

1. Ziele

- Sie können Docker swarm korrekt konfigurieren und eine Webapplikation über mehrere Nodes skalieren

2. Vorbereitung

1. Erstellen Sie 3 EC2-Instanzen (Typ: Ubuntu Server 22.04 t2.micro) in der AWS-Cloud. Benennen Sie diese als Node 1, Node 2 und Node 3. Verwenden Sie für alle Nodes dasselbe Schlüsselpaar.

Instances (3)
[Info](#)

[Verbinden](#)

[Instance-Status ▾](#)

[Aktionen ▾](#)

[Ins...](#)

[Instance-Status = running](#)
✕

[Filter löschen](#)

<input type="checkbox"/>	Name ▾	Instance-ID	Instance-Status ▾	Instance-Typ ▾	Statusüberprüfung
<input type="checkbox"/>	Node 2	i-09f8ac74b17107907	Läuft	t2.micro	2/2-Prüfungen bestanden
<input type="checkbox"/>	Node 3	i-073d9134b6936108	Läuft	t2.micro	2/2-Prüfungen bestanden
<input type="checkbox"/>	Node 1	i-0d2ca3784616e7e69	Läuft	t2.micro	2/2-Prüfungen bestanden

2. Öffnen Sie zu jeder Instanz eine ssh-Sitzung. Zur Erinnerung:
 - Der Pfad zum privaten Schlüssel kann mit `ssh -i c:\path\to\privatekey.pem ubuntu@publicIP` angegeben werden
 - Beim privaten Schlüssel müssen alle vererbten Berechtigungen entfernt werden und dem aktuellen Benutzer Vollzugriff erteilt werden.
3. Damit die drei Instanzen untereinander kommunizieren können, muss eine gemeinsame Security-Group konfiguriert und zugewiesen werden, die der Einfachheit halber jeglichen eingehenden Datenverkehr zulässt.

Regeln für eingehenden Datenverkehr (1/1)

Tags verwalten

Regeln für eingehenden Datenverkehr bearbeiten

< 1 >

<input checked="" type="checkbox"/>	Name	ID der Sicherheitsg...	IP-Version	Typ	Protokoll
<input checked="" type="checkbox"/>	-	sgr-0fa7fc5ecd0695713	IPv4	Gesamter Datenverkehr	Alle

4. Überprüfen Sie, ob die Instanzen sich gegenseitig anpingen können. Notieren Sie sich die **privaten** IP-Adressen der drei Instanzen:

[illegible]

5. Installieren Sie auf allen Instanzen docker gemäss Anleitung auf <https://gbssg.gitlab.io/m347/docker-installation/>

[illegible]

3. Aufgaben

- a. Konfigurieren Sie einen docker swarm (1 Manager Node + 2 Worker Nodes) gemäss Anleitung auf <https://gbssg.gitlab.io/m347/orchestrierung-swarm/> . Verwenden Sie dazu die privaten IP-Adressen der EC2-Instanzen

[illegible]

- b. Erstellen Sie eine php-Webapplikation bestehend aus index.php, Dockerfile und docker-compose.yml. Die Webapplikation soll den Hostnamen (= ID des Containers) ausgeben. Deployen Sie diese und skalieren Sie diese zuerst auf 3, dann auf 10 Container. Wie werden die Container auf die 3 Nodes verteilt?

[illegible]

- c. Rufen Sie die Webseite über die öffentliche IP-Adresse des Managar Nodes mehrfach auf. Was stellen Sie fest?

[illegible]

- d. Simulieren Sie einen Ausfall eines Nodes, indem Sie einen der Worker-Node EC2-Instanzen stoppen. Was stellen Sie fest?

[illegible]

- e. Schalten Sie die Instanz wieder ein und führen Sie auf dem Manager Node das Kommando `docker service update --force helloswarm_web` aus. Was stellen Sie fest?

[illegible]

Zeit: 60 Minuten

4. Erwartete Resultate

Screenshots und Kommandos und beantwortete Fragen der erfolgreich gelösten Teilaufgaben