

Raspi Docker – Portainer

Das Wichtigste der Benutzer beim pi muss unbedingt pi sein ansonsten Funktioniert das ganze nicht!!!!!!

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

```
sudo curl -fsSL get.docker.com -o get-docker.sh && sh get-docker.sh
```

```
sudo docker volume create portainer_data
```

```
sudo docker start portainer
```

Achtung den nächsten Befehl nicht kopieren.

Der Befehl muß in der ganzen länge ausgeschrieben werden d.h. nach **always -v auf keinen Fall eine nächste Zeile beginnen sondern den Befehl in der vollen Länge ausführen. Am besten Händisch eingeben und nicht kopieren!!!**

```
sudo docker run -d -p 9000:9000 --name portainer --restart always -v  
/var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer
```

```
sudo docker start portainer
```

<http://192.168.178.70:9000>

Benutzer: admin

Password: administrator123

(Password muss 12 Zeichen verlangen, wenn nur 8 Zeichen verlangt werden Funktioniert das ganze nicht!)

Benutzer erstellen anklicken.

Wen der Start nicht Funktioniert.

```
sudo docker restart portainer
```

Bitte erstellen Sie den anfänglichen Administratorbenutzer.

Nutzername

Passwort

Passwort bestätigen

✗ Das Passwort muss mindestens 8 Zeichen lang sein

[+Benutzer erstellen Benutzer erstellen](#)

Erfassung anonymer Statistiken zulassen. Weitere Informationen dazu finden Sie in unserem [Datenschutzrichtlinie](#).

Die erste Variante wählen: Docker und verbinden.

Docker erstes Icon wählen und verbinden.

Verbinden Sie Portainer mit der Containerumgebung, die Sie verwalten möchten.

Docker
Verwalten Sie die lokale Docker-Umgebung

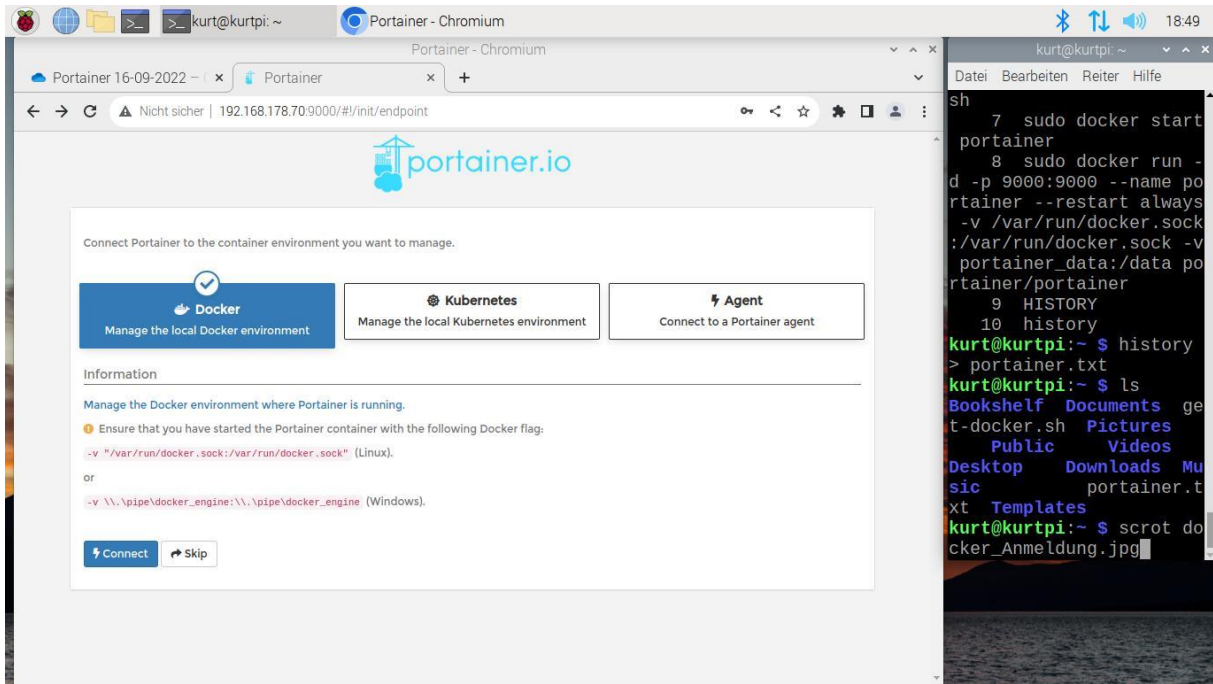
Kubernetes
Verwalten Sie die lokale Kubernetes-Umgebung

