

Docker Swarm on AWS

[<< Back to Docker main](#)



<https://docs.docker.com/machine/examples/aws/#step-1-sign-up-for-aws-and-configure-credentials>
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/get-set-up-for-amazon-ec2.html>

Contents

- 1 User létrehozása
- 2 VM létrehozása
- 3 Mi jött létre
- 4 Apache telepítése
- 5 VM törlése

User létrehozása

Ahhoz hogy a docker-machine be tudjon lépni az AWS account-unkba, létre kell hozni egy IAM (Identity and Access Management) adminisztrátor jogú user-t. <https://console.aws.amazon.com/iam>

	User	Access key ID	Secret access key
▶	✓ dockeruser	AKIAJ622JUHS4RIJOI3A	

A user létrehozása közben kapni fogunk egy **access-id**-t és egy egy titkos kulcsot. Ezt a kettőt kell majd megadni a docker-machine-nak. Létre kell hozni egy **credentials** fájlt a .aws mappában. Alapértelmezetten itt keresni majd az aws driver. Másoljuk bele az id-t és a kulcsot: `~/aws/credentials`

```
[default]
aws_access_key_id = AKIAJ622JUHS4RIJOI3A
aws_secret_access_key = XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

VM létrehozása

Az amazon driver alapértelmezetten benne van a docker-machine-ben, nem kell külön telepíteni mint a KVM esetén láttuk. Ha ingyenes VM-et akarunk megadni, akkor használjuk a t2.micro típust.

Meg kell adni, hogy melyik klászterben van az amazon el?fizetésünk. Ez az EC2 Dashboard-on is látszik, illetve az URL-b?l is (<https://ap-southeast-1.console.aws.amazon.com>). A minnek a szingapúri 1-es ben van.

Ezen felül fontos hogy megadjuk, hogy az új VM-en melyik portok legyenek nyitva. Alapértelmezetten csak a 22 SSH és a 2375 Docker port van nyitva. Az amazec2 driver alapértelmezetten ubuntu image-t telepít az új VM-ekre.

```
# docker-machine create --driver amazec2 \
--amazec2-instance-type "t2.micro" \
--amazec2-region ap-southeast-1 \
--amazec2-open-port 8080 docker-sendbox
```

```
Running pre-create checks...
Creating machine...
(docker-sendbox) Launching instance...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with ubuntu(systemd)...
Installing Docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: docker-machine env docker-sendbox
```

Mi jött létre

Az EC2 konzolon láthatjuk, hogy létrejött az új VM-ün docker-sendbox néven, amit megadtunk a docker-machine parancsban.

☐	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
☐	docker-send...	i-07ad023d54b70e2dc	t2.micro	ap-southeast-1a	running	2/2 checks ...	None	ec2-13-228-78-148.ap-s...	13.228.78.148

Láthatjuk, hogy 2476 docker port, a 22 SSH port és az általunk megadott 8080 port van nyitva. Ide fogjuk rákötni az apache-ot.

Security Groups associated with i-07ad023d54b70e2dc			
Ports	Protocol	Source	docker-machine
8080	tcp	0.0.0.0/0	✓
22	tcp	0.0.0.0/0	✓
2376	tcp	0.0.0.0/0	✓

Láthatjuk az Amazon által, a VM-hez rendelt publikus IP címet és domain nevet.

Public DNS (IPv4)	ec2-13-228-78-148.ap-southeast-1.compute.amazonaws.com
IPv4 Public IP	13.228.78.148

Ha listázzuk a saját hostunkon a docker gépeket, láthatjuk, hogy regisztrálta nálunk a docker-sendbox nev? VM-et, ami az el?z? képen is látható publikus IP címen (54.169.84.170) érhet? el amzaonec2 driverrel futtatva.

# docker-machine ls							
NAME	ACTIVE	DRIVER	STATE	URL	SWARM	DOCKER	ERRORS
docker-sendbox	-	amazonec2	Running	tcp://54.169.84.170:2376		v18.05.0-ce	

Bámulatos, de a lokális gépünkr?l a docker-machine ssh paranccsal be tudunk lépni minden további nélkül az új docker-es VM-re, anélkül hogy bármilyen IP címet vagy felhasználó nevet meg kéne adni. Ez azért lehetséges, mert a docker-machine elhelyezett az új VM-en egy publikus SSH kulcsot, a privát párját meg a mi hostunkon.
Láthatjuk, hogy tényleg **Ubuntu 16.04.2** verzió van az új VM-en.

```
# docker-machine ssh docker-sendbox
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-1020-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

184 packages can be updated.
92 updates are security updates.

ubuntu@docker-sendbox:~$
```

Apache telepítése

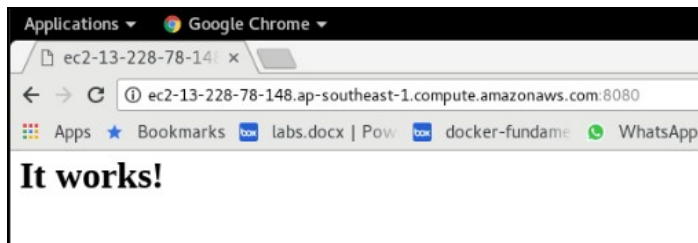
Az új VM-re telepítsünk fel egy apache-t, és a 80-as portját irányítsuk a VM 8080 portjára, ami publikusan is elérhet? port az amazonon, mivel így hoztuk létre a VM-et.

```
ubuntu@docker-sendbox1:~$ sudo docker run -dit --name apache -p 8080:80 httpd:2.4
Unable to find image 'httpd:2.4' locally
2.4: Pulling from library/httpd
1c7fe136a31e: Pull complete
72d2d593b3a7: Pull complete
96cd507e10fd: Pull complete
ec6bb3492a46: Pull complete
0ea3a980fa47: Pull complete
ebe42c0aa1a8: Pull complete
a640bae86fe0: Pull complete
Digest: sha256:963ecd717afb125c7a867d82d6935930bb93acc5078ea8bc37854d4b4de766d9
Status: Downloaded newer image for httpd:2.4
21b8f150ed88434b73a95e47a4a07c78859965c19325b08578b11aca3dcc4761
```

Nézzük meg, hogy tényleg fut e az apache konténer a VM-en:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
2d6843f4f464	httpd:2.4	"httpd-foreground"	41 seconds ago	Up 40 seconds	0.0.0.0:8080->80/tcp	apache

Most írjuk be a böngész?be az EC2-s konzolon mutatott domain nevet a 8080 porttal, ahol az apache-nak kell hallgatóznia. Ekkor be kell jöjjön a http image alapértelmezett oldala, ahogy az a képen is látható



VM törlése

Az EC2-es VM-et a lokális hosztunkrol ki lehet törölni az **rm** (--force) paranccsal. Ekkor a mi docker-ünkből és az amazon-rol is törli:

```
# docker-machine rm -f docker-sendbox
About to remove docker-sendbox1
WARNING: This action will delete both local reference and remote instance.
Successfully removed docker-sendbox1
```

Láthatjuk, hogy az **rm** parancs kiadása után shutting-down állapotba került a VM:

☐	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
	docker-send...	i-07ad023d54b70e2dc	t2.micro	ap-southeast-1a	shutting-do...		None	ec2-13-228-78-148.ap-s...	13.228.78.148

Majd terminated állapotba. (A terminated VM-ek kb 20 perc után eltűnnek az **Instances** listáról.)

☐	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP
	docker-send...	i-07ad023d54b70e2dc	t2.micro	ap-southeast-1a	terminated		None		-