



Szerveroldali fejlesztés



ÓBUDAI EGYETEM
NEUMANN JÁNOS INFORMATIKAI KAR

MODUL 10

Földi rendszerek

Konténerizáció

Docker konténerek felhasználása

Docker konténerek készítése

Környezeti változók kezelése

- A felhő számítástechnika flexibilis és rugalmas, a legtöbb esetben jól használható
- Mikor nem jó megoldás a felhő?
 - Hogyha rendelkezünk fizikai szerverrel
 - Hogyha még egy megbízható felhő providerre sem szeretnénk rábízni az adatainkat
 - Hogyha kiszámíthatóbbak szeretnénk lenni → rugalmasság feláldozása
- Honnan lehet szerverünk?
 - Otthoni hálózatban egy Raspberry PI/régebbi laptop/low power PC + fix IP / DDNS
 - Saját fizikai gép elhelyezése hosting szolgáltatónál (ekkor garantált a fix IP + nagy sávszélesség)
 - Hosting szolgáltatótól fizikai gép bérlete
 - Hosting szolgáltatótól virtuális gép bérlete
 - Felhő szolgáltatótól virtuális gép bérlete

- PI: 4 mag, 8gb ram, 100gb storage 1 évre
 - Otthoni hálózatban egy Raspberry PI/régebbi laptop/low power PC + fix IP / DDNS
 - Raspberry PI 4: 30k + SSD 20k + fogyasztás kb. 8k/év → **8.000 ft / év**
 - Átlag desktop pc 0-24 fogyasztása: **52.000 ft / év**
 - Saját fizikai gép elhelyezése hosting szolgáltatónál (ekkor garantált a fix IP + nagy sávszélesség)
 - Br. 33.000 ft / hó → **400.000 ft / év**
 - Hosting szolgáltatótól fizikai gép bérlete
 - Br. 25.000 ft / hó → **300.000 ft / év**
 - Hosting szolgáltatótól virtuális gép bérlete
 - Br. 7.500 ft / hó → **90.000 ft / év**
 - Felhő szolgáltatótól virtuális gép bérlete
 - Br. 30.000 ft / hó → **360.000 ft / év**
- Saját javaslat: magyar szolgáltatótól VPS bérlet + Azure Blob Storage 😊



Virtual machines

Instance details

Virtual machine name * ⓘ

Region * ⓘ

Availability options ⓘ

Security type ⓘ

Image * ⓘ

[See all images](#) | [Configure VM generation](#)

VM architecture ⓘ ☐ Arm64 ☒ x64

ⓘ Arm64 is not supported with the selected image.

Run with Azure Spot discount ⓘ ☐

Size * ⓘ

[See all sizes](#)

Administrator account

Authentication type ⓘ ☐ SSH public key ☒ Password

Username * ⓘ

Password * ⓘ

Confirm password * ⓘ

Oprendszer

Hardver

Belépési adatok

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ

☐ None

☒ Allow selected ports

Select inbound ports *

⚠ This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

SSH port legyen nyitva a távoli hozzáféréshez

The screenshot displays the Azure portal interface for a virtual machine named 'dockerTester'. The left sidebar shows the navigation menu with options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, and Connect. The main content area shows the 'Essentials' tab with various properties of the VM. A yellow box labeled 'vezérlés' (control) points to the 'Start', 'Restart', and 'Stop' buttons. Another yellow box labeled 'IP cím' (IP address) points to the 'Public IP address' field, which contains the value '20.107.42.239'.

vezérlés

IP cím

dockerTester Virtual machine

Search

Connect Start Restart Stop Capture Delete Refresh Open in mobile CLI / PS Feedback

Essentials

Resource group (move) : [oenik](#)

Status : Running

Location : West Europe

Subscription (move) : [Azure for Students](#)

Subscription ID : fb495e35-2521-42b7-901f-7a3d101bbd1

Tags (edit) : [Click here to add tags](#)

Operating system : Linux (debian 11)

Size : Standard B2s (2 vcpus, 4 GiB memory)

Public IP address : [20.107.42.239](#)

Virtual network/subnet : [oenik-vnet/default](#)

DNS name : [Not configured](#)

Properties Monitoring Capabilities (7) Recommendations Tutorials

dockerTester | Networking

Virtual machine

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Networking

Connect

Disks

Size

Microsoft Defender for Cloud

Advisor recommendations

Extensions + applications

Continuous delivery

Availability + scaling

Attach network interface

Detach network interface

dockertester829

IP configuration ⓘ

ipconfig1 (Primary)

Network Interface: dockertester829

Effective security rules

Troubleshoot VM connection issues

Topology

Virtual network/subnet: oenik-vnet/default

NIC Public IP: 20.107.42.239

NIC Private IP: 10.2.0.4

Accelerated networking: Disabled

Inbound port rules

Outbound port rules

Application security groups

Load balancing


Network security group dockertester-nsg (attached to network interface: dockertester829)

Impacts 0 subnets, 1 network interfaces

Priority	Name	Port	Protocol	Source	Destination	Action	
300	SSH	22	TCP	Any	Any	Allow	...
310	AllowAnyCustom80Inbound	80	TCP	Any	Any	Allow	...
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow	...
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow	...
65500	DenyAllInBound	Any	Any	Any	Any	Deny	...

