Please review the following two instructional videos on Virtualization and Docker (Containerization):

- Understanding Virtualization
- Introduction to Docker and Containerization

While viewing these videos, I encourage you to actively participate by following the demonstrations as much as possible. In the section covering virtualization, the presenter uses VirtualBox for demonstration purposes. However, the principles apply similarly to UTM.

While going through the first video, especially during discussions on 32-bit versus 64-bit systems, be mindful of the architectural choices available today. For instance, Windows users and those with older Macs typically utilize an Intel-compatible chipset. Conversely, modern Macs are equipped with an ARM-compatible chipset (Apple Silicon). This distinction is crucial for understanding how different architectures may influence your virtualization and containerization experience.

****For UTM users**: you'll encounter an option not available in VirtualBox: the choice between Virtualization and Emulation. For the purposes of this exercise, please select the Virtualization option. Additionally, you will need to download the ARM (often uses a -aarm extension) versions of the Linux ISOs for this activity.

Ubuntu ARM version (M1 Mac or newer)
Kali ARM version (M1 Mac or Newer)

Docker (all platforms)
For the portion on Docker, the video suggests using Linode. Instead, I'd like you to install Docker directly on your own operating system. This should be done outside of Kali Linux. To get started with Docker, please visit:

Docker - Get Started

This hands-on approach is designed to enhance your understanding of both virtualization and containerization. We'll delve deeper into the differences between Virtualization and Emulation in tomorrow's kickoff discussion. This knowledge will help with the Wazuh component of this week's sprint, which we will also discuss in the kickoff tomorrow. Happy learning!

**** Bonus work:***
Wazuh: https://wazuh.com/

Wazuh is an open-source security platform designed for threat detection, integrity monitoring, incident response, and compliance. It offers a comprehensive solution for organizations looking to enhance their cybersecurity posture. Here's an overview of its key features and components:

- **Threat Detection and Response**: Wazuh helps in detecting and responding to threats in real-time. It uses advanced analytics and machine learning algorithms to identify suspicious activities and potential security threats across the network. Once a threat is detected, it can automatically respond based on predefined rules.
- **Log Data Analysis**: Wazuh provides extensive log data analysis capabilities, allowing organizations to collect, aggregate, index, and analyze data from different sources (such as operating systems, applications, and network devices) for security insights.
- **File Integrity Monitoring (FIM)**: It monitors and detects changes in files and configurations, helping in the early detection of potential threats like malware, rootkits, or unauthorized modifications.
- **Vulnerability Detection**: Wazuh assesses and identifies vulnerabilities in the system by integrating with various vulnerability databases and scanning tools. This helps in proactive threat mitigation by identifying weak points before they can be exploited.
- **Compliance Management**: It helps organizations comply with various regulatory requirements by providing tools and features for continuous monitoring, reporting, and analysis of the security posture in the context of specific standards (e.g., PCI DSS, GDPR, HIPAA).
- **Incident Response**: Wazuh facilitates a rapid response to identified threats with its automated response capabilities. It can execute custom scripts or commands to mitigate or contain threats, reducing the time to respond to incidents.
- **Scalability and Integration**: Designed to be scalable, Wazuh can be deployed across large infrastructures. It integrates with a wide range of tools and platforms (like Elastic Stack for data visualization and analysis) to enhance its capabilities and provide a unified security solution.
- **Agent and Agentless Monitoring**: Wazuh offers both agent-based and agentless monitoring, allowing for comprehensive coverage of various devices and systems, even those where installing an agent is not feasible.
- **Open Source and Community-Driven**: Being open-source, Wazuh benefits from a community-driven approach where users and developers contribute to its continuous improvement and feature enhancement.

Wazuh can be deployed using docker containers using Docker Compose (included in the docker desktop that you installed).

Docker compose is a tool designed to help define and share multi container applications like Wazuh.

You will need to install git to follow along in the documentation: Git Download

Instructions for Wazuh docker deployment (follow the single-node instructions, and self certificate generation):
Wazuh Docker Deployment

You can run the commands from either the command line in Windows (Start->cmd.exe) or the ‘Terminal’ app in MacOS.

I recommend running the command:

Unset

```
# docker-comose up -b
```

This will run Wazuh in the background and will restart with your computer.

If you run just docker-compose up, you can use CTRL-C to shut down the containers.

Complete the instructions up to mult-node deployment. Now you will be able to browse to your localhost from any browser: https://localhost

The default credentials for the web page will be:
Username: admin
Password: SecretPassword

Additional resources on creating rules in Wazuh and testing them:

https://tryhackme.com/room/wazuhct

https://medium.com/@josephalan17201972/custom-alert-rules-in-wazuh-tryhackme-write-up-613e8e99a6b3

Resource for Integrating Wazuh with Shuffle:

https://wazuh.com/blog/integrating-wazuh-with-shuffle/

Complete everything up to, but **NOT** including “Use case: Detecting and responding to Windows SAM credential dumping”

Protip: To edit the file `/var/ossec/etc/ossec.conf` if you’re running Wazuh in UTM or Docker you’ll need to mount the Wazuh docker images by doing the following:

```
Unset
# docker exec -it single-node-wazuh.manager-1 bash
```

Reference:
Wazuh Docker Container Userage - (at the end)

Use nano or vi to edit the file configuration file. Nano and vi are command line editors. E.g.

```
Unset
# nano /var/ossec/etc/ossec.conf
```

or

```
Unset
# vi /var/ossec/etc/ossec.conf
```

Personally I find nano more intuitive and easier to use key-combinations than vi.

Nano Cheetsheet

VI Command Help

Special note for UTM and Docker users to restart:

When finished editing the config files, instead of using the systemctl as per the instructions to restart the docker instance, you'll need to exit from the docker exec from above by issuing the command

Unset

```
# exit
```

And then issue the following commands:

Unset

```
# docker-compose down
# docker-compose up -d
```

(make sure you're in the same directory as the .yml configuration file. On UTM it's located under /home/wazuh/wazuh-docker/single-node)