Agent
Stellen Sie eine Verbindung zu einem Portainer-Agenten her

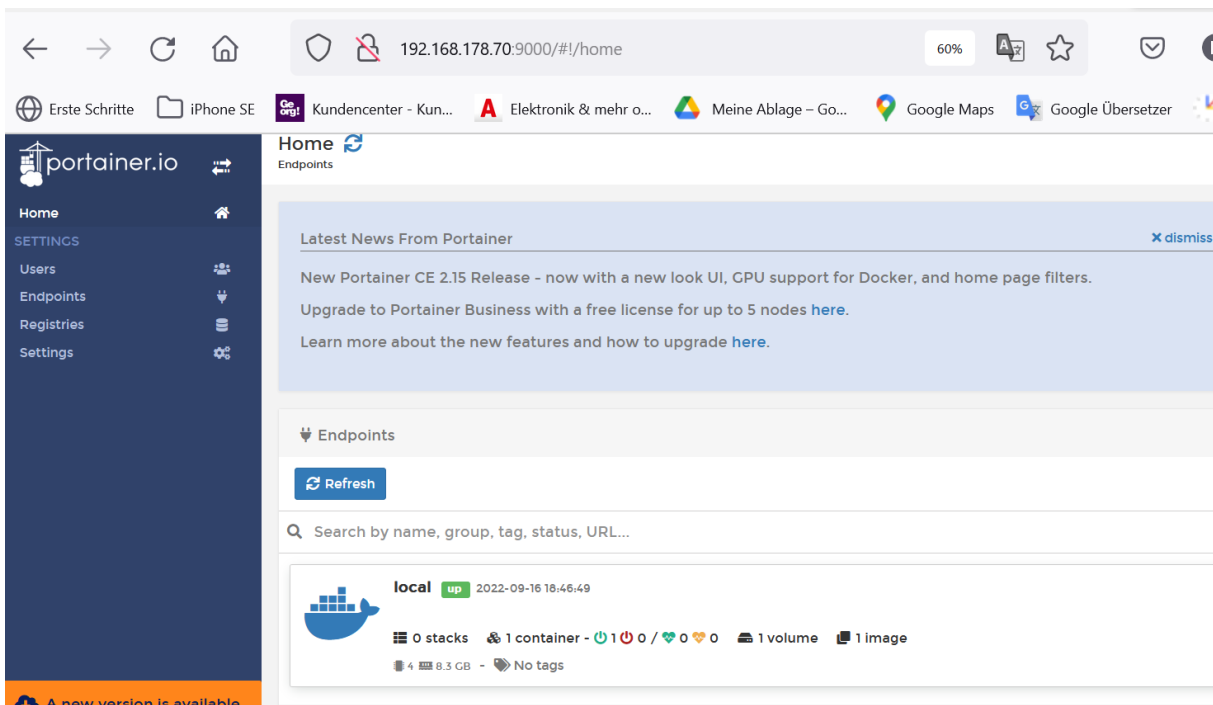
Information

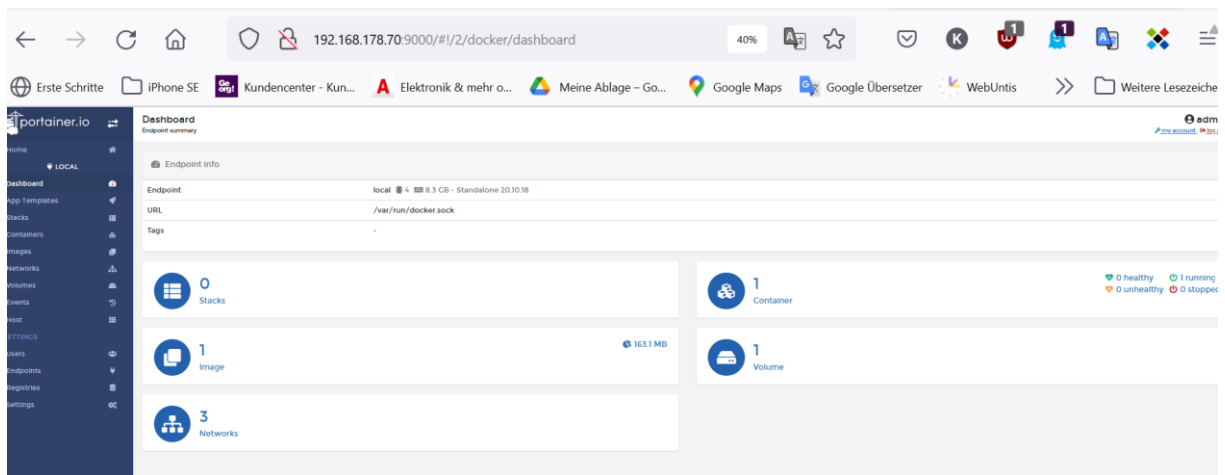
Verwalten Sie die Kubernetes-Umgebung, in der Portainer ausgeführt wird.