Portok kinyitása

Add inbound port rule

 **Add inbound security rule** ×

dockerTester-nsg

Source ⓘ

Any

Source port ranges * ⓘ

*

Destination ⓘ

Any

Service ⓘ

Custom

Destination port ranges * ⓘ

8080

Protocol

☒ Any

☐ TCP

☐ UDP

☐ ICMP

Action

☒ Allow

☐ Deny

Priority * ⓘ

320

Name *

AllowAnyCustom8080Inbound

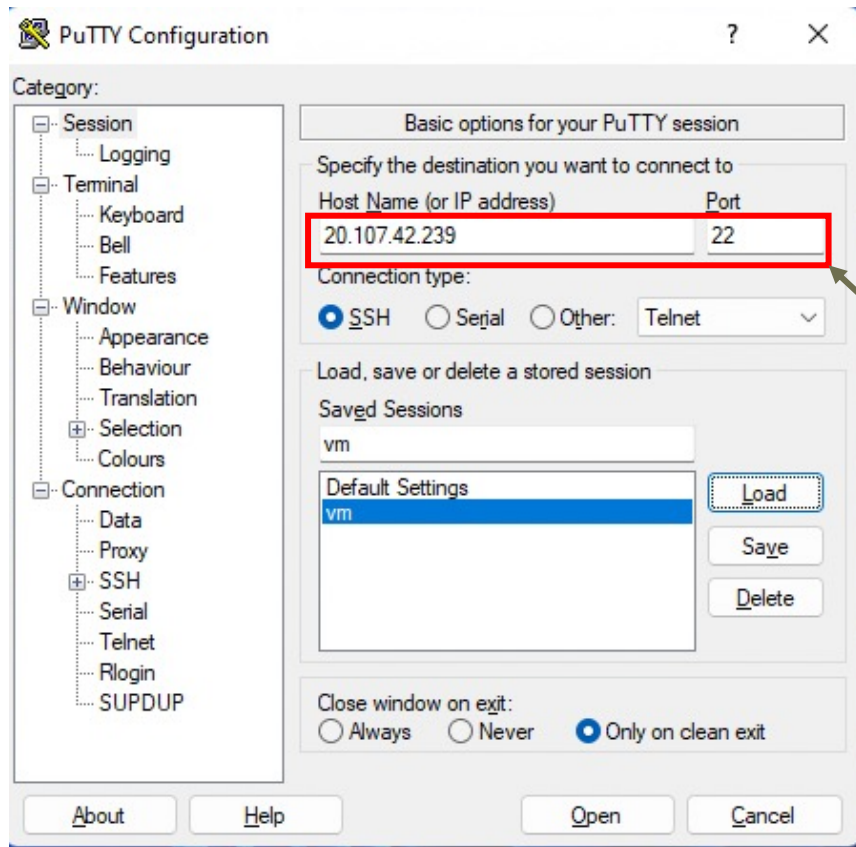
Description

Add

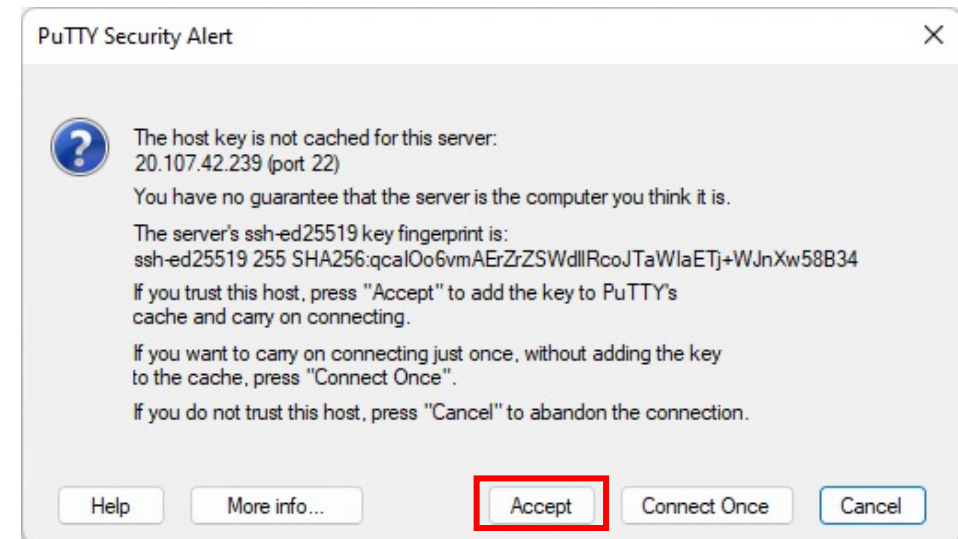
Cancel

Cél port a szerveren

- **Putty**: SSH kliens szinte minden platformra



IP cím



```
kovacsandras@dockerTester: ~  
login as: kovacsandras  
kovacsandras@20.107.42.239's password:  
Linux dockerTester 5.10.0-18-cloud-amd64 #1 SMP Debian 5.10.140-1 (2022-09-02) x  
86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Thu Oct 13 16:12:19 2022 from 46.107.208.88  
kovacsandras@dockerTester:~$
```

- Szerezzünk be egy Debian alapú virtuális szervert!



- free → mennyi memóriánk van

	total	used	free	shared	buff/cache	available
Mem:	4027704	114408	3809636	404	103660	3732528
Swap:	0	0	0			

- df -h → mennyi szabad helyünk van a háttértáron

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	2.0G	0	2.0G	0%	/dev
tmpfs	394M	404K	393M	1%	/run
/dev/sda1	30G	824M	27G	3%	/
tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
/dev/sda15	124M	5.9M	118M	5%	/boot/efi
/dev/sdb1	7.8G	24K	7.4G	1%	/mnt
tmpfs	394M	0	394M	0%	/run/user/1000

- top → Processzor és RAM terhelés

```
top - 16:19:21 up 12 min, 1 user, load average: 0.00, 0.00, 0.00
Tasks: total, running, sleeping, stopped, zombie
%Cpu(s): us, sy, ni, id, wa, hi, si, st
MiB Mem : total, free, used, buff/cache
MiB Swap: total, free, used, avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
641	root	20	0	175336	25576	10076	S	0.3	0.6	0:02.22	python3
1	root	20	0	99548	10408	7792	S	0.0	0.3	0:01.41	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
5	root	20	0	0	0	0	I	0.0	0.0	0:00.04	kworker/0:0-events
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-events_highpri
