GROUP PORTFOLIO 2 ASSIGNMENT

Docker and Zabbix Real Use-Case

DATA 2410 – Networking and Cloud Computing
18.05.2022

Group 04
Ali Haitham Amin – S351952
Kader Hussein Abdi – S354379
Hamnah Hussain – S354401
Yasmin Abubaker Hassan – S348816

Table of Contents

Introduction

This assignment was done on both WSL, windows subsystems for Linux, and tested on the intel1 server provided by the university. The Zabbix version used is version 6.0, and it is the same version throughout the whole assignment. In the first section, we implemented a docker-compose stack, defined a docker bridge network, as well as volume mount and configuration of the docker containers. In the second section, we implemented and used the Zabbix monitoring tool. We configured a Zabbix Proxy and a Zabbix Agent in each virtual machine and configured a connection between the Zabbix Server to the proxy and agent. In the third section, we installed, configured, and started a web server that was connected to localhost and redirected the Zabbix-server frontend. The fourth section included the Zabbix frontend, basic host adding, psk encryption usage, and templates with items and triggers.

Zabbix and Docker

Zabbix is a software that monitors parameters of a network and the status of a server, virtual machine, applications, databases, websites and more. Zabbix, among other things, helps catch server problems faster by flexible notification mechanisms. Zabbix server is a repository that stores all configuration, statistics, and operations. Agent also reports availability and integrity information and statistics to the server. Zabbix proxy collects data for the Zabbix server. The Zabbix Agent's purpose is the monitoring of the status of system resources such as CPU load, network utilization, hard drivers, statistics, and more. Docker compose is used to define and share multiple container applications. With Docker compose, you can create YAML/YML files, define application stacks in a file, and much more.

VM1: Docker containers setup

We started by setting up the docker containers by creating docker-compose stacks with Zabbix-server -MySQL, Zabbix-server-web, MySQL-server, and Zabbix-agent. We defined a docker bridge network for the stack and assigning an ipv4 address for each docker container in the correct subnet, along with the correct gateway for the subnet. The reason for using a bridge network will allow the containers to connect to the same bridged network and be able to communicate with each other. We started by creating all the Zabbix containers and MySQL container. In the docker compose file, we created a network for the Zabbix and MySQL, so they would have permanent ip-addresses.

```
mysql-server:
  image: haakco/mysql80
  ports:
    - 3306:3306
  hostname: mysql-server
  container_name: mysql-server
  restart: unless-stopped
  volumes:
    - mysql-database:/var/lib/mysql
  environment:
    - MYSQL_ROOT_PASSWORD=123
    - MYSQL_DATABASE=zabbix
    - MYSQL_USER=zabbix
    - MYSQL_PASSWORD=123
  cap_add:
    - SYS_NICE
  networks:
    compose_network:
      ipv4_address: 172.25.0.5
```

Figure 1. MySQL server

```
zabbix-server:
  image: zabbix/zabbix-server-mysql
  ports:
    - 10051:10051
  hostname: zabbix-server
  container_name: zabbix-server
  restart: unless-stopped
  volumes:
    - zabbix-server-config:/etc/zabbix
  environment:
    - DB_SERVER_HOST=mysql-server
    - MYSQL_DATABASE=zabbix
    - MYSQL_USER=zabbix
    - MYSQL_PASSWORD=123
  depends_on:
    - mysql-server
  networks:
    compose_network:
      ipv4_address: 172.25.0.2
```

Figure 2. Zabbix-server

```
zabbix-web:
  image: zabbix/zabbix-web-nginx-mysql
  ports:
    - 80:8080
  hostname: zabbix-web
  container_name: zabbix-web
  restart: unless-stopped
  volumes:
    - zabbix-web-config:/etc/zabbix
  environment:
    - DB_SERVER_HOST=mysql-server
    - MYSQL_DATABASE=zabbix
    - MYSQL_USER=zabbix
    - MYSQL_PASSWORD=123
    - ZBX_SERVER_HOST=zabbix-server
  depends_on:
    - zabbix-server
    - mysql-server
  networks:
    compose_network:
      ipv4_address: 172.25.0.3
```

Figure 3. Zabbix-web

```
zabbix-agent:
  image: zabbix/zabbix-agent
  ports:
    - 10050:10050
  hostname: zabbix-agent
  container_name: zabbix-agent
  restart: unless-stopped
  volumes:
    - zabbix-agent-config:/etc/zabbix
  environment:
    - ZBX_SERVER_HOST=zabbix-server
  depends_on:
    - zabbix-server
  networks:
    compose_network:
      ipv4_address: 172.25.0.4
```

Figure 4. Zabbix-agent

```
networks:
  compose_network:
    external: true

volumes:
  mysql-database:
  zabbix-server-config:
  zabbix-web-config:
  zabbix-agent-config:
```

Figure 5. Volumes and Network

VM2 and VM3: Install Zabbix-agent and Zabbix-proxy

The first thing done with at VM2 was installing both MariaDB and Zabbix proxy. Then we connected the Zabbix server from VM1 with MariaDB and Zabbix proxy in VM2.