[Verbinden](#) [Überspringen](#)

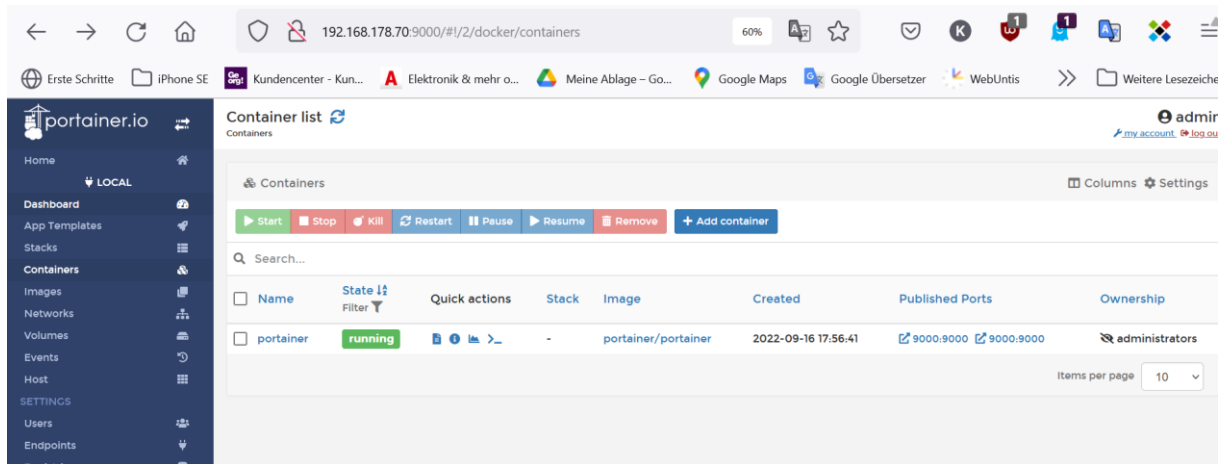


Dann Local up – Container anklicken





Container und Add Container: mariadb



Name: mariadb

Image: linuxserver/mariadb

Advance mode

Allways pull the image on

Publish a new network port.

Host: 3306

Container: 3306

TCP

Enable Access Contoll: on

Administrators

Auto Remove: off

Volumes: Map additional volume

Container:

/config

bind

Host:

/srv/dev-disk-by-uuid-82011d01/config/mariadb

ENV

Add environment variable 7 mal anklicken!

ENV

PUID=998
 PGID=100
 MYSQL_ROOT_PASSWORD=123456
 TZ=Europe/London
 MYSQL_DATABASE=db_wordpress
 MYSQL_USER=pidb
 MYSQL_PASSWORD=password

Advanced container settings

Command & logging Volumes Network **Env** Labels Restart policy Runtime & Resources Capabilities

Environment variables
 These values will be applied to the container when deployed

Advanced mode
 Switch to advanced mode to copy & paste multiple variables

+ Add an environment variable Load variables from .env file

name	PUID	value	Remove value	Trash	
name	PUID	value	998	- Remove value	🗑️
name	PGID	value	100	- Remove value	🗑️
name	MYSQL_ROOT_PASSWORD	value	123456	- Remove value	🗑️
name	TZ	value	Europe/London	- Remove value	🗑️
name	MYSQL_DATABASE	value	db_wordpress	- Remove value	🗑️
name	MYSQL_USER	value	pidb	- Remove value	🗑️
name	MYSQL_PASSWORD	value	password	- Remove value	🗑️

Restart Policies: Unless stopped

Advanced container settings

Command & logging Volumes Network Env Labels **Restart policy** Runtime & Resources Capabilities

