



Group Portfolio 2 Assignment

Docker and Zabbix Real Use-Case

- This is a **group** assignment.
- **Deadline: Wednesday May 18th, 2022, at 23:59.**
- There will be **no extensions** for any reason. If you are not done by then, hand in what you have.
- The assignment is a part of your portfolio and counts towards your final grade.
- A PDF report with a description of your implementation with screenshots The PDF file must contain detailed instructions on how to run the scenario documentation of all the implemented functionality, with screenshots showing what to expect when running. APA-7 should be used, with a title, student name, introduction, and other subsections with a clear explanation of your outcome for each step, followed by a conclusion. There is no need to use the references in your report writing. Your report should be no more than twenty pages.
- You can preferably use Ubuntu 20.04.4 LTS (Focal Fossa), and you can use your own VMware Workstation, Virtualbox, or UTM. Otherwise, you can work on TKD-OsloMet servers that come with Ubuntu 20.04.4 LTS (Focal Fossa). However, you will be informed in your group about the access instructions.
- Feel free to add in more scenarios as you like, such as automation, layer four load balancing NAT Mode, etc.

Grading criteria for the required report			
Assessment and deliverables	VM1: Docker containers setup	10%	Total = 40%
	VM2 and VM3: Install zabbix-agent and zabbix-proxy	10%	
	VM2: Nginx proxy	10%	
	VM1: Zabbix frontend	10%	

Table of Contents:

I.	Introduction	2
II.	VM1: Docker containers setup	4
III.	VM2 and VM3: Install zabbix-agent and zabbix-proxy	5
IV.	VM2: Nginx proxy	6
V.	VM1: Zabbix frontend	7



I. Introduction

The purpose of the first section of the exam is to ensure that the student shows his understanding of Docker containers through implementing a docker-compose stack on VM1. The student will show knowledge of understanding docker networking, volume mounts, environment variables, and basic configuration of each docker container configured through the docker-compose stack. **(10 points)**

The second section has the goal of showing students a real-world scenario of how the Zabbix monitoring tool can be implemented and used. Students will use their knowledge of Zabbix to configure a docker-compose stack and install Zabbix-agent and Zabbix-proxy on separate virtual machines. That will provide hands-on experience with 2 different approaches (docker-compose stack and manual installation of a Zabbix-agent and Zabbix-proxy service). Students should show an understanding of network communication between different VMs to successfully configure Zabbix-agent and Zabbix-proxy to communicate with Zabbix-server. **(10 points)**

The third section of the exam is based on web server usage. It is asked of a student to show skills that are required to install, configure, and start a web server that will listen on localhost and redirect its requests to the Zabbix-server frontend. **(10 points)**

The fourth section of the exam will examine the student's familiarity with the Zabbix frontend, basic host adding, psk encryption usage¹, and the creation of a template with new items and triggers. This student is ready to start working on a real Zabbix monitoring stack and gives the student understanding and hands-on experience in the real-world scenario. **(10 points)**

However, the architecture diagram is shown in Figure 1.

Do not panic

The final exam could become something worth showing a future employer. Use that opportunity. At the same time, remember that the main objective is to show what you have learned about networking and cloud computing. While it is great to make a fun and impressive production, do not lose that focus. Good luck and have fun!