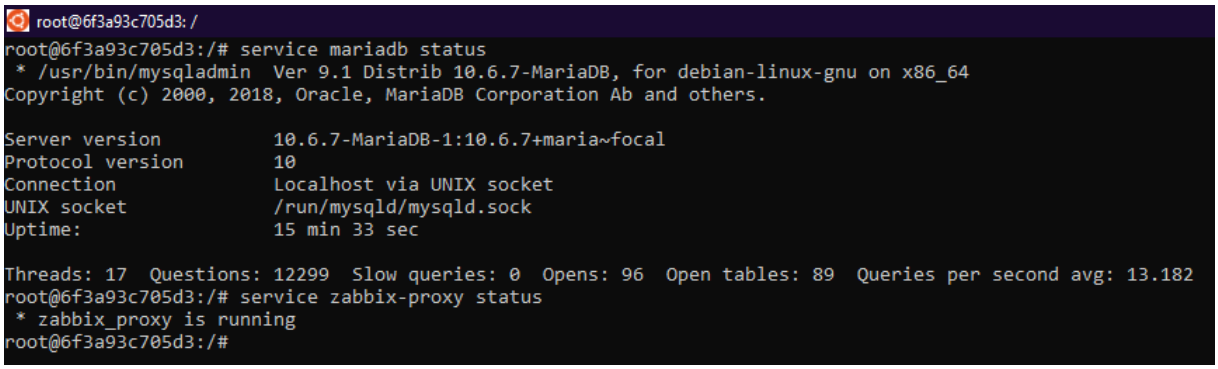


Figure 6. MariaDB

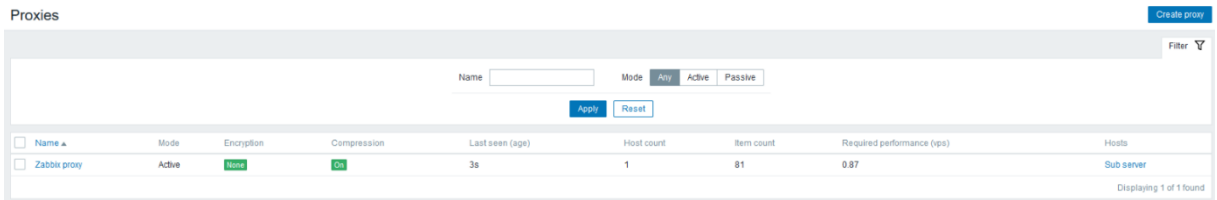


Figure 7. Zabbix proxy

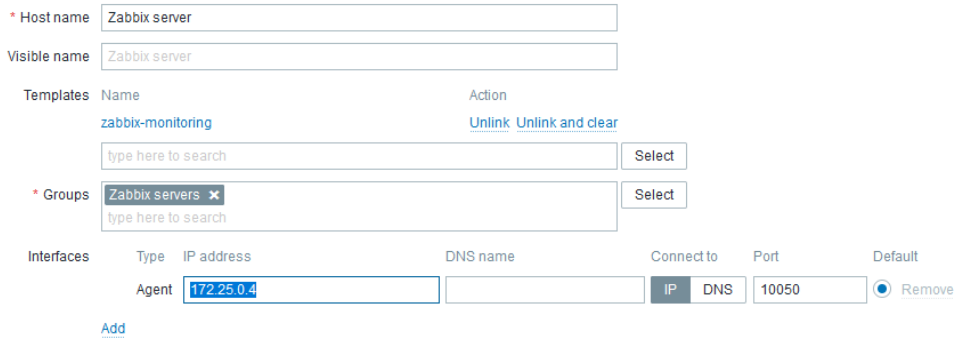


Figure 8. Zabbix server

Later we installed Zabbix agent to VM3.

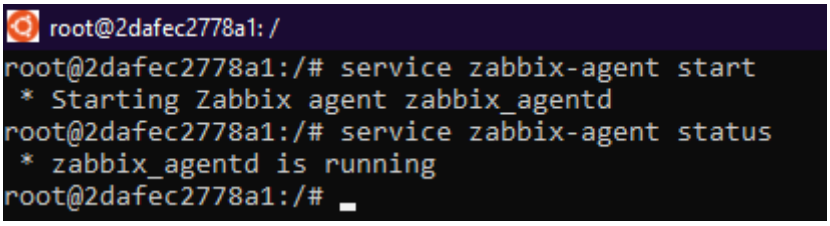


Figure 9. Zabbix agent running

We then generated the hex values for psk encryption file on VM3. Then moved the file to directory giving it permission so that Zabbix agent can have access to the file:

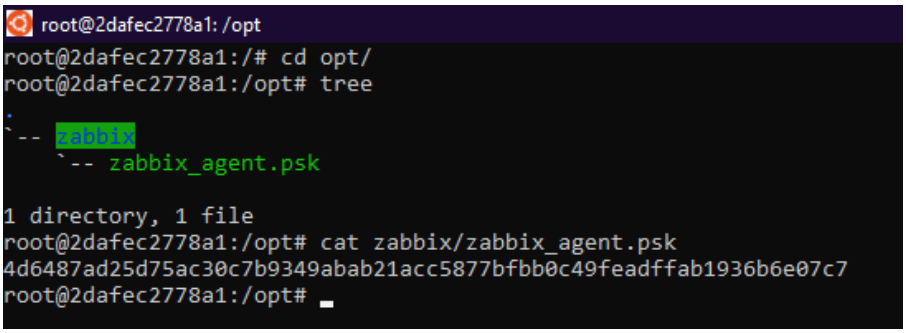


Figure 10. Zabbix agent psk file

We enabled the psk encryption in the Zabbix agent, configured the file and set the valid server address:

Figure 11. Zabbix agent information

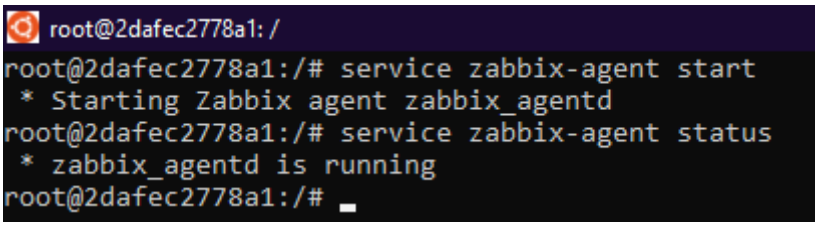


Figure 12. Zabbix proxy running

VM2: Nginx proxy

First, we installed nginx proxy and prepared the configuration file for nginx:

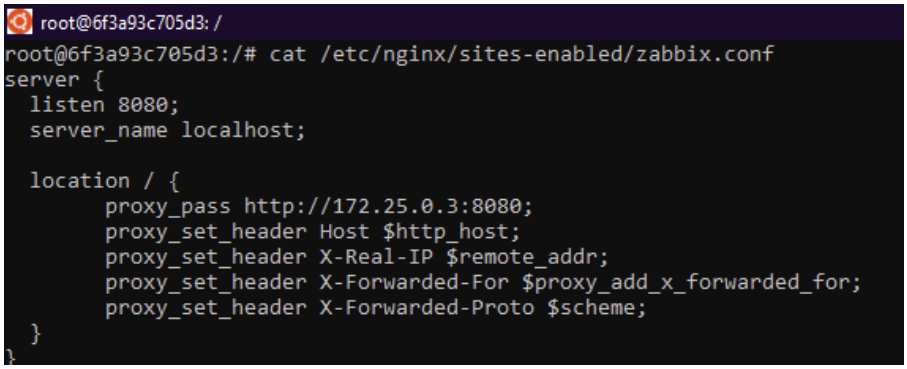


Figure 13. Zabbix.conf

Then we added directories to the nginx directory, including sites-enabled and includes. We configured the nginx.conf file to connect to the Zabbix-server-web, and to be able to accessed on port 8080.

```
        include /etc/nginx/conf.d/*.conf;
        include /etc/nginx/sites-enabled/*;
        include /etc/nginx/includes/*;
```

Figure 14. Enabling and including directories

VM1: Zabbix frontend

To start the frontend process we created a host group named Zabbix-monitoring.

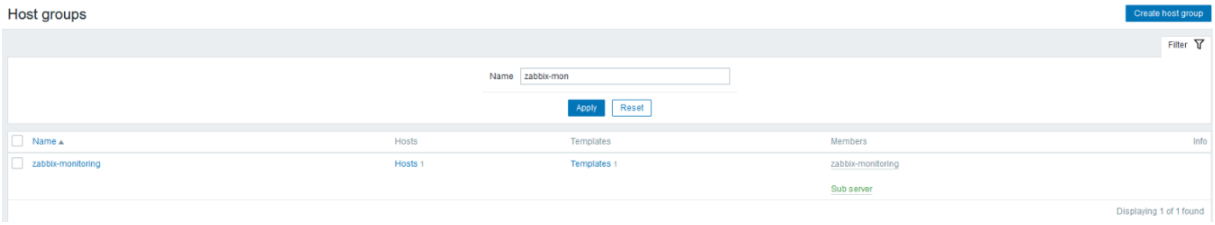


Figure 15. Zabbix-monitoring

We added a host named “Sub server” that is configured with VM3 ipv4 address and port used by Zabbix-agent.

