

# VIRTUALIZATION & CLOUD COMPUTING

AUTHORS:

SENATORI DAVIDE

XIA CHENGHAO

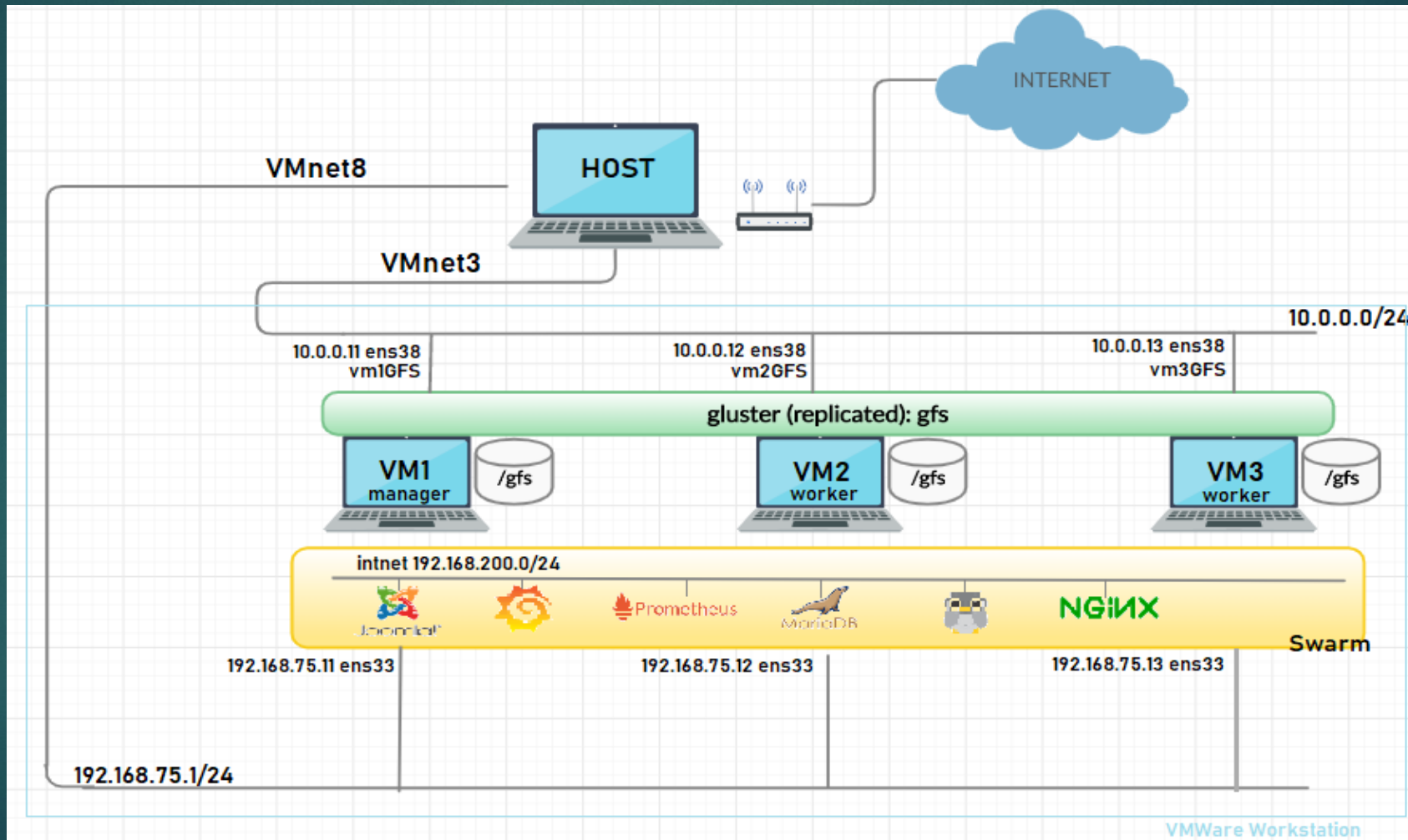


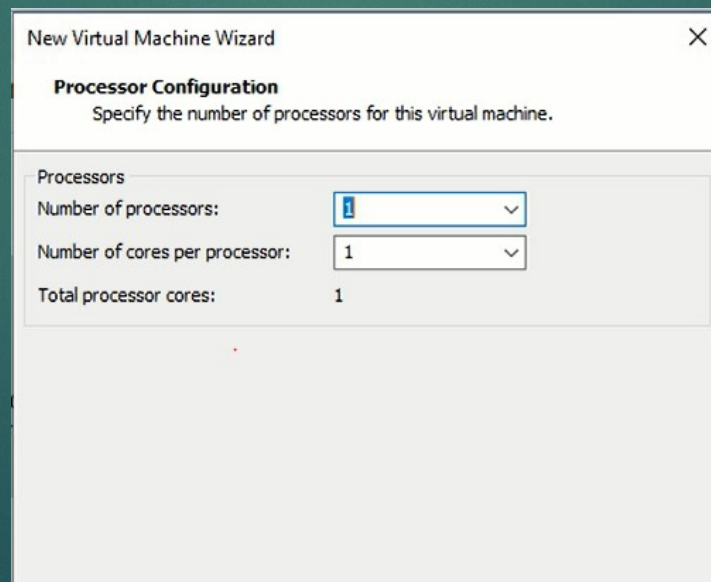
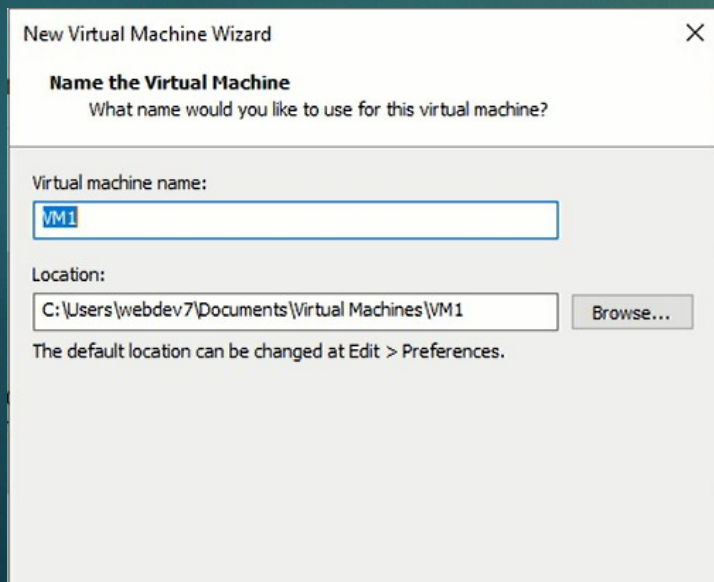
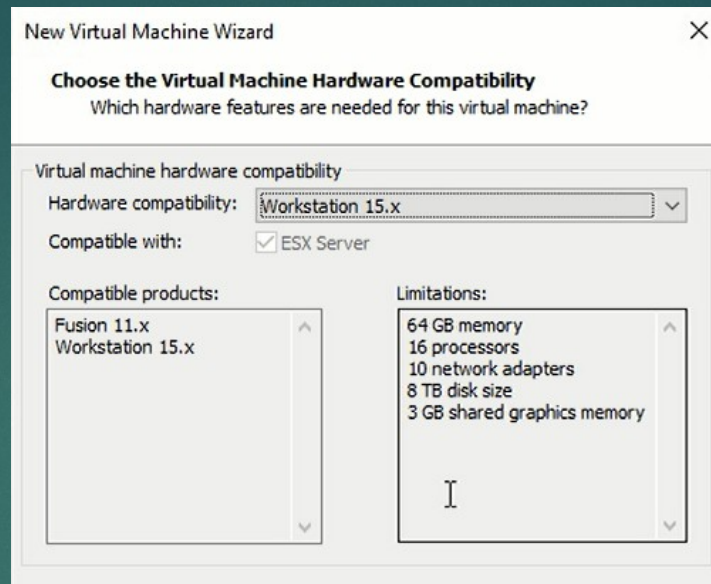
docker