Restart policy **Never** Always On failure Unless stopped

Deploy the container

portainer.io COMMUNITY EDITION

Containers

Container list

Containers Search... Start Stop Kill Restart Pause Resume Remove + Add container

Name	State	Quick Actions	Stack	Image	Created	IP Address	GPUs	Published Ports	Owner
mariadb	running	🔍 ⏸️ ⏹️ 🔄	-	linuxserver/mariadb:latest	2022-09-17 16:44:50	172.17.0.3	none	🔗3306:3306	admin
portainer	running	🔍 ⏸️ ⏹️ 🔄	-	portainer/portainer-ce	2022-09-17 16:13:03	172.17.0.2	none	🔗8000-8000 🔗9000-9000	admin

Items per page 10

Nächsten Container erstellen.

Die gleichen Einstellungen wie zuvor.

Add new container: **myphpadmin**

Name:

myphpadmin

Image:

ebspace/armhf-phpmyadmin

advanced mode

Always pull the image: on

Network configuration

Manual network port publishing

publish a new network port

host: 8085

container: 80

tcp

Access Control

Enable access control: on

Administrators

Actions

Auto remove:off

The screenshot shows the Portainer.io web interface for configuring a new container. The container name is 'myphpadmin'. The registry is set to 'DockerHub (anonymous)' and the image is 'docker.io/ebspace/armhf-phpmyadmin'. The 'Advanced mode' is enabled, and 'Always pull the image' is checked. Under 'Network ports configuration', 'Manual network port publishing' is enabled, and a new port is being published: host 8085 to container 80, using TCP. Under 'Access control', 'Enable access control' is checked, and the container is restricted to administrators. The 'Actions' section shows 'Auto remove' is disabled and a 'Deploy the container' button. The 'Advanced container settings' section is partially visible, showing 'Volumes' as the active tab.

Volumes

+map additional volume

Container

/etc/phpmyadmin/config.user.inc.php

Bind

Host

/srv/dev-disk-by-uuid-82011d01/config/phpMyAdmin

Administrators

I want to restrict the management of this resource to administrators only

Restricted

I want to restrict the man

Actions

Auto remove ?



Deploy the container

Advanced container settings

Command & logging

Volumes

Network

Env

Labels

Restart p

Volume mapping + map additional volume

container	/etc/phpmyadmin/config.user.inc.php	Volume	Bind	
→ host	/srv/dev-disk-by-uuid-82011d01/config/phpMyAdmin	Writable	Read-only	

ENV

PMA_ARBITRARY = 1
 PMA_HOST = pikurt
 PMA_VERBOSE = wordpress
 PMA_PORT = 3306
 PMA_USER = pidb
 PMA_PASSWORD = password

Advanced container settings

Command & logging

Volumes

Network

Env

Labels

Restart policy

Runtime & Resources

Capabilities

Environment variables

These values will be applied to the container when deployed

Advanced mode

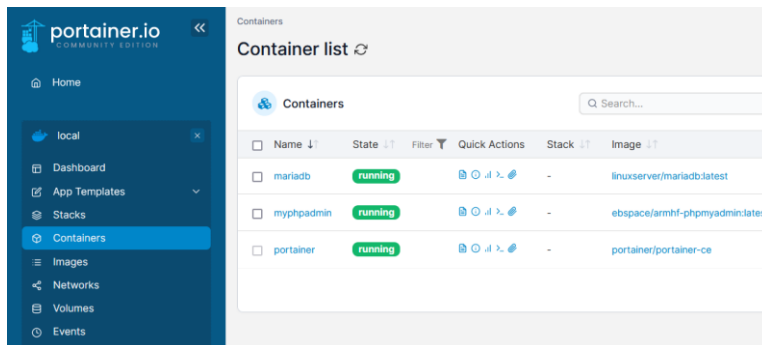
Switch to advanced mode to copy & paste multiple variables

