ONDERWIJSADVIES
LEREN MET ACTIE

Minecraft Server op Linux Ubuntu met Docker

Onderwijsmateriaal voor studenten Systeem- & Netwerkbeheer / Applicatiebeheer
Inclusief Bijlagen voor Bedrock Edition & Cross-Play functionaliteit

1. Inleiding en Doelstelling

In dit practicum leer je hoe je op een professionele en reproduceerbare wijze een Minecraft-server host op een Ubuntu Linux-omgeving met behulp van containerisatie (Docker & Docker Compose). Door gebruik te maken van containers blijft het besturingssysteem schoon, zijn updates triviaal uit te voeren en kunnen serverbestanden eenvoudig geback-upt worden via persistent storage (volumes). De basisinstructie richt zich op de Java Edition (PaperMC), waarna de bijlagen ingaan op specifieke Bedrock-ondersteuning en Cross-Play.

2. Vereisten & Installatie van Docker

Zorg ervoor dat je Ubuntu-server up-to-date is en dat Docker inclusief Docker Compose zijn geïnstalleerd. Voer de onderstaande commando's uit in de terminal:

```
# 1. Pakketbronnen bijwerken en Docker installeren
sudo apt update
sudo apt install docker.io docker-compose-v2 -y

# 2. Docker daemon starten en inschakelen bij opstarten
sudo systemctl enable --now docker

# 3. (Optioneel) Voeg je huidige gebruiker toe aan de docker-groep
sudo usermod -aG docker $USER
```

> **Tip voor studenten**
> Na het toevoegen van je gebruiker aan de docker-groep dien je eenmaal uit en opnieuw in te loggen om Docker-opdrachten zonder 'sudo' te draaien.

© 2026 Onderwijsadvies Leren met Actie

3. Stapsgewijze Implementatie (Java Edition / PaperMC)

Stap 1: Werkmap aanmaken

Maak een specifieke projectmap aan waarin de configuratie en data van de server worden opgeslagen:

```
mkdir -p ~/minecraft-server && cd ~/minecraft-server
```

Stap 2: Docker Compose configuratie opstellen

We gebruiken de breed ondersteunde image `itzg/minecraft-server` en kiezen voor 'PaperMC' server-software, die sterk geoptimaliseerd is. Maak het configuratiebestand aan met:

```
nano docker-compose.yml
```

Plak de volgende YAML-definitie in het bestand en sla op (Ctrl+O, Enter, Ctrl+X):

```
services:
  mc:
    image: itzg/minecraft-server:latest
    container_name: mc-server
    restart: unless-stopped
    ports:
      - "25565:25565"        # Standaard Java TCP-poort
    environment:
      EULA: "TRUE"           # Vereist: accepteer de Minecraft EULA
      MEMORY: "4G"           # Toegewezen RAM-geheugen voor Java
      TYPE: "PAPER"          # Performancegerichte server core
    volumes:
      - ./data:/data         # Persistent storage voor gamewereld en instellingen
```

Stap 3: Container starten

Start de container op de achtergrond (detached mode):

```
docker compose up -d
```

Stap 4: Initialisatie en logboeken monitoren

De server downloadt bij de eerste start automatisch de benodigde componenten en genereert de gamewereld. Volg dit proces live via de logging:

```
docker logs -f mc-server
```

Zodra de melding **[Server thread/INFO]: Done (...)! For help, type "help"** verschijnt, is de server operationeel. Druk op **Ctrl + C** om de logging te verlaten.

© 2026 Onderwijsadvies Leren met Actie

Bijlage A: Zuivere Minecraft Bedrock Server

Voor mobiele apparaten (iOS/Android), Windows 10/11 Bedrock Edition en consoles

Indien je specifiek een officiële Bedrock Dedicated Server wilt hosten (in plaats van Java), gebruikt men de image `itzg/minecraft-bedrock-server`. Belangrijk netwerkdetail: Bedrock communiceert standaard over UDP-poort 19132.

Voorbeeld `docker-compose.yml` voor Bedrock:

```
services:
  bedrock:
    image: itzg/minecraft-bedrock-server:latest
    container_name: bedrock-server
    restart: unless-stopped
    ports:
      - "19132:19132/udp"    # Let op: Bedrock gebruikt UDP (geen TCP)!
    environment:
      EULA: "TRUE"           # Vereist
      SERVER_NAME: "Mijn School Bedrock Server"
      GAMEMODE: "survival"
      DIFFICULTY: "normal"
    volumes:
      - ./bedrock-data:/data
```

> **Netwerkaandachtspunt**
> Bij port forwarding in de router of firewall moet voor Bedrock het UDP-protocol worden gekozen in plaats van TCP!

© 2026 Onderwijsadvies Leren met Actie

Bijlage B: Cross-Play (Java & Bedrock op 1 Server via GeyserMC)

Ideaal voor klaslokalen waarin studenten gemengde apparaten gebruiken

In een onderwijsomgeving hebben studenten vaak verschillende versies: sommige spelen op PC met Java Edition, terwijl anderen met een telefoon, tablet of spelcomputer meedoen via Bedrock. Door een PaperMC-server uit te rusten met de plug-in GeyserMC (en optioneel Floodgate) kunnen beide platformen tegelijk in dezelfde spelwereld spelen.

```
services:
  mc:
    image: itzg/minecraft-server:latest
    container_name: mc-crossplay
    restart: unless-stopped
    ports:
      - "25565:25565"        # Java Edition netwerkpoort (TCP)
      - "19132:19132/udp"    # Bedrock Edition netwerkpoort (UDP via Geyser)
    environment:
      EULA: "TRUE"
      MEMORY: "4G"
      TYPE: "PAPER"
      PLUGINS: |

https://download.geysermc.org/v2/projects/geyser/versions/latest/builds/latest/downlo
ads/spigot

https://download.geysermc.org/v2/projects/floodgate/versions/latest/builds/latest/dow
nloads/spigot
    volumes:
      - ./crossplay-data:/data
```

Wat doen deze plug-ins precies?

• GeyserMC: Vertaalt 'on-the-fly' de netwerkpakketten en spelmachanieken tussen het Bedrock-protocol en het Java-protocol.

• Floodgate: Zorgt ervoor dat Bedrock-spelers kunnen inloggen met hun Microsoft/Xbox-account zonder dat ze verplicht een Java Edition licentie hoeven te bezitten.

5. Controle-opdrachten voor Studenten

Opdracht 1: Controleer met het commando `docker ps` of de container actief is en verifieer welke TCP- en UDP-poorten zijn gepubliceerd.

Opdracht 2: Leg het fundamentele verschil uit tussen transportprotocollen TCP (gebruikt voor Java op 25565) en UDP (gebruikt voor Bedrock op 19132).

Opdracht 3: Maak jezelf beheerder (Operator) via `docker exec -i mc-server rcon-cli` en test of je bevoegdheden werken.

© 2026 Onderwijsadvies Leren met Actie