# INTRODUZIONE

L'obiettivo del progetto è di implementare una applicazione multi-tier deployata in un Docker Swarm attraverso 3 Virtual Machines.





## CONFIGURAZIONE VMS



New Virtual Machine Wizard

### Memory for the Virtual Machine

How much memory would you like to use for this virtual machine?

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine:  MB

64 GB -  
32 GB -  
16 GB -  
8 GB -  
4 GB -  
2 GB -  
1 GB -  
512 MB -  
256 MB -  
128 MB -  
64 MB -  
32 MB -  
16 MB -  
8 MB -  
4 MB -

Maximum recommended memory:  
6.1 GB

Recommended memory:  
2 GB

Guest OS recommended minimum:  
1 GB

New Virtual Machine Wizard

### Network Type

What type of network do you want to add?

Network connection

☐ Use bridged networking  
Give the guest operating system direct access to an external Ethernet network. The guest must have its own IP address on the external network.

☒ Use network address translation (NAT)  
Give the guest operating system access to the host computer's dial-up or external Ethernet network connection using the host's IP address.

☐ Use host-only networking  
Connect the guest operating system to a private virtual network on the host computer.

☐ Do not use a network connection

New Virtual Machine Wizard

### Select I/O Controller Types

Which SCSI controller type would you like to use?

I/O controller types

SCSI Controller:

☐ BusLogic (Maximum disk capacity: 2 TB)

☐ LSI Logic (Recommended)

☐ LSI Logic SAS

☒ Paravirtualized SCSI

New Virtual Machine Wizard

### Specify Disk Capacity

How large do you want this disk to be?

Maximum disk size (GB):

Recommended size for Ubuntu: 20 GB

☐ Allocate all disk space now.  
Allocating the full capacity can enhance performance but requires all of the physical disk space to be available right now. If you do not allocate all the space now, the virtual disk starts small and grows as you add data to it.

☐ Store virtual disk as a single file

☒ Split virtual disk into multiple files  
Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.

# CONFIGURAZIONE VMS

Add Hardware Wizard

Specify Disk Capacity

How large do you want this disk to be?

Maximum disk size (GB):

Recommended size for Ubuntu 64-bit: 20 GB

☐ Allocate all disk space now.

Allocating the full capacity can enhance performance but requires all of the physical disk space to be available right now. If you do not allocate all the space now, the virtual disk starts small and grows as you add data to it.

☐ Store virtual disk as a single file

☒ Split virtual disk into multiple files

Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.

< Back

Next >

Cancel

Add Hardware Wizard

Hardware Type

What type of hardware do you want to install?

Hardware types:

Hard Disk

CD/DVD Drive

Floppy Drive

Network Adapter

USB Controller

Sound Card

Parallel Port

Serial Port

Printer

Generic SCSI Device

Trusted Platform Module

Explanation

Add a network adapter.

Finish

Cancel

Virtual Machine Settings

Hardware

Options

Device	Summary
Memory	2 GB
Processors	1
Hard Disk (SCSI)	16 GB
Hard Disk 2 (SCSI)	5 GB
CD/DVD (SATA)	Auto detect
Network Adapter	NAT
Network Adapter 2	Custom (VMnet1)
Network Adapter 3	Custom (VMnet2)
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine:  MB

3 GB

2 GB

1 GB

512 MB

256 MB

128 MB

64 MB

32 MB

16 MB

8 MB

4 MB

Maximum recommended memory  
(Memory swapping may occur beyond this size.)  
6.1 GB

Recommended memory  
2 GB

Guest OS recommended minimum  
1 GB

The virtual machine must be powered off to reduce the amount of memory.

The virtual machine will use up to 768 MB of this memory for graphics memory. You can change this amount in the Display settings page.

Add...

Remove

OK

Cancel

Help

# CONFIGURAZIONE VMS



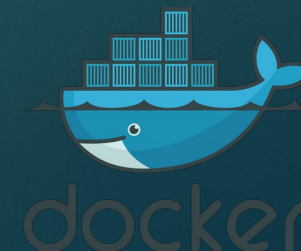
```
dav@ubuntu: ~  
dav@ubuntu:~$ sudo fdisk -l  
Disk /dev/fd0: 1.4 MiB, 1474560 bytes, 2880 sectors  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disklabel type: dos  
Disk identifier: 0x90909090  
  
Device      Boot      Start          End      Sectors  Size Id Type  
/dev/fd0p1             2425393296  4850786591  2425393296   1.1T 90 unknown  
/dev/fd0p2             2425393296  4850786591  2425393296   1.1T 90 unknown  
/dev/fd0p3             2425393296  4850786591  2425393296   1.1T 90 unknown  
/dev/fd0p4             2425393296  4850786591  2425393296   1.1T 90 unknown  
  
Disk /dev/sda: 15 GiB, 16106127360 bytes, 31457280 sectors  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disklabel type: dos  
Disk identifier: 0x960dd11f  
  
Device      Boot      Start          End      Sectors  Size Id Type  
/dev/sda1    *           2048  29456383  29454336   14G 83 Linux  
/dev/sda2                29458430  31455231  1996802    975M  5 Extended  
/dev/sda5                29458432  31455231  1996800    975M 82 Linux swap / Solaris
```

```
dav@ubuntu: ~  
dav@ubuntu:~$ sudo mkfs.ext4 /dev/sdb  
mke2fs 1.42.13 (17-May-2015)  
Creating filesystem with 1310720 4k blocks and 327680 inodes  
Filesystem UUID: 9befd926-2524-4f83-b861-ca5451058b0b  
Superblock backups stored on blocks:  
        32768, 98304, 163840, 229376, 294912, 819200, 884736  
  
Allocating group tables: done  
Writing inode tables: done  
Creating journal (32768 blocks): done  
Writing superblocks and filesystem accounting information: done
```

```
dav@ubuntu: ~  
dav@ubuntu:~$ sudo apt-get install \  
> apt-transport-https\  
> ca-certificates \  
> curl \  
> gnupg-agent \  
> software-properties-common
```

```
dav@ubuntu: ~  
dav@ubuntu:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt  
-key add -  
OK  
dav@ubuntu:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.  
com/linux/ubuntu $(lsb_release -cs) stable"  
dav@ubuntu:~$ sudo apt-get update  
Get:1 https://download.docker.com/linux/ubuntu xenial InRelease [66.2 kB]  
Hit:2 http://security.ubuntu.com/ubuntu xenial-security InRelease  
Hit:3 http://us.archive.ubuntu.com/ubuntu xenial InRelease  
Get:4 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages [13.  
1 kB]  
Hit:5 http://us.archive.ubuntu.com/ubuntu xenial-updates InRelease  
Hit:6 http://us.archive.ubuntu.com/ubuntu xenial-backports InRelease  
Fetched 79.3 kB in 0s (145 kB/s)  
Reading package lists... Done  
dav@ubuntu:~$  
  
File Edit View Search Terminal Help  
dav@vm1:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io  
  
dav@ubuntu:~$ sudo usermod -aG docker dav  
dav@ubuntu:~$ sudo systemctl restart docker
```