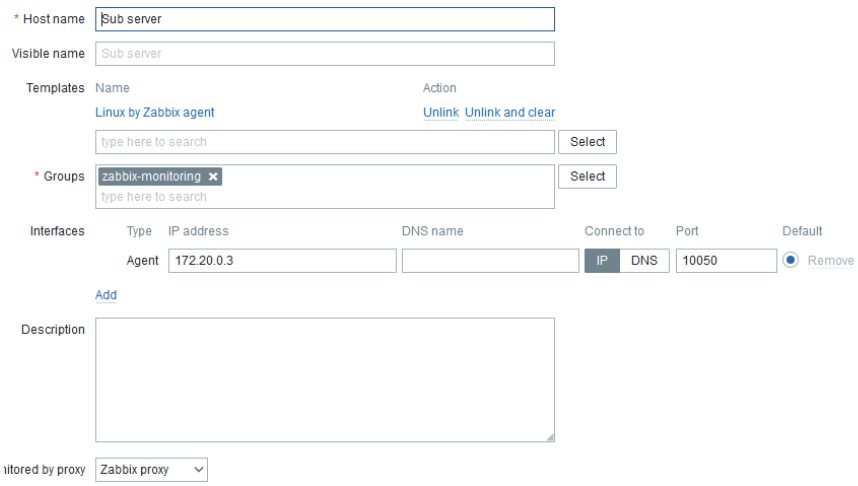


Figure 16. Sub server

Then we enabled the encryption with PSK and configured it.

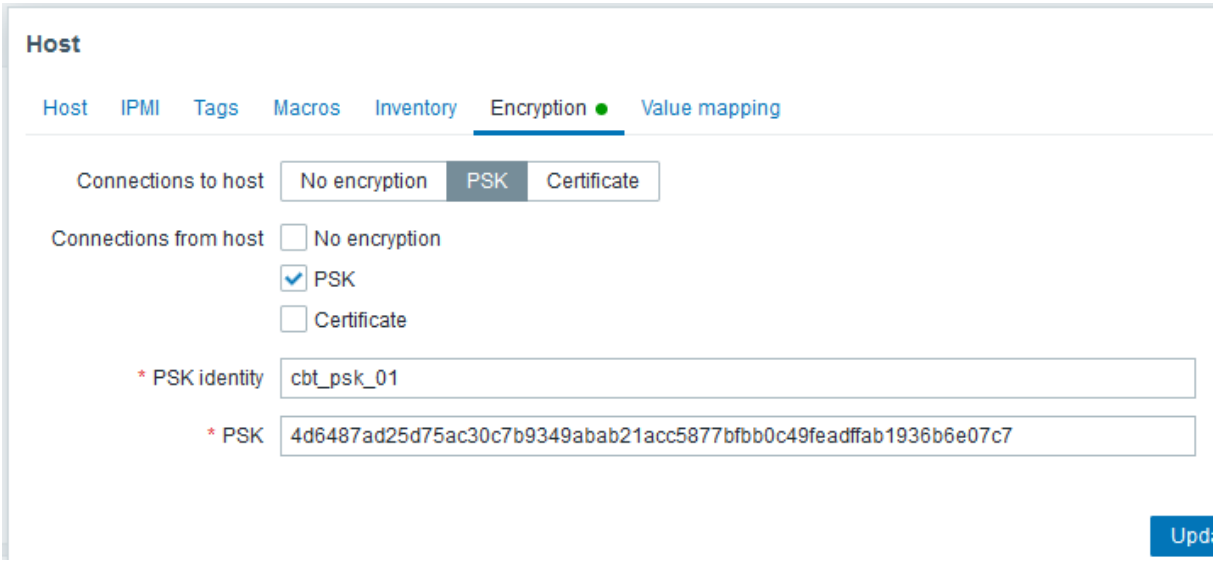


Figure 17. Sub server psk

We then created a new template named Zabbix-monitoring and added the following items and triggers:

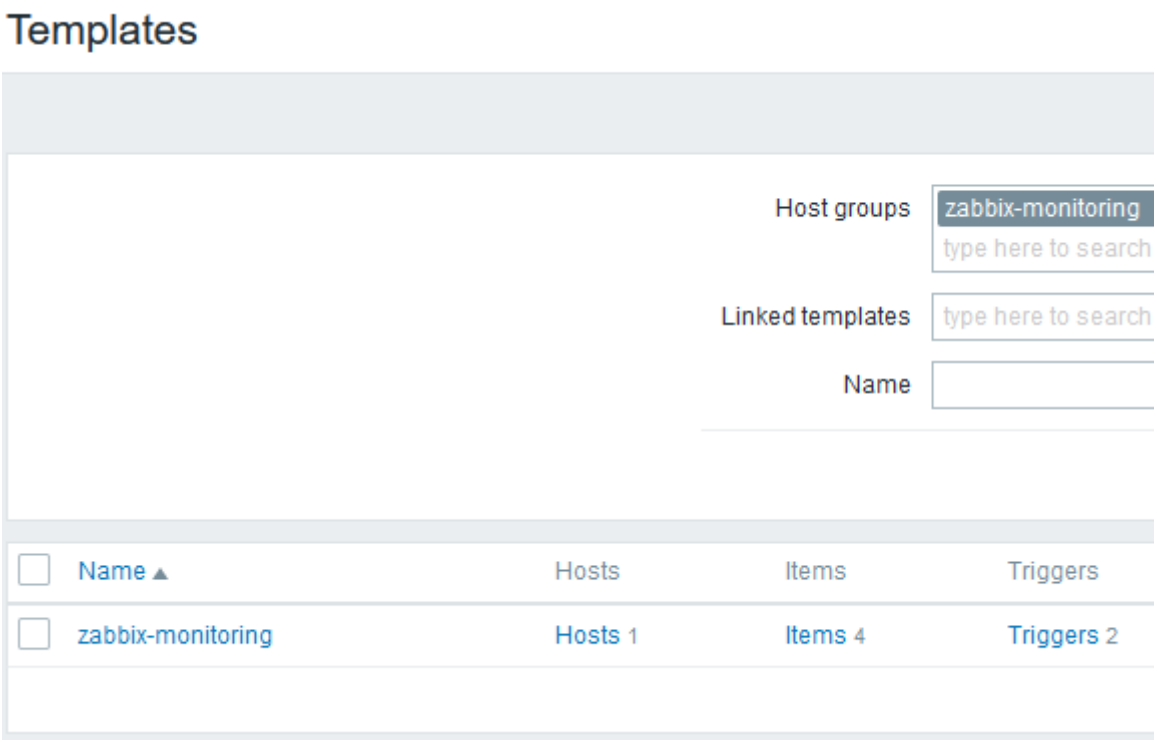
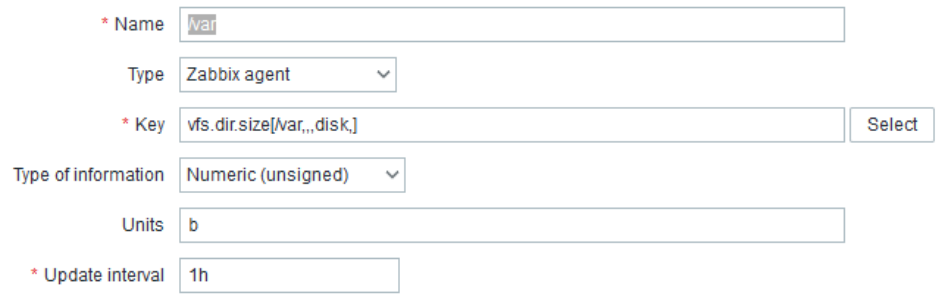
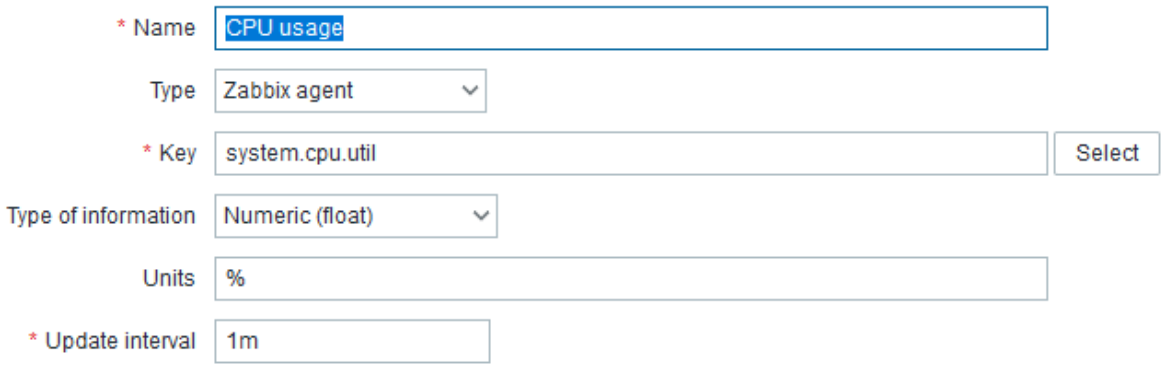


Figure 18. Zabbix-monitoring template

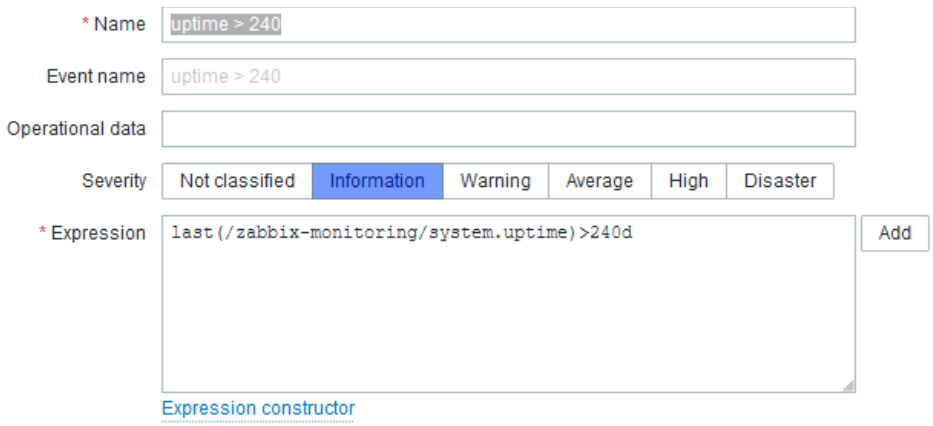
1. An item in the template that monitors the total used disk space on the directory /var, interval 1h (one hour).