¹ https://en.wikipedia.org/wiki/Pre-shared_key

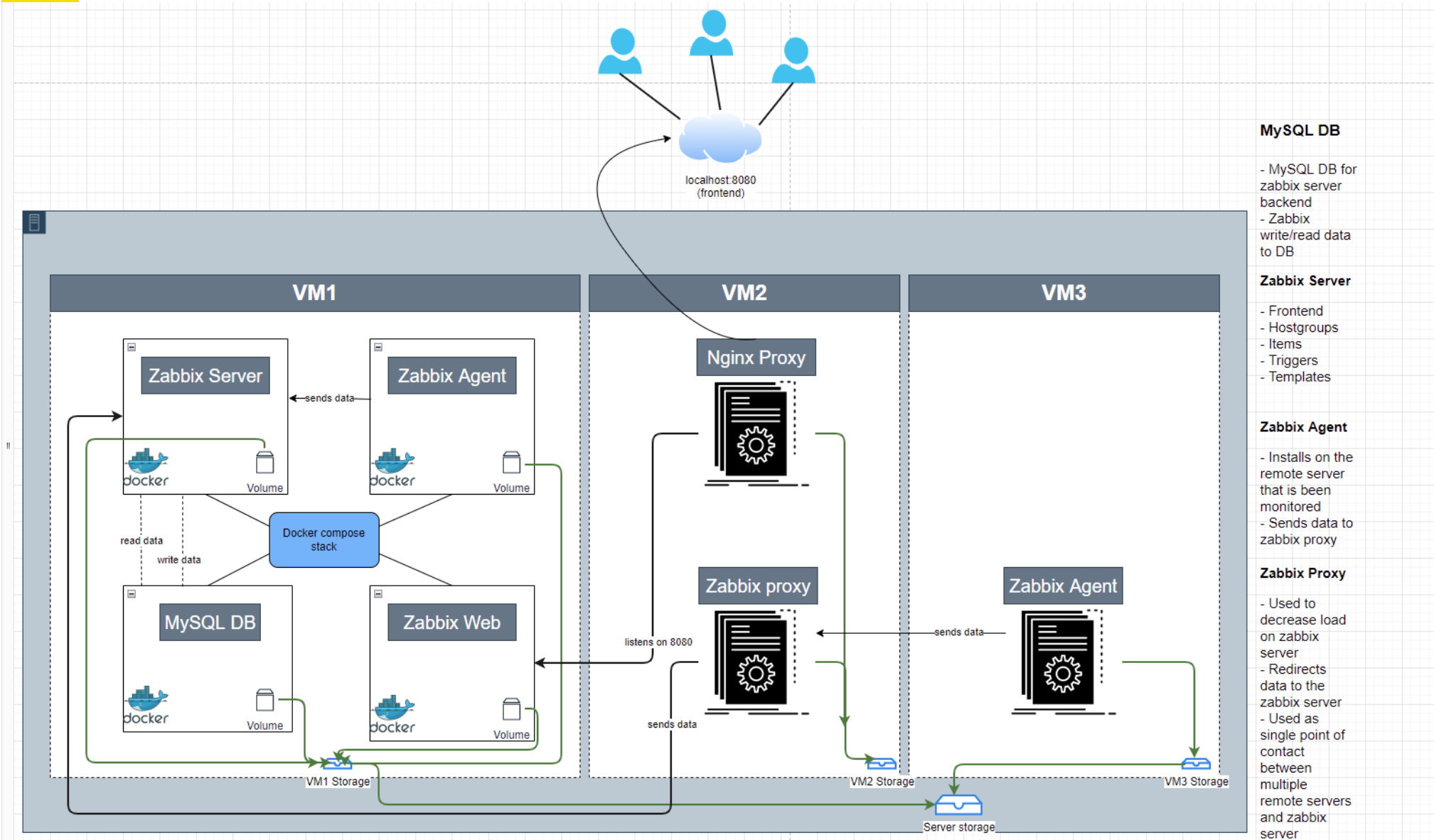


Figure 1: Architecture diagram



II. VM1: Docker containers setup

1. Install Docker v20.10.8 and docker-compose from the official website.
`https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)`
2. Create a docker-compose stack with zabbix-server-mysql, zabbix-server-web, mysql-server, and zabbix-agent
 - a. Define a docker bridge network for the stack, assign it subnet and gateway, and assign each container with static ipv4 address
3. Container 1: mysql-server
 - a. Image: haakco/mysql80
 - b. Port: 3306
 - c. Set hostname and container_name to mysql-server
 - d. Set restart policy: unless-stopped
 - e. Create a local mysql-server-data volume to the /var/lib/mysql path in the container, declare the volume
 - f. Set environment variables (MYSQL_ROOT_PASSWORD: 123, MYSQL_DATABASE: zabbix, MYSQL_USER: zabbix, MYSQL_PASSWORD: 123)
 - g. Add the capability CAP_SYS_NICE so mysql-server can handle the mbind error silently with cap_add: SYS_NICE
4. Container 2: zabbix-server-mysql
 - a. Image: zabbix/zabbix-server-mysql
 - b. Port: 10051
 - c. Set hostname and container_name to zabbix-server
 - d. Set restart policy: unless-stopped
 - e. Create a local zabbix-server-config volume to the /etc/zabbix path in the container, declare the volume
 - f. Set environment variables (DB_SERVER_HOST: mysql-server, MYSQL_DATABASE: zabbix, MYSQL_USER: zabbix, MYSQL_PASSWORD: 123)
 - g. Set dependency on mysql-server
5. Container 3: zabbix-web-nginx-mysql
 - a. Image: zabbix/zabbix-web-nginx-mysql