7	root	20	0	0	0	0	I	0.0	0.0	0:00.07	kworker/u4:0-flush-8:0
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.01	kworker/0:1H-events_highpri
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq

- Ctrl + c –vel tudunk kilépni belőle
- sudo su → rendszergazda szintre lépés

- Régen egy webalkalmazás hostolása
 - Linux vagy Windows szerver operációs rendszer telepítése
 - RAID kialakítása és rendszeres biztonsági mentés
 - Adatbázis motor telepítése (pl. MS SQL)
 - Webszerver szoftver telepítése (pl. IIS)
 - Alkalmazás SDK telepítése (pl. NET 5 SDK)
 - Alkalmazás fordítása
- Ma egy webalkalmazás hostolása
 - Bezárjuk a függőségeivel együtt egy önálló konténerbe
 - Teljesen mindegy az operációs rendszer
 - Telepítjük a Dockert
 - Letöltjük a konténerünket
 - Futtatjuk + beállítjuk, hogy automatikusan induljon minden indításkor

- Egy konténer részei
 - Futtatható állomány vagy állományok
 - Statikus fájlok
 - Átirányított mappák a host operációs rendszerről
 - Indítási parancs → mi történjen ha elindítják
- A konténerek osztoznak az operációs rendszer kernelén, de egy olyan izolált környezetben vannak, hogy azt hiszik magukról, hogy ők az egyetlen futó folyamat
- Nem férnek hozzá egymáshoz
- Előnyök
 - Programok és szolgáltatások egységbezárása a függőségekkel → 1 db konténerfájl
 - A konténerek átmozgathatóak könnyedén (copy + paste)
 - Szolgáltatás telepítése = egy fájl letöltése és elindítása
 - Szolgáltatás törlése = egy fájl kitörlése

- Telepítés lépései SSH-n keresztül
 - apt-get update
 - apt-get install ca-certificates curl gnupg lsb-release
 - mkdir -p /etc/apt/keyrings
 - curl -fsSL https://download.docker.com/linux/debian/gpg | gpg --dearmor -o /etc/apt/keyrings/docker.gpg
 - echo "deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/debian \$(lsb_release -cs) stable" | tee /etc/apt/sources.list.d/docker.list > /dev/null
 - apt-get update
 - apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
- Ellenőrzés
 - docker info

- Konténer letöltése és futtatása

