



KN04: Docker Compose

A) Docker Compose: Lokal (60%)

Teil a) Verwendung von Original Images

Teil b) Verwendung Ihrer eigenen Images

B) Docker Compose: Cloud (40%)

KN04: Docker Compose

Beachten Sie die [allgemeinen Informationen zu den Abgaben](#).

Grundlagen für diesen Auftrag:

- [TBZ: Docker Compose](#)
- [Docker: Docker Compose Reference](#)

A) Docker Compose: Lokal (60%)

Teil a) Verwendung von Original Images

In KN02 mussten Sie für die beiden Container, zwei Images starten mit den jeweiligen Befehlen. Docker Compose erlaubt Ihnen beide Images in einem Rutsch zu starten mit Hilfe einer YAML-Datei. [Das Format von YAML kennen Sie bereits aus m346](#).

Erstellen Sie die Docker Compose (YAML) Datei, um Ihre Umgebung wieder lokal auf Ihrem Computer zum Laufen zu kriegen unter Einhaltung der folgenden Bedingungen:

- Für die Datenbank verwenden Sie alle Anweisungen direkt in der Docker Compose Datei, benutzen also kein Dockerfile im Hintergrund. Verwenden Sie das **image mariadb:latest** und setzen Sie die notwendige Konfiguration in der YAML-Datei
- Für den Webserver verwenden Sie ein Dockerfile und weisen Docker Compose an dieses zu verwenden. Verwenden Sie Ihre Datei aus KN02.
- Geben Sie beiden Containern die Namen **m347-kn04a-web** und **m347-kn04a-db**.
- Erstellen Sie ein Netzwerk in der Docker Compose Konfiguration, so dass die beiden Server kommunizieren können mit den Eigenschaften:
 - subnet: 172.10.0.0/16
 - ip_range: 172.10.5.0/24
 - gateway: 172.10.5.254

Schauen Sie nach was der Befehl `docker compose up` macht. Er ist eine Zusammenfassung für mehrere Befehle. Schlagen Sie diese nach und dokumentieren Sie alle

Abgaben:

- Screenshot der Seite info.php. Scrollen Sie dabei zuerst runter bis die Felder **REMOTE_ADDR** und **SERVER_ADDR** sichtbar sind
- Screenshot der Seite db.php. Sie zeigen, dass beide Images im gleichen Netzwerk sind.
- Docker-Compose File (yaml-Datei)
- Dockerfile für Webserver
- Liste der Befehle, die `docker compose up` ausführt und deren Erklärungen

Teil b) Verwendung Ihrer eigenen Images

Falls Sie Ihre Images aus KN02 noch nicht im Repository publiziert haben, tun Sie dies jetzt. Verwenden Sie nun Ihre publizierten Images in der Docker Compose Datei. Sie benötigen also *kein Dockerfile* mehr für den Webserver. Räumen Sie Ihre Docker Compose Datei auf. Entfernen Sie was Sie nicht mehr benötigen, Ändern Sie was notwendig ist. **Verwenden Sie einen anderen IP-Range als eben.**

Führen Sie nun folgende Schritte durch:

- Rufen Sie die Seite info.php auf und erstellen Sie den Screenshot wieder mit den beiden IPs sichtbar.
- Rufen Sie die Seite db.php auf (**es wird ein Fehler geben!**) und erstellen Sie den Screenshot.
- Erklären Sie wieso Sie diesen Fehler sehen! Erklären Sie auch wie man dies lösen kann.

Abgaben:

- Screenshots der beiden Seiten
- Docker Compose Datei (yaml)
- Erklärung wieso der Fehler auftritt

B) Docker Compose: Cloud (40%)

Wiederholen Sie A) nun in der Cloud. Bauen Sie Ihre Lösung in Docker Compose ein. Sie können dabei von Teil a oder Teil b ausgehen (wobei Teil a wahrscheinlich einfacher ist). Nutzen Sie Cloud-Init um Ihre Anwendungen zu installieren. Bedingungen/Bemerkungen:

- Die gelernten Cloud-Init-Anweisungen aus m346 reichen aus, um die Applikation zu installieren. [Sie kriegen hier ein vorbereitetes cloud-init, welches Docker installiert.](#) Sie müssen die Bereiche **write_files** und **runcmd** erweitern.
- Fügen Sie Ihren und den [öffentlichen Schlüssel der Lehrperson](#) hinzu, so dass die Lehrperson auf Ihren Server zugreifen kann.
- Evtl. hilft es, wenn Sie zuerst alle Schritte manuell durchführen (mit Bash-Befehlen), bevor Sie das Cloud-Init-Skript erstellen. Am Schluss muss aber Ihr Cloud-Init Script korrekt ausführbar sein - ohne zusätzliche manuelle Schritte.
- Sie können in der Log-Datei (/var/log/cloud-init-output.log) nachschauen, ob die Installation durch ist oder nicht. Am Schluss sollte der Text aus der Cloud-init Datei (final_message) angezeigt werden.

Abgaben

- Screenshots der aufgerufenen Seiten, inkl. sichtbarer URLs. Bei info.php sollen wieder die IPs sichtbar sein!
- Cloud-Init Datei, die ja alle anderen Dateien enthalten sollte.