- b. Port: 80:8080
 - c. Set hostname and container_name to zabbix-web
 - d. Set restart policy: unless-stopped
 - e. Create a local zabbix-web-config volume to the /etc/zabbix path in the container, declare the volume
 - f. Set environment variables (DB_SERVER_HOST: mysql-server, MYSQL_DATABASE: zabbix, MYSQL_USER: zabbix, MYSQL_PASSWORD: 123, ZBX_SERVER_HOST: zabbix-server)
 - g. Set dependency on mysql-server and zabbix-server
6. Container 4: zabbix/zabbix-agent
- a. Image: zabbix/zabbix-agent
 - b. Port: 10050
 - c. Set hostname and container_name to zabbix-agent
 - d. Set restart policy: unless-stopped
 - e. Create a local zabbix-agent-config volume to the /etc/zabbix path in the container, and declare the volume
 - f. Set environment variables (ZBX_SERVER_HOST: zabbix-server)
 - g. Set dependency on zabbix-server

III. VM2 and VM3: Install zabbix-agent and zabbix-proxy

1. Zabbix proxy installation on VM2
 - a. Install zabbix-proxy from [https://repo.zabbix.com/zabbix/6.1/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.1-1+ubuntu\\$\(lsb_release -rs\)_all.deb](https://repo.zabbix.com/zabbix/6.1/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.1-1+ubuntu$(lsb_release -rs)_all.deb) and required packages zabbix-proxy-mysql zabbix-sql-scripts
 - b. Install MariaDB
 - c. Configure zabbix-proxy to connect to the MariaDB
 - d. Configure zabbix-proxy to connect to zabbix-server
 - e. Start and enable zabbix-proxy
2. Zabbix agent installation on VM3
 - a. Install zabbix-agent from [https://repo.zabbix.com/zabbix/6.1/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.1-1+\\$\(lsb_release -sc\)_all.deb](https://repo.zabbix.com/zabbix/6.1/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.1-1+$(lsb_release -sc)_all.deb)
 - b. Start and enable zabbix-agent
 - c. Generate hex value for psk encryption file on your local VM3. (Command to generate: openssl rand -hex 32 > zabbix_agent.psk)
 - d. Move file to /opt/zabbix/ directory and give it permissions which will enable



zabbix-agent to access the file.

- e. Enable psk encryption in the zabbix-agent configuration file and set valid server address
(TLSConnect=psk,TLSAccept=psk,TLSPSKIdentity=cbt_psk_01,TLSPSKFile=/zabbix_agent.psk,Server=ipaddressofzabbixproxy)

IV. VM2: Nginx proxy

1. Install nginx proxy and prepare the files to configure nginx proxy

- File1: /etc/nginx/sites-enabled/zabbix.conf

```
server {  
    listen 8080;  
    server_name localhost;  
  
    location / {  
        proxy_pass http://${ipaddressofzabbixserver}:8080;  
        proxy_set_header Host $http_host;  
        proxy_set_header X-Real-IP $remote_addr;  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_set_header X-Forwarded-Proto $scheme;  
    }  
}
```

2. Add to /etc/nginx/nginx.conf include directories
 - a. include /etc/nginx/sites-enabled/*;
 - b. include /etc/nginx/includes/*;
3. zabbix-server-web has its own nginx.conf and it needs to be configured to listen on port 8080



V. VM1: Zabbix frontend

1. Create a host group named zabbix-monitoring
2. Create a host, assign it to the newly created group, and configure the agent with VM3 ipv4 address and port used by zabbix-agent
3. Assign a hostname and visible name
4. Add a template named Linux by zabbix agent
5. Enable encryption with PSK and configure it
6. Create a new template named zabbix-monitoring and add the following items and triggers
 - a. Create an item in the template that will monitor the total used disk space on the directory /var, interval 1h
 - b. Create an item in the template that will monitor docker process usage, interval 1m, units %
 - c. Create a trigger in the template that will trigger when uptime is longer than 240 days, assign it type information (solution: {test-hostname:system.uptime.last()}>240d)
 - d. Create a trigger in the template that will trigger when disk i/o is overloaded, higher than 20, average 5min (solution: {test-hostname:system.cpu.util[,iowait].avg(5m)}>20)