# CONFIGURAZIONE HD & Docker





## Modifica file hosts ed interfaces:

```
dav@vm1:~$ sudo nano /etc/hosts

127.0.0.1      localhost
127.0.1.1      vm1

192.168.75.11  vm1
192.168.75.12  vm2
192.168.75.13  vm3

10.0.0.11      vm1GFS
10.0.0.12      vm2GFS
10.0.0.13      vm3GFS
```

```
dav@vm1:~$ sudo nano /etc/network/interfaces

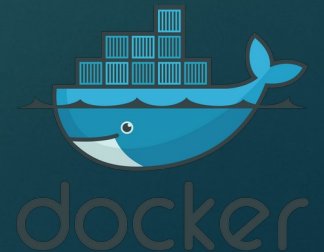
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto ens39
iface ens39 inet static
address 10.0.0.11
netmask 255.255.255.0
network 10.0.0.0
gateway 10.0.0.2

auto ens33
iface ens33 inet static
address 192.168.75.11
netmask 255.255.255.0
network 192.168.75.0
gateway 192.168.75.2
```

A questo punto, viene eseguita la clonazione della VM1 in VM2 e VM3. Vengono inoltre modificati i valori dei files `/etc/hosts`, `/etc/network/interfaces` ed `/etc/host` nelle due nuove VM.

## CONFIGURAZIONE IP ADDRESS



E' stato creato un nuovo swarm con il seguente cmd:

```
dav@vm1:~$ sudo docker swarm init --advertise-addr ens33
[sudo] password for dav:
Swarm initialized: current node (lv668l9aepejl2wb8f8dtlyv9) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-0hrrk1f9adtls4bc7egi2uifx7h176vb18v10ios
umdqnpcvo-934cpzv4iitd7gzw3uyhh0uv7 192.168.75.11:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow
the instructions.
```

In seguito è stato eseguito il seguente cmd nelle altre due VM:

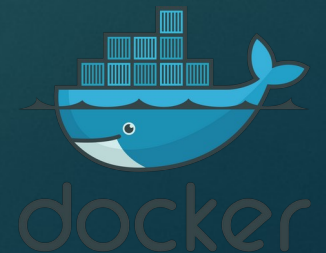
```
dav@vm3:~$ docker swarm join --token SWMTKN-1-4b6qkg951n22jc1f4fjfg5xcjks9jvquj1
i5h8l8ja0tasglz0-7ju8db7nwotmirxonsipi6hpg 192.168.1.172:2377
```

Controllo:

```
dav@vm1:~$ docker node ls
```

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGER STATUS	ENGINE VERSION
jmqqamktw541ngrkriuk6y6d1	ubuntu	Ready	Active		19.03.8
fbulkznkmv8e7k2ll7ip66we2 *	vm1	Ready	Active	Leader	19.03.8
gsiti4bg5rp3krz4u8c2q2gcw	vm3	Ready	Active		19.03.8

## CREAZIONE DOCKER SWARM





## Installazione e avvio di GlusterFs:

```
dav@vm3:~$ sudo apt -y install glusterfs-server glusterfs-client
```

## Start del service e peer con gli altri nodi:

```
dav@vm3:~$ sudo gluster peer probe vm1GFS
[sudo] password for dav:
peer probe: success. Host vm1GFS port 24007 already in peer list
dav@vm3:~$ sudo gluster peer probe vm2GFS
peer probe: success. Host vm2GFS port 24007 already in peer list
dav@vm3:~$ sudo gluster peer status
Number of Peers: 2

Hostname: vm2GFS
Uuid: 6c444dc7-498b-413e-8091-12e3233c463e
State: Peer in Cluster (Connected)

Hostname: vm1GFS
Uuid: 5714bbd6-6a82-434b-96a1-4c6ec669ba5c
State: Peer in Cluster (Connected)
```

## Setup delle cartelle di GlusterFS:

```
mkdir -p /gluster/bricks/1
mkdir -p /gluster/bricks/2
mkdir -p /gluster/bricks/3
```

```
echo '/dev/sdb /gluster/bricks/1 ext4 defaults 0 0' >> /etc/fstab
echo '/dev/sdb /gluster/bricks/2 ext4 defaults 0 0' >> /etc/fstab
echo '/dev/sde /gluster/bricks/3 ext4 defaults 0 0' >> /etc/fstab
```

```
dav@vm1:~$ sudo mount -a
```

# CONFIGURAZIONE GLUSTERFS



## Creazione e start del nuovo volume gfs:

```
dav@vm1:~$ sudo gluster volume create gfs replica 3 \  
> vm1GFS:/gluster/bricks/1/brick \  
> vm2GFS:/gluster/bricks/2/brick \  
> vm3GFS:/gluster/bricks/3/brick
```

```
dav@vm1:~$ sudo gluster volume start
```

## Controllo info e stato:

```
dav@vm1:~$ sudo gluster volume status gfs
```

[sudo] password for dav:

Status of volume: gfs

Gluster process	TCP Port	RDMA Port	Online	Pid
Brick vm1GFS:/gluster/bricks/1/brick	49152	0	Y	1743
Brick vm2GFS:/gluster/bricks/2/brick	49152	0	Y	1773
Brick vm3GFS:/gluster/bricks/3/brick	49152	0	Y	5256
NFS Server on localhost	N/A	N/A	N	N/A
Self-heal Daemon on localhost	N/A	N/A	Y	1416
NFS Server on vm2GFS	N/A	N/A	N	N/A
Self-heal Daemon on vm2GFS	N/A	N/A	Y	1656
NFS Server on vm3GFS	N/A	N/A	N	N/A
Self-heal Daemon on vm3GFS	N/A	N/A	Y	5278