+ Add an environment variable Load variables from .env file

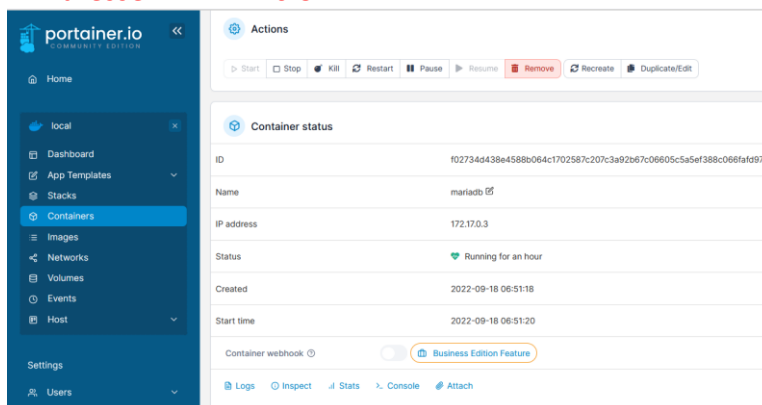
name	PMA_ARBITRARY	value	1	Remove value	
name	PMA_HOST	value	pikurt	Remove value	
name	PMA_VERBOSE	value	wordpress	Remove value	
name	PMA_PORT	value	3306	Remove value	
name	PMA_USER	value	pidb	Remove value	
name	PMA_PASSWORD	value	password	Remove value	

