

Rendszerüzemeltetés 3 BSc

1. Gyakorlati feladat

2023. 10. 18.

Készítette: Nagy Bence
Neptunkód: WH8L7E

Miskolc, 2023

1.Feladat: HA és LB megoldások demonstrálása Cisco Packet Tracer segítségével

