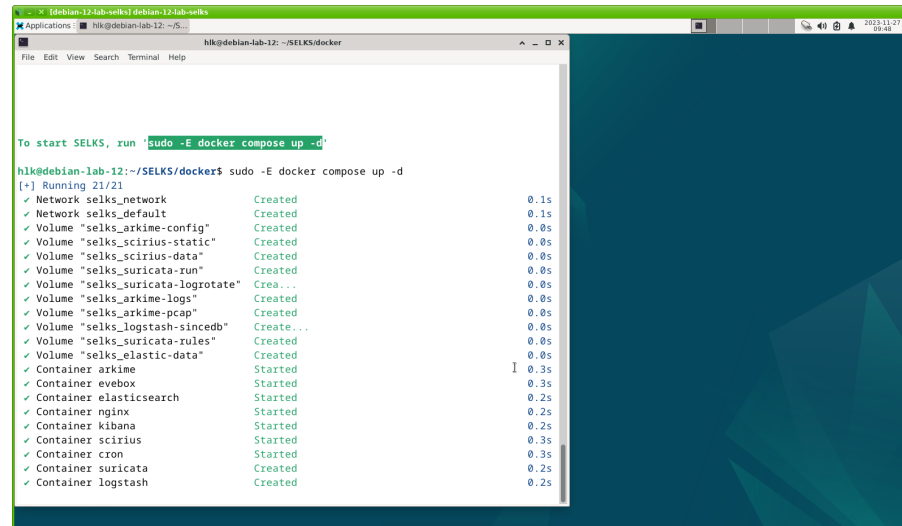


A terminal window titled 'hik@debian-lab-12: ~/SELKS/docker' showing the progress of pulling Docker images. The output shows a list of images being pulled, including 'nginx', 'cron', 'suricata', 'scirius', 'evebox', 'logstash', 'arkime', 'kibana', and 'elasticsearch'. The progress bar for 'scirius' is visible, showing 45.43MB/85.91MB pulled.

```
hik@debian-lab-12: ~/SELKS/docker
[+] Pulling 9/28
✓ nginx Pulled 1.9s
" cron 13 layers [ ] 08/08 Pulling 13.1s
" suricata 9 layers [ ] 08/08 Pulling 13.1s
" scirius 14 layers [ ] 45.43MB/85.91MB Pulling 13.1s
" evebox 4 layers [ ] 08/08 Pulling 13.1s
" logstash 11 layers [ ] 08/08 Pulling 13.1s
" arkime 5 layers [ ] 08/08 Pulling 13.1s
" kibana 13 layers [ ] 08/08 Pulling 13.1s
" elasticsearch 8 layers [ ] 08/08 Pulling 13.1s
```

Kickstart 2: SIEM and Log Analysis – SELKS

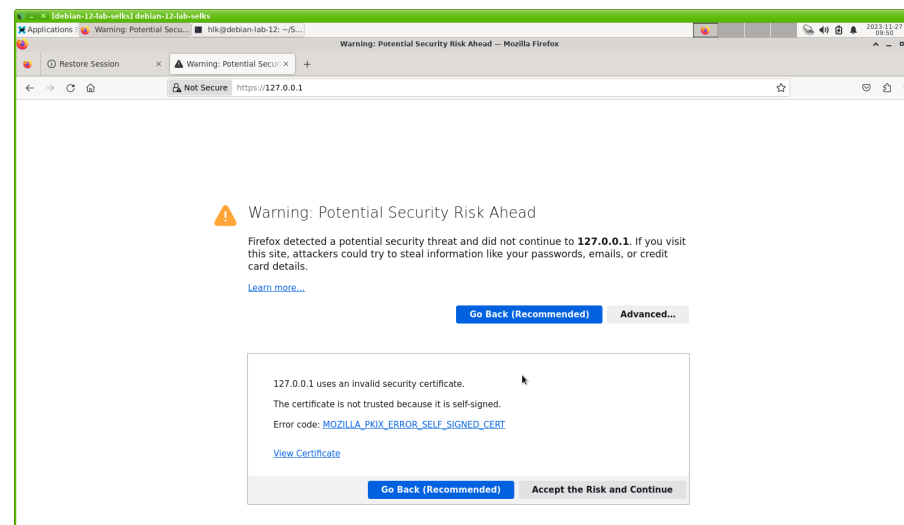
Start the docker containers:



```
hik@debian-lab-12: ~/SELKS/docker
To start SELKS, run 'sudo -E docker compose up -d'

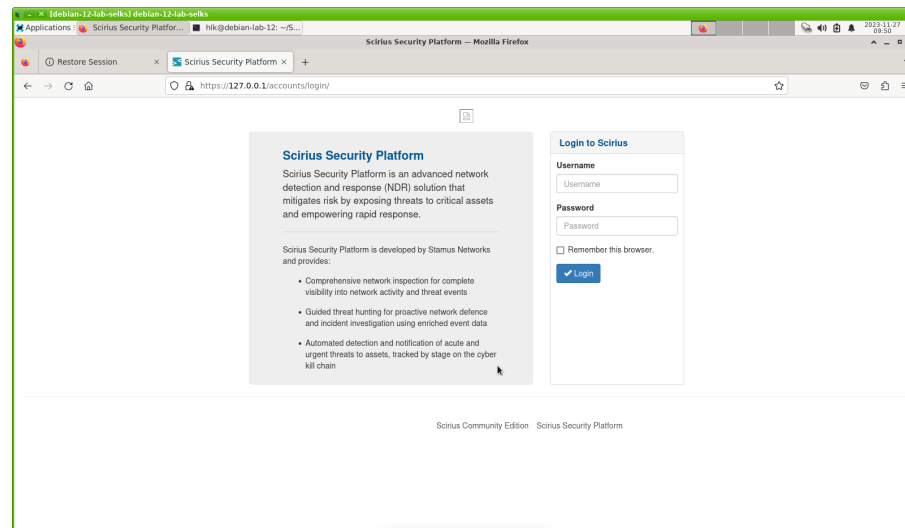
hik@debian-lab-12:~/SELKS/docker$ sudo -E docker compose up -d
[+] Running 21/21
 ✓ Network selks_network      Created           0.1s
 ✓ Network selks_default      Created           0.1s
 ✓ Volume "selks_arkime-config" Created           0.0s
 ✓ Volume "selks_scirius-static" Created           0.0s
 ✓ Volume "selks_scirius-data" Created           0.0s
 ✓ Volume "selks_suricata-run" Created           0.0s
 ✓ Volume "selks_suricata-logrotate" Created           0.0s
 ✓ Volume "selks_arkime-logs" Created           0.0s
 ✓ Volume "selks_arkime-pcap" Created           0.0s
 ✓ Volume "selks_logstash-sincedb" Created           0.0s
 ✓ Volume "selks_suricata-rules" Created           0.0s
 ✓ Volume "selks_elastic-data" Created           0.0s
 ✓ Container arkime            Started           0.3s
 ✓ Container evebox            Started           0.3s
 ✓ Container elasticsearch     Started           0.2s
 ✓ Container nginx             Started           0.2s
 ✓ Container kibana             Started           0.2s
 ✓ Container scirius            Started           0.3s
 ✓ Container cron               Started           0.3s
 ✓ Container suricata          Created           0.2s
 ✓ Container logstash          Created           0.2s
```

Start a browser and accept the self-signed certificate:



Kickstart 2: SIEM and Log Analysis – SELKS

Success – hopefully, login with username: **selks-user** and password: **selks-user**:



After browsing to a few sites::