Task Status of Volume gfs

There are no active volume tasks

## Setup security:

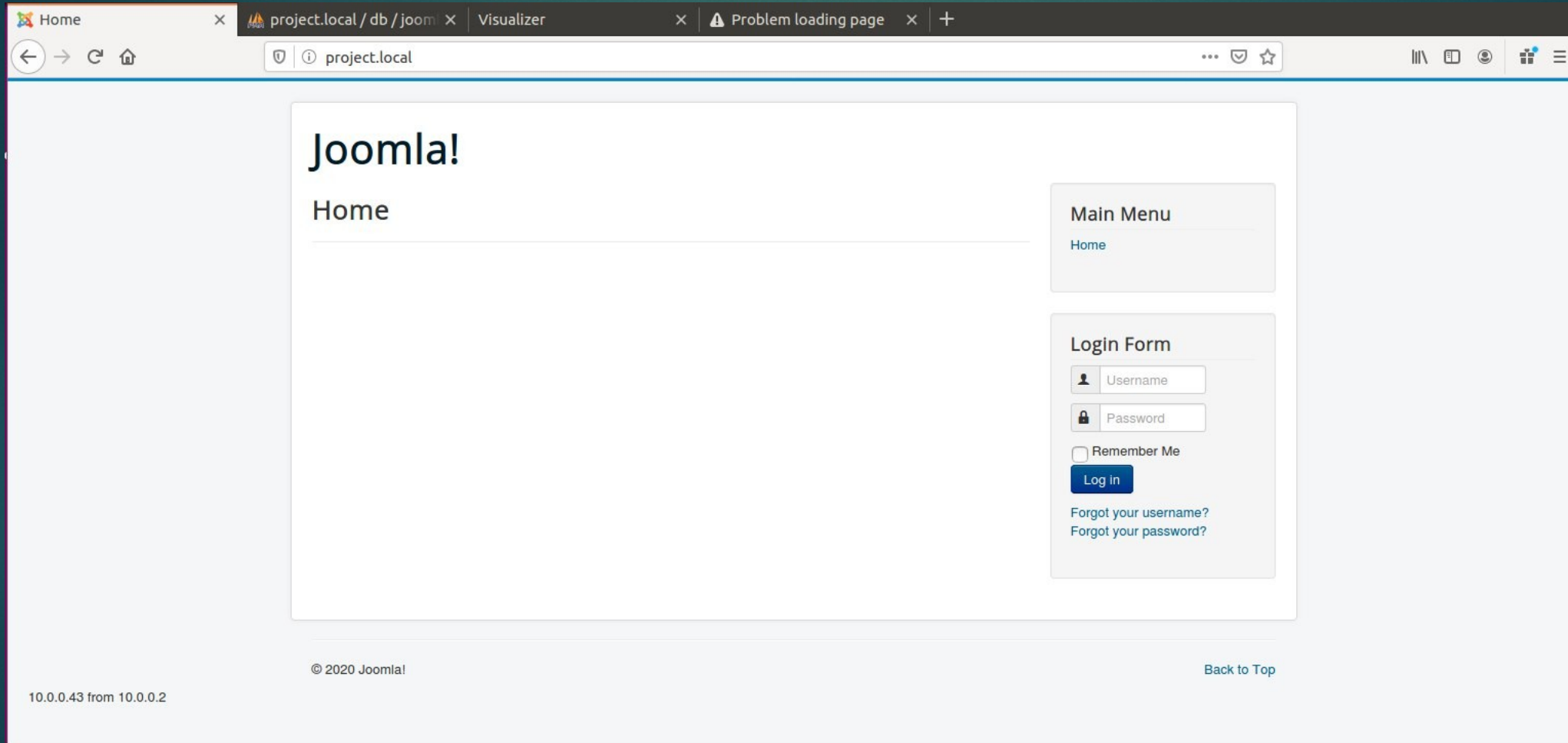
```
dav@vm1:~$ sudo gluster volume set gfs auth.allow 10.0.0.11,10.0.0.12,10.0.0.13  
volume set: success  
dav@vm1:~$ sudo gluster volume set gfs nfs.disable Off  
volume set: success
```