Dann Restart policy
 Unless stopped
 Deploy the container

IP-mariadb herausfinden
Container: Name – mariadb anklicken



IP-Adresse = 172.17.0.3



Add new container: **wordpress**

Name: wordpress

Image: wordpress:latest
advanced mode

Always pull the image: on

Network configuration

Manual network port publishing

publish a new network port

host: 8081

container: 80

tcp

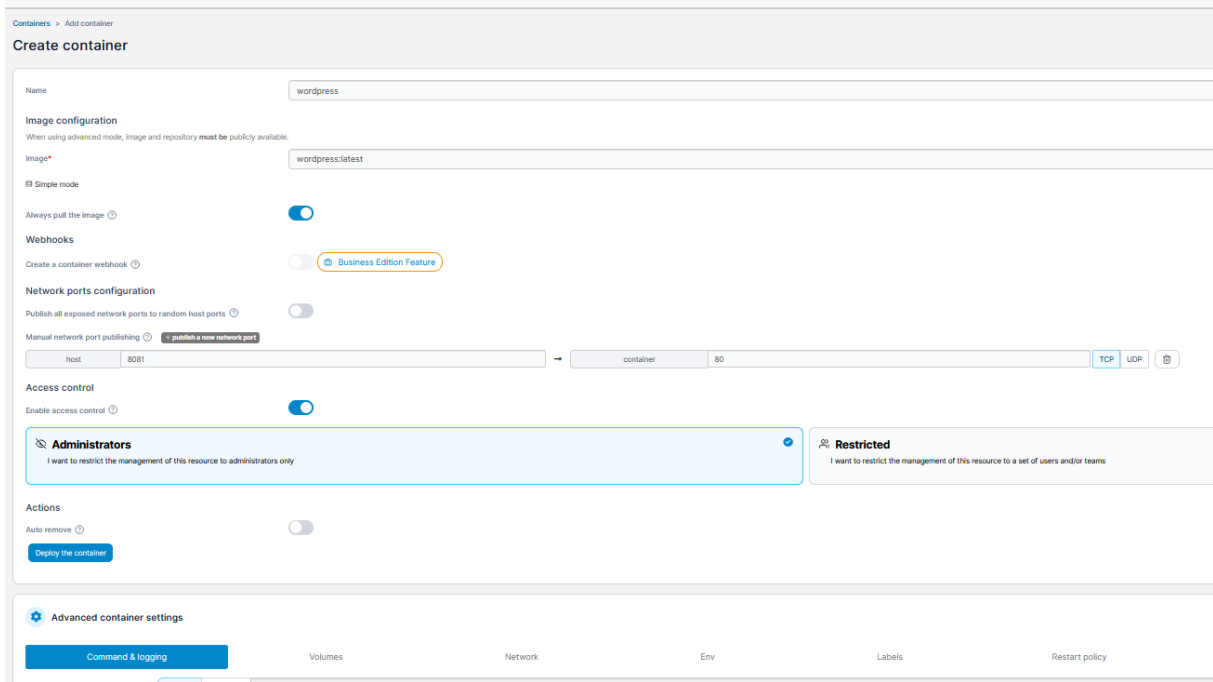
Access Control

Enable access control: on

Administrators

Actions

Auto remove:off



Volumes

+map additional volume

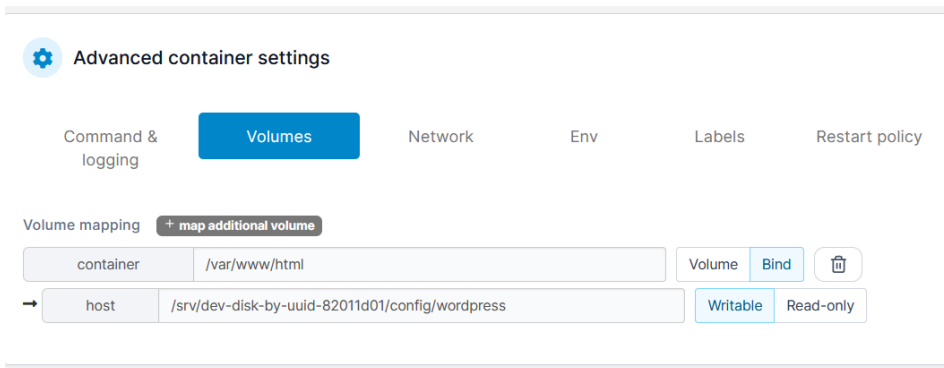
Container

/var/www/html

Bind

Host

/srv/dev-disk-by-uuid-82011d01/config/wordpress



ENV

WORDPRESS_DB_HOST = IP-mariadb:3306

IP-mariadb in diesem Fall: 172.17.0.3

WORDPRESS_DB_USER = pidb

WORDPRESS_DB_PASSWORD = password

WORDPRESS_DB_NAME = db_wordpress

Advanced container settings

Command & logging | Volumes | Network | **Env** | Labels | Restart policy | Runtime & Resources | Capabilities

Environment variables

These values will be applied to the container when deployed

Advanced mode
Switch to advanced mode to copy & paste multiple variables

+ Add an environment variable | Load variables from .env file

name	WORDPRESS_DB_HOST	value	172.17.0.3:3306	Remove value	🗑️
name	WORDPRESS_DB_USER	value	pidb	Remove value	🗑️
name	WORDPRESS_DB_PASSWORD	value	password	Remove value	🗑️
name	WORDPRESS_DB_NAME	value	wordpress	Remove value	🗑️

Restart policy

Unless stopped

Deploy the container

Übersicht der Installierten Container

Containers

Container list [↻](#) admin

Containers ▶ Start ⏏ Stop 🗑 Kill ↺ Restart ⏸ Pause ▶ Resume 🗑 Remove + Add container ⌵

Name	State	Filter	Quick Actions	Stack	Image	Created	IP Address	GPUs	Published Ports	Ownership
❑ mariadb	running		🔍 ⏏ ▶ 🗑	-	linuxserver/mariadb:latest	2022-09-18 06:51:18	172.17.0.3	none	🔗3306:3306	administrators
❑ myphpadmin	running		🔍 ⏏ ▶ 🗑	-	ebspace/armhf-phpmyadmin:latest	2022-09-18 07:22:37	172.17.0.4	none	🔗8085:80	administrators
❑ portainer	running		🔍 ⏏ ▶ 🗑	-	portainer/portainer-ce	2022-09-17 18:33:52	172.17.0.2	none	🔗8000:8000 🔗9000:9000	administrators
❑ wordpress	running		🔍 ⏏ ▶ 🗑	-	wordpress:latest	2022-09-18 08:10:01	172.17.0.5	none	🔗8081:80	administrators

Items per page 10

Start der Wordpress Installation mit IP-dresse;8081 (meine IP=192.168.178.70)

mariadb container MYSQL Datenbanken überprüfen

Container list [↻](#)

Containers ▶ Start ⏏ Stop 🗑 Kill ↺ Restart ⌵

Name	State	Filter	Quick Actions	Stack	Image	Created
❑ mariadb	running		🔍 ⏏ ▶ 🗑	-	linuxserver/mariadb:latest	2022
❑ myphpadmin	running		🔍 ⏏ ▶ 🗑	-	ebspace/armhf-phpmyadmin:latest	2022
❑ portainer	running		🔍 ⏏ ▶ 🗑	-	portainer/portainer-ce	2022

Console

Use custom command

Mysql -u root -p

Password: 123456

SHOW DATABASES;

Strichpunkt bei SQL Befehlen immer setzen!

```

Containers > mariadb > Console
Container console
Execute
Exec into container as default user using command mysql -u root -p Disconnect

Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 5
Server version: 10.5.17-MariaDB-log MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| db_wordpress |
| information_schema |
| mysql |
| performance_schema |
+-----+
4 rows in set (0.003 sec)

MariaDB [(none)]>

```