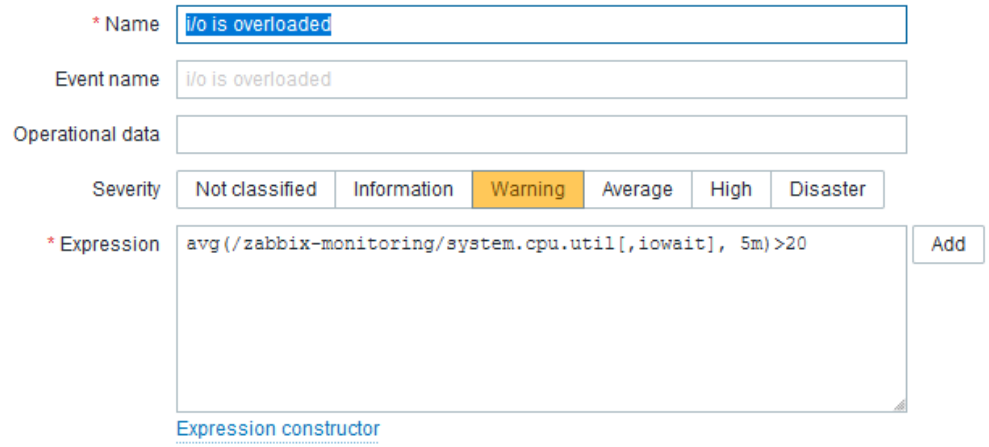
2. An item in the template that monitors docker process usage, interval 1m, units %.



3. A trigger in the template that triggers when uptime is longer than 240 days, assign it type information.

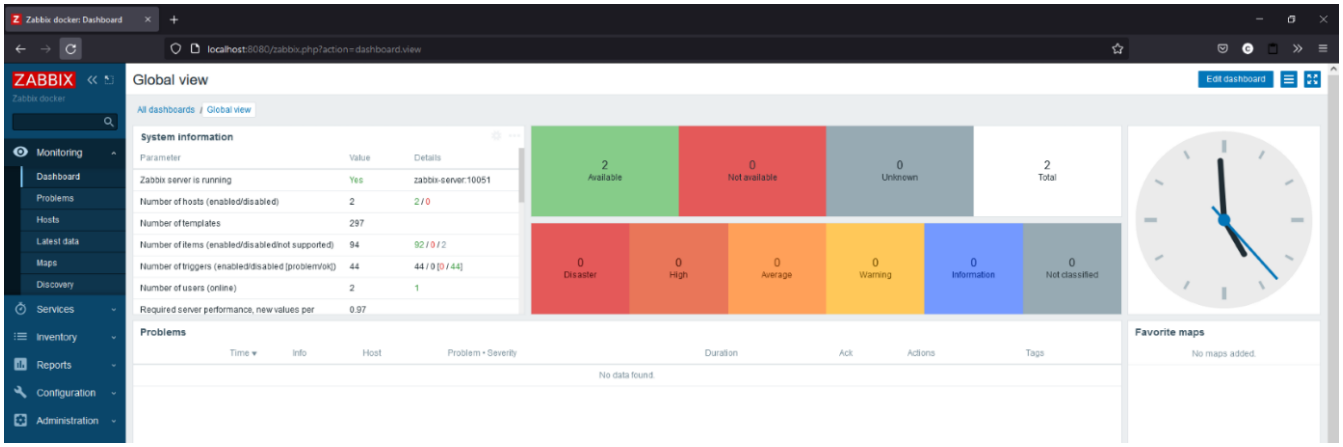


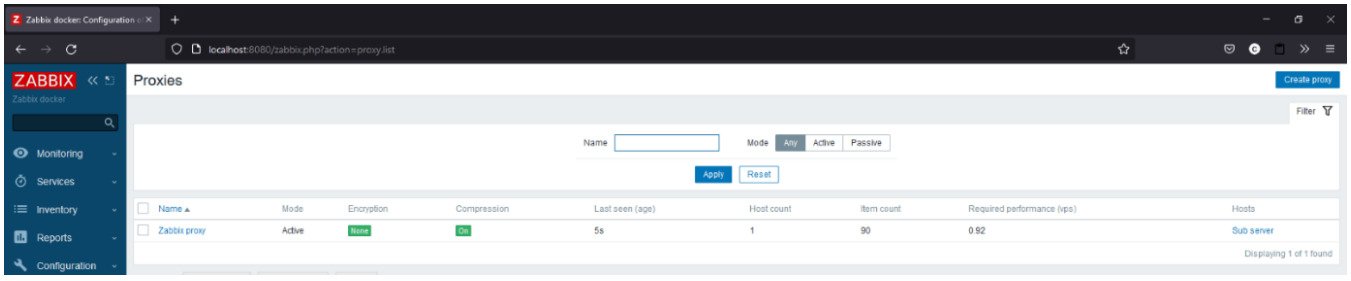
4. A trigger in the template that triggers when disk I/o is overloaded, higher than 20, average 5min