# CONFIGURAZIONE GLUSTERFS





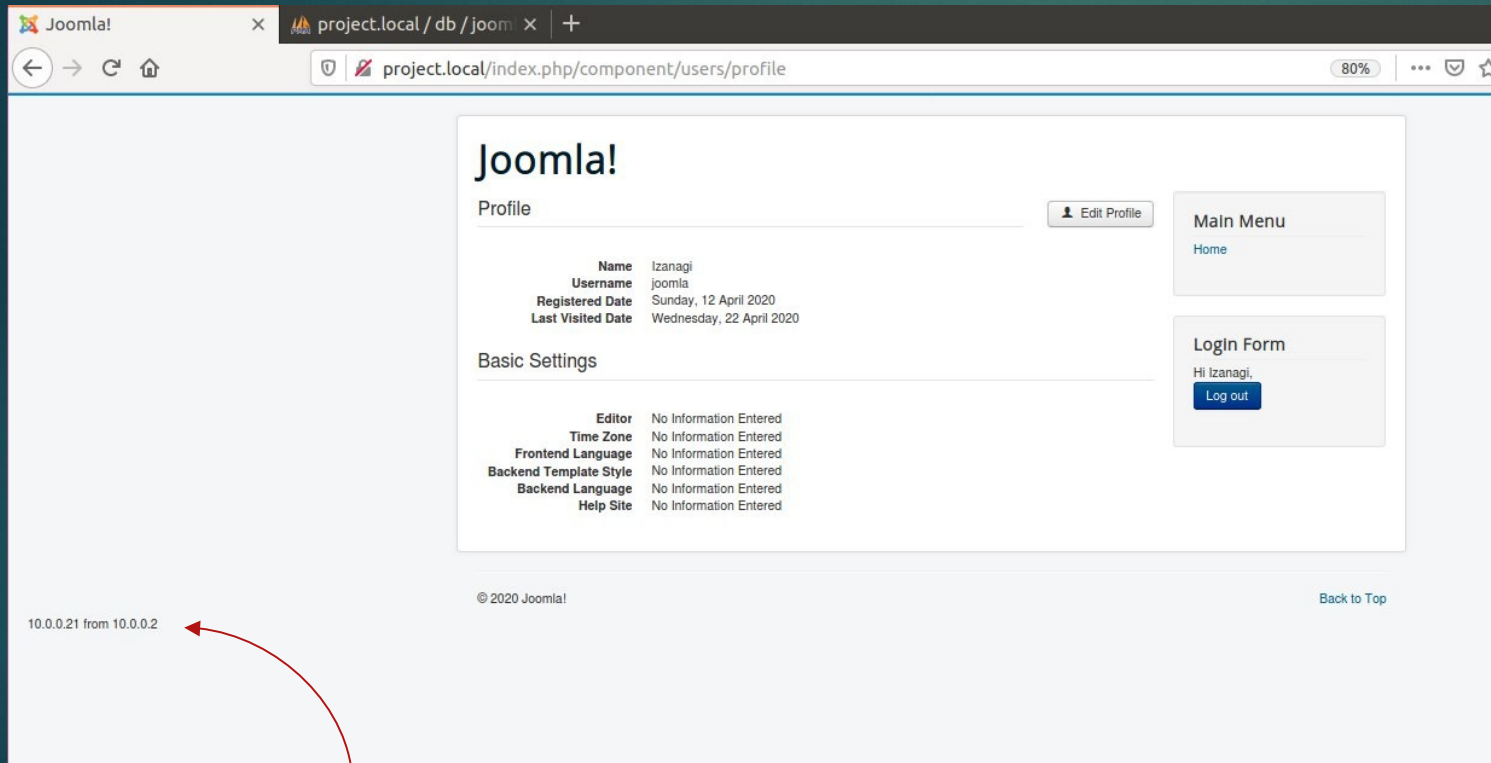
# JOOMLA



RISULTATO  
JOOMLA



# JOOMLA



```
version: "3"

services:
  joomla:
    image: myjoomla
    hostname: jo
    container_name: jo
    ports:
      - 8081:80
    depends_on:
      - db
    deploy:
      mode: replicated
      replicas: 3
    labels:
      - "com.docker.lb.hosts=project.local"#
      - "com.docker.lb.sticky_session_cookie=session"
      - "com.docker.lb.port=3001"
    volumes:
      - /gfs/joomla/:/var/www/html/
    networks:
      - intnet
#   - extnet
```

```
echo $_SERVER['SERVER_ADDR'];
echo " from ";
echo gethostbyaddr($_SERVER['REMOTE_ADDR']);
```



Joomla!



# MARIADB

The screenshot displays the phpMyAdmin web interface in a browser window. The address bar shows the URL `project.local/managedb/index.php`. The interface includes a sidebar with a tree view of databases: `information_schema`, `joomla`, `mysql`, and `performance_schema`. The main content area is divided into several panels:

- General settings:** Includes a "Change password" link and a "Server connection collation" dropdown set to `utf8mb4_unicode_ci`.
- Appearance settings:** Includes a "Language" dropdown set to "English", a "Theme" dropdown set to "pmahomme", and a "Font size" dropdown set to "82%". A "More settings" link is also present.
- Database server:** Lists server details:
  - Server: db via TCP/IP
  - Server type: MariaDB
  - Server connection: **SSL is not being used**
  - Server version: 10.4.12-MariaDB-1:10.4.12+maria~bionic - mariadb.org binary distribution
  - Protocol version: 10
  - User: root@192.168.200.8
  - Server charset: cp1252 West European (latin1)
- Web server:** Lists web server details:
  - Apache/2.4.38 (Debian)
  - Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - \$Id: 3591daad22de08524295e1bd073aceff11e6579 \$
  - PHP extension: mysqli, curl, mbstring
  - PHP version: 7.2.29
- phpMyAdmin:** Lists version and resource information:
  - Version information: 4.9.1, latest stable version: 4.9.5
  - Documentation
  - Official Homepage
  - Contribute
  - Get support
  - List of changes
  - License

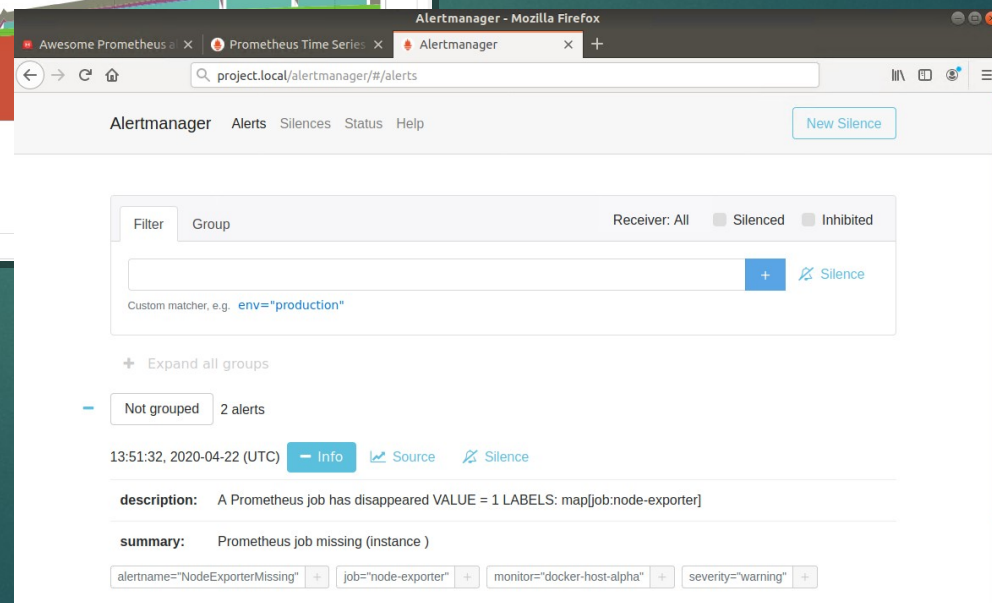
A notification at the bottom states: "A newer version of phpMyAdmin is available and you should consider upgrading. The newest version is 4.9.5, released on 2020-03-21." A console message at the very bottom reads: "phpMyAdmin configuration storage is not completely configured. Some extended features have been deactivated. Find out why."

RISULTATO  
MARIADB



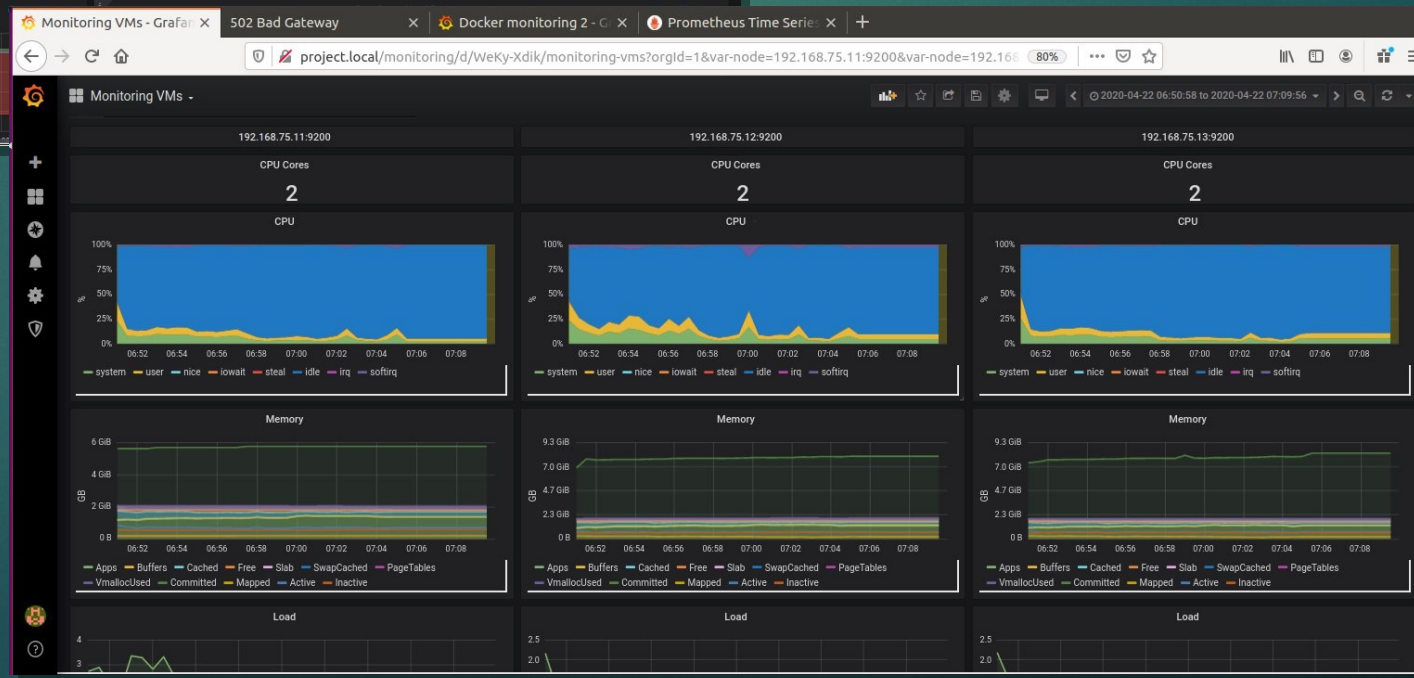
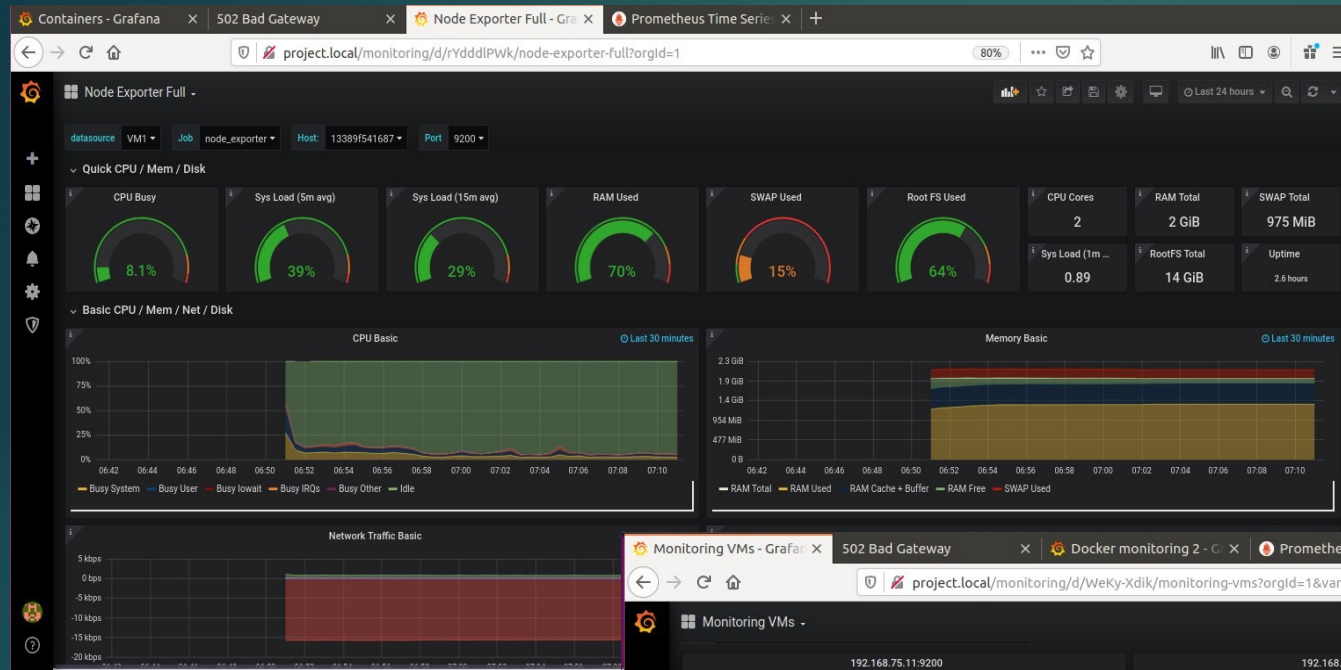
## Docker Containers

```
project_node-exporter.fbulknzkmv8e7k2l7ip66we2.8u...  
/4c72335abe5c2237cddc1ed1ca4b345293906da2ae51eb...  
  
project_prometheus.fbulknzkmv8e7k2l7ip66we2.7w1eu...  
/2fc9b854811b037ea4408a2fbde3bc298d8c1f80c8d8083...  
  
project_grafana.1_graghfbf7wdya5dfui5hw32zl /docker  
/61cacb3c00f3dcab10641720ac68b3052ab13065a42af1c...
```



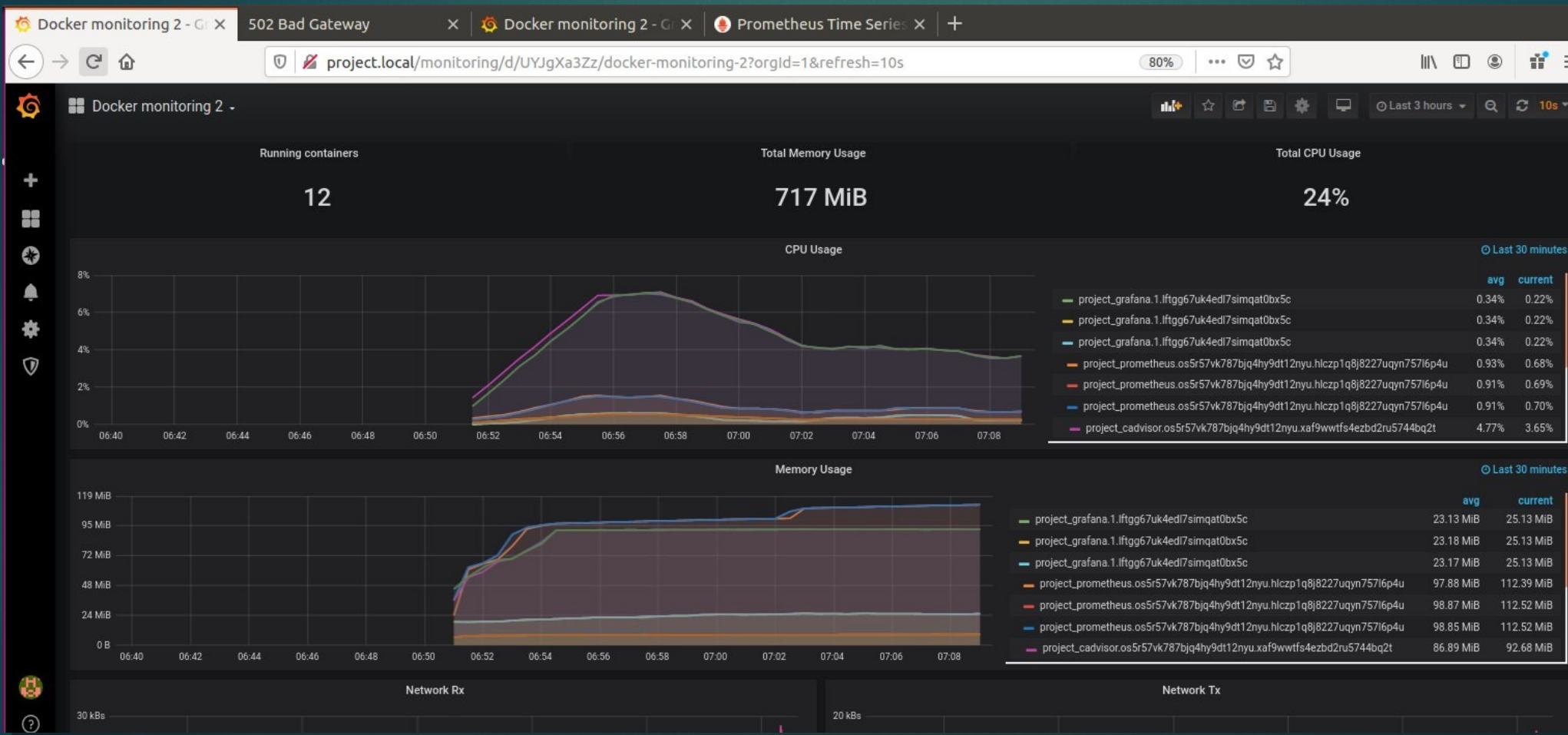


# MONITORING



Monitoring  
VMS





# Containers Monitoring





# MONITORING

File /gfs/prometheus/alertmanager/config.yml :

```
route:
  group_by: [Alertname]
  receiver: email-me

receivers:
- name: email-me
  email_configs:
  - to: davsens.ge@gmail.com
    from: senatori.projects@gmail.com
    smarthost: smtp.gmail.com:587
    auth_username: "senatori.projects@gmail.com"
    auth_identity: "senatori.projects@gmail.com"
    auth_password: "mypwd"
```

File /gfs/prometheus/prometheus.yml

```
rule_files:
- "alert.rules"

# alert
alerting:
  alertmanagers:
  - scheme: http
    static_configs:
    - targets:
      - "alertmanager:9093"
```

File docker-compose:

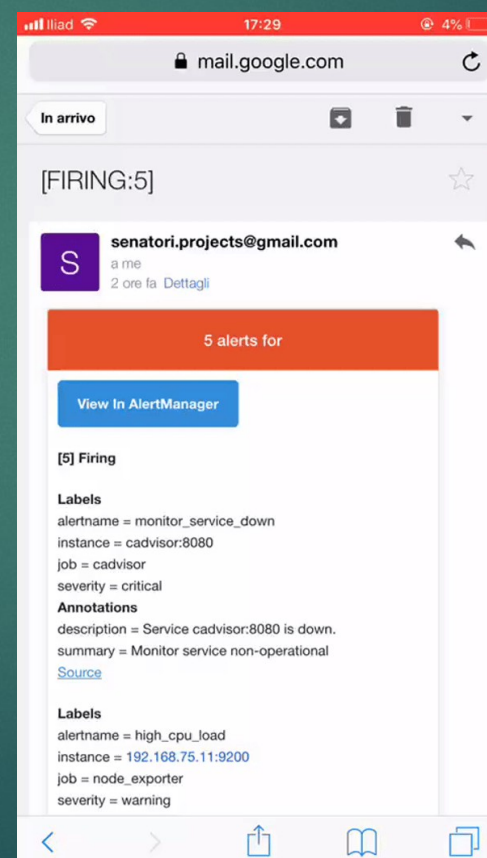
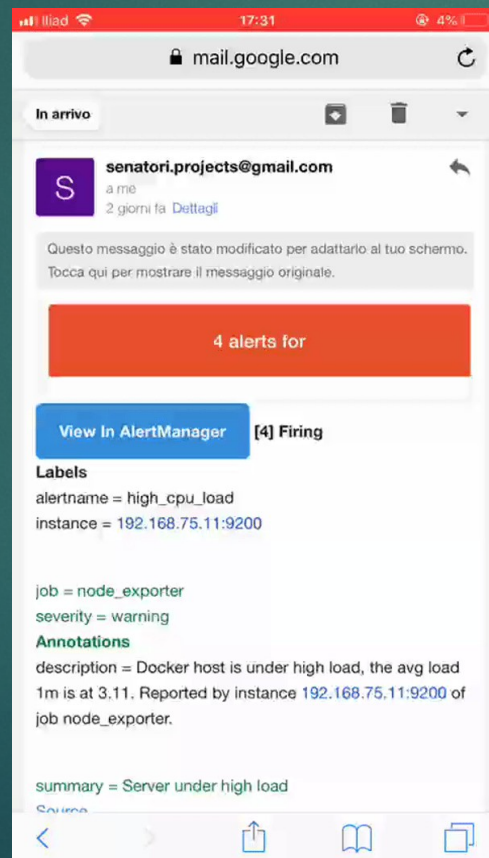
```
alertmanager:
  image: prom/alertmanager
  ports:
    - 9093:9093
  volumes:
    - /gfs/prometheus/alertmanager:/etc/alertmanager
  command:
    - '--config.file=/etc/alertmanager/config.yml'
    - '--storage.path=/alertmanager'
  deploy:
    placement:
      constraints:
        - node.role == manager
```

AlertManager



# MONITORING

<input type="checkbox"/> ☆ ➤	senatori.projects 20	[FIRING:4] - 4 alerts for View in AlertManager [4] Firing Labels alertname = high_cpu_load instan...
<input type="checkbox"/> ☆ ➤	senatori.projects 11	[FIRING:5] - 5 alerts for View in AlertManager [5] Firing Labels alertname = monitor_service_do...
<input type="checkbox"/> ☆ ➤	senatori.projects 2	[FIRING:3] - 3 alerts for View in AlertManager [3] Firing Labels alertname = monitor_service_do...
<input type="checkbox"/> ☆ ➤	senatori.projects 3	[FIRING:2] - 2 alerts for View in AlertManager [2] Firing Labels alertname = high_memory_load s...



AlertManager





# MONITORING

Home x project.local / db / joomla x Visualizer x Problem loading page x +

← → ↺ 🏠 🔒 project.local/visualizer/ 80% ⋮ ⌵ ⭐ 📄 📱 📧 ☰

vm1 manager 1.924G RAM x86_64/linux	vm2 worker 1.924G RAM x86_64/linux	vm3 worker 1.924G RAM x86_64/linux
<div><div>● project_cadvisor</div><div>image : cadvisor:latest@sha256:8153tag : latest@sha256:815386eb9a3updated : 22/4 10:43d05d81088c851448b7f1a2c1c171536state : running</div></div>	<div><div>● project_node-exporter</div><div>image : node-exporter:latest@sha25tag : latest@sha256:a2f29256e3cc9cmd : --path.procfs=/host/proc...pathupdated : 22/4 10:4309ab53c3e1397d28050b2f1de12ea0state : running</div></div>	<div><div>● project_node-exporter</div><div>image : node-exporter:latest@sha25tag : latest@sha256:a2f29256e3cc9cmd : --path.procfs=/host/proc...pathupdated : 22/4 10:439538493fb20fbd659ac7c8aca34d0ffcstate : running</div></div>
<div><div>● project_prometheus</div><div>image : prometheus:latest@sha256tag : latest@sha256:42d2393cd7192cmd : --config.file=/etc/prometheus/updated : 22/4 10:43b98a55883cd2d8865d13a92109528bstate : running</div></div>	<div><div>● registry</div><div>image : registry:2@sha256:7d081088tag : 2@sha256:7d081088e4bf0e32aupdated : 22/4 10:465aba225a89fb86894137387e8a8acstate : running</div></div>	<div><div>● project_joomla</div><div>image : myjoomla:latesttag : latestupdated : 22/4 10:43afe7c485c229e7f8326812de3b07b3state : running</div></div>
<div><div>● project_grafana</div><div>image : grafana:latest@sha256:d28tag : latest@sha256:c62889a0ee3c15updated : 22/4 10:433c993a6c5a5f9393a4ed37d1190026state : running</div></div>	<div><div>● project_db</div><div>image : mariadb:latest@sha256:c11tag : latest@sha256:c1182046a87fdupdated : 22/4 10:45a301d052110d23fb98dfecdc87c7211state : running</div></div>	<div><div>● project_cadvisor</div><div>image : cadvisor:latest@sha256:815tag : latest@sha256:815386eb9a3updated : 22/4 10:434e2324d43727b014edbf0e2db3843state : running</div></div>
<div><div>● project_reverse</div><div>image : nginx:latest@sha256:d81f01tag : latest@sha256:d81f0109557493updated : 22/4 10:43</div></div>	<div><div>● project_phpmyadmin</div><div>image : myphpmyadmin:latesttag : latestupdated : 22/4 10:43</div></div>	<div><div>● project_joomla</div><div>image : myjoomla:latesttag : latestupdated : 22/4 10:43</div></div>

Visualizer

# RISPOSTE E BONUS

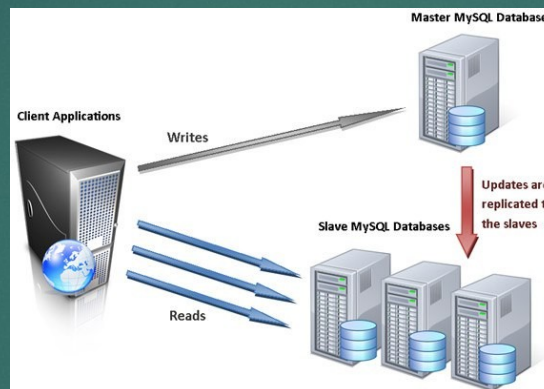
## Split-brain



Soluzioni per attenuare:

- 1) Replica 3 volume
- 2) Arbitro

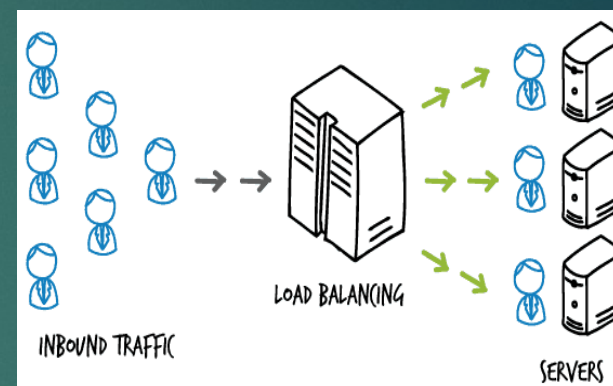
## DB replicato



MySQL group replication (GR) è un server MySQL plugin che permette di creare repliche high-availability e fault-tolerant.

- Single primary: solo un server alla volta accetta update.
- Multi primary: tutti i servers accettano update.

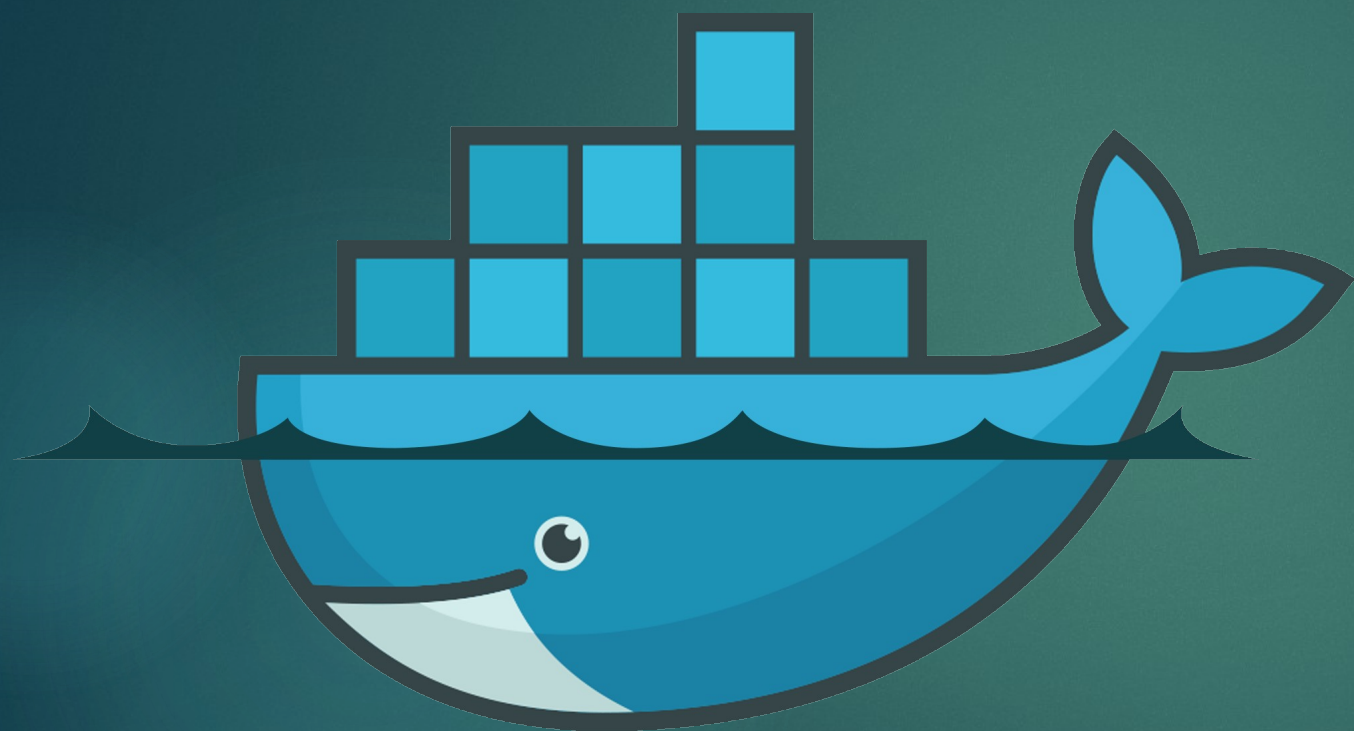
## Load Balancing



Implementato attraverso Nginx

Sviluppi futuri?  
-Manager Fault tollerant





GRAZIE PER  
L'ATTENZION  
E