```
docker run kovi91/vzwelcomeconsole
```

- Futtatás és interaktálás (pl. Konzol műveletek)

```
docker run -it kovi91/vzwelcomeconsole
```

- Végtelenül futó konténer indítása, port átirányítása (ex:in), elnevezése, háttérbe küldése és auto indítása a rendszer indulásakor

```
docker run -p 80:80 --name vzmvc -d --restart=always kovi91/vzwelcomemvc
```

- Végtelenül futó konténer megállítása

```
docker stop vzmvc
```

- Konténer törlése

```
docker rm vzmvc
```

- Konténerek listázása

```
docker container ls
```

```
docker container ls -a
```

- Konténer logjainak olvasása

```
docker logs vzmvc
```

- Konténer erőforrás használat

```
docker stats
```

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
85ff0b49668f	vzmvc	0.00%	22.11MiB / 3.841GiB	0.56%	1.02kB / 0B	0B / 4.1kB	14

- Honnan jönnek a konténerek?
 - hub.docker.com → rengeteg publikus előregyártott konténer
 - mcr.microsoft.com → szintén
- Konténerek testreszabása
 - Környezeti változókon keresztül (-e kapcsoló)
- Példa: MSSQL kiszolgáló telepítése

```
docker run -e "ACCEPT_EULA=Y" -e "MSSQL_SA_PASSWORD=Almafa123!!!" -p 1433:1433 --name sql --hostname sql -d mcr.microsoft.com/mssql/server:2022-latest
```

- Telepítsünk MSSQL szerveret!
- Nyissuk meg a VM-en az 1433-as portot!
- A fényképnézegető projekt adatbázisát migráljuk át ide!



- VS → Project → jobb klikk → Add → Docker Support → Linux
- Elkészül egy Dockerfile

```
FROM mcr.microsoft.com/dotnet/aspnet:6.0 AS base  
WORKDIR /app  
EXPOSE 80
```

Konténer template, workdir és port nyitás

```
FROM mcr.microsoft.com/dotnet/sdk:6.0 AS build  
WORKDIR /src  
COPY ["VZWelcomeMVC/VZWelcomeMVC.csproj", "VZWelcomeMVC/"]  
RUN dotnet restore "VZWelcomeMVC/VZWelcomeMVC.csproj"  
COPY . .  
WORKDIR "/src/VZWelcomeMVC"  
RUN dotnet build "VZWelcomeMVC.csproj" -c Release -o /app/build
```

Build folyamat

```
FROM build AS publish  
RUN dotnet publish "VZWelcomeMVC.csproj" -c Release -o /app/publish
```

Release folyamat

```
FROM base AS final  
WORKDIR /app  
COPY --from=publish /app/publish .  
ENTRYPOINT ["dotnet", "VZWelcomeMVC.dll"]
```

Belépési pont definiálása

Publish

Where are you publishing today?

Target



Azure

Publish your application to the Microsoft cloud



Docker Container Registry

Publish your application to any supported Container Registry that works with Docker images



Folder

Publish your application to a local folder or file share



FTP/FTPS Server

Publish your application to an FTP/FTPS server



Web Server (IIS)

Publish your application to IIS using Web Deploy or Web Deploy Package

Back

Next

Finish

Cancel

Publish

Which Container Registry you would like to use to host your application?

Target

Specific target



Azure Container Registry

Publish your application as a Docker image to Azure Container Registry



Docker Hub

Publish your application as a Docker image to Docker Hub



Other Docker Container Registry

Publish your application as a Docker image to any supported Container Registry

Back

Next

Finish

Cancel

x

Publish

Enter publish credentials

Target

Specific target

Registry

☒ Publish to a personal repository

User name

xyz

Password

●●●●●●●●●●

(Password is optional. Leave empty if the password is available in the docker config.)

Back

Next

Finish

Cancel

- Készítsünk a fényképnézegető alkalmazásunkból konténert!
- Telepítsük a szerverre a konténert!



- Érzékeny adatok (jelszavak, api kulcsok, connection stringek)
 - Eddig: beleégettük a kódba
 - Innentől
 - a kód környezeti változóból vegye át
 - A konténer indításakor adjuk át a változót és tartalmát az –e kapcsoló után
- C# környezeti változó olvasás

```
string getEnv = Environment.GetEnvironmentVariable("envVar");
```

- Igényeljük tanúsítványt az alkalmazásunknak!



- Regisztráljunk domain nevet az alkalmazásunknak!



- Készítsünk több alkalmazást – több aldomainre!



Köszönöm a figyelmet!

Kérdés esetén e-mailben szívesen állok rendelkezésre.