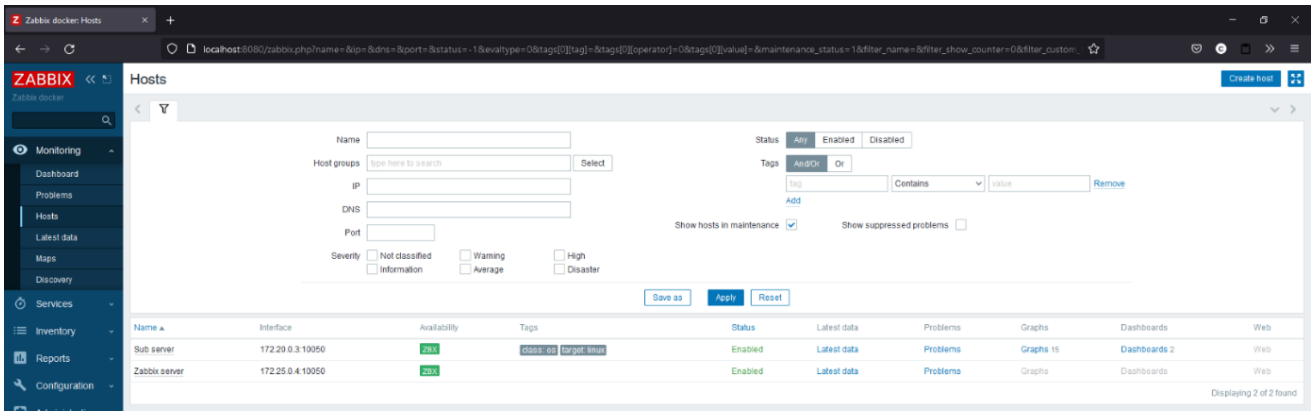


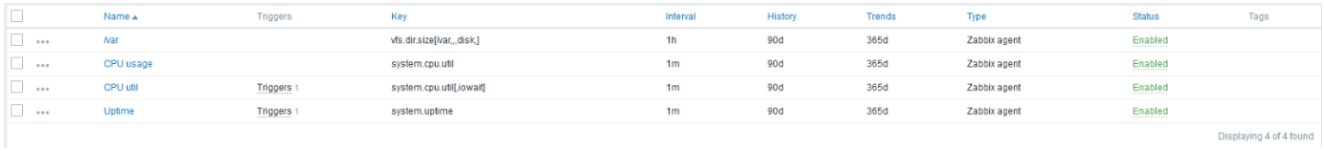
Results

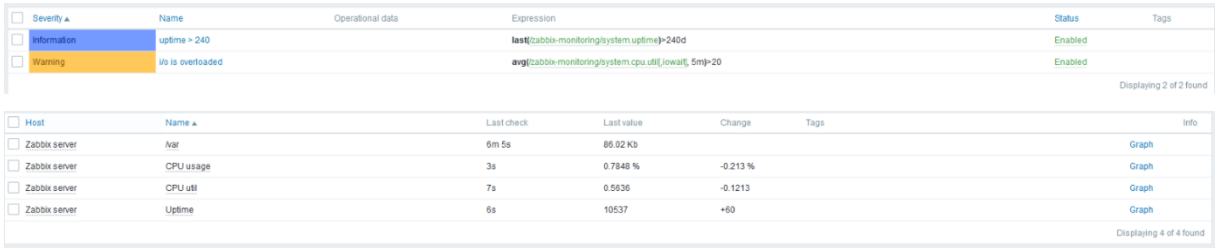
Zabbix dashboard:

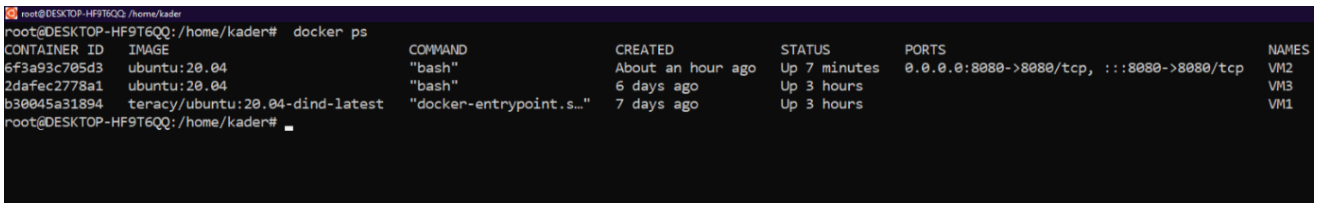




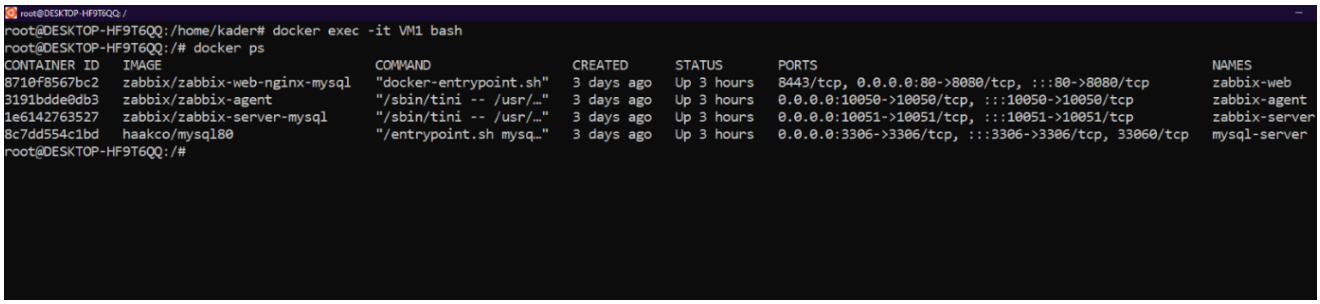




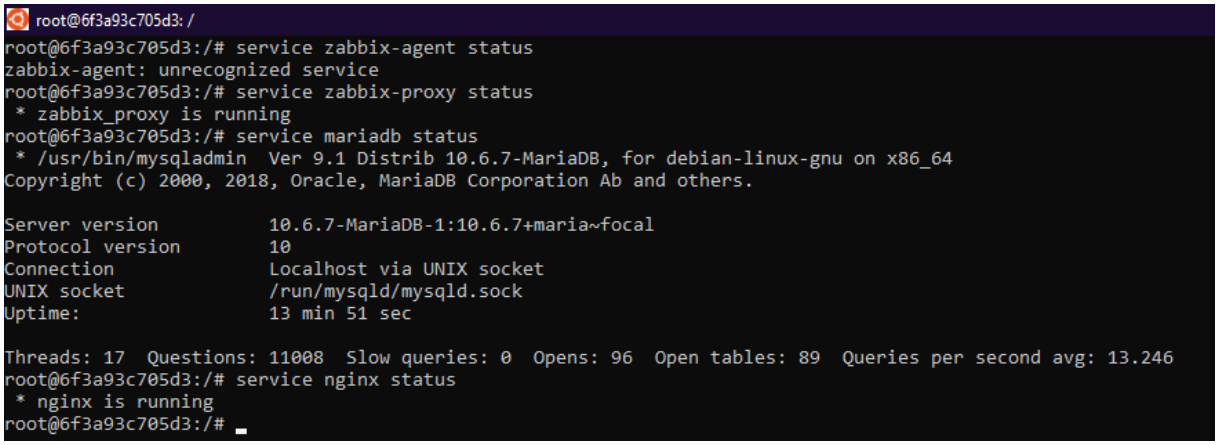




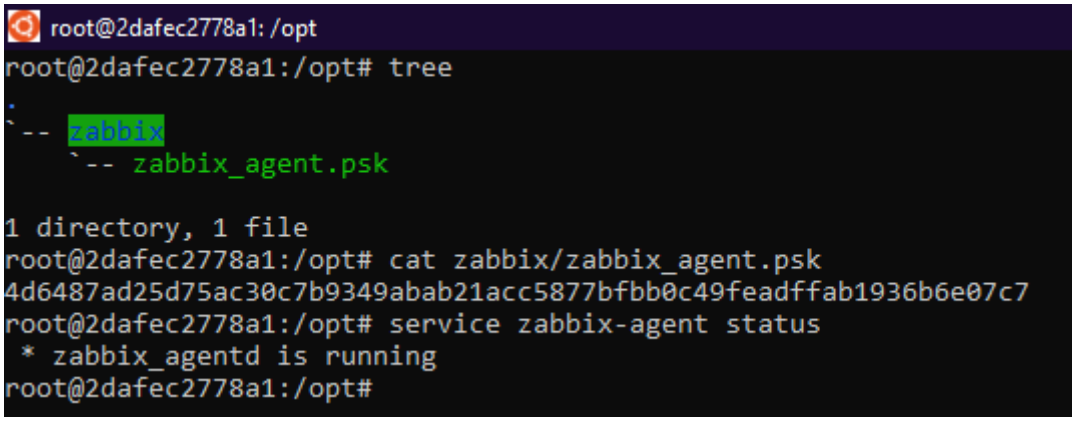
VM1 – with the docker containers:



VM2:



VM3:



Conclusion

In this assignment we used WSL, and we copied all the files to the intel1 server and tested the containers on the intel1 server. It compiled the files successfully. We started by creating the docker-compose files for VM1. Thereafter we made the Zabbix-Proxy and nginx proxy in VM2 and configured them so they could work as they should. In VM3 we created Zabbix-agent, configured, and connected it to Zabbix-proxy in VM2. Then we configured nginx, setting them up correctly. At last, we worked on the Zabbix frontend site. All the applications work on both the WSL and intel1 server.

Sources

Zabbix (n.d.). *2 What is Zabbix*. Zabbix Documentation. Retrieved May 01, 2022, from https://www.zabbix.com/documentation/current/en/manual/introduction/about

Zabbix (n.d.). *4 Zabbix Overview*. Zabbix Documentation. Retrieved May 10, 2022, from https://www.zabbix.com/documentation/current/en/manual/introduction/overview

Docker (n.d.). *Use Docker compose*. Docker Documentation. Retrieved April 29, 2022, from https://docs.docker.com/get-started/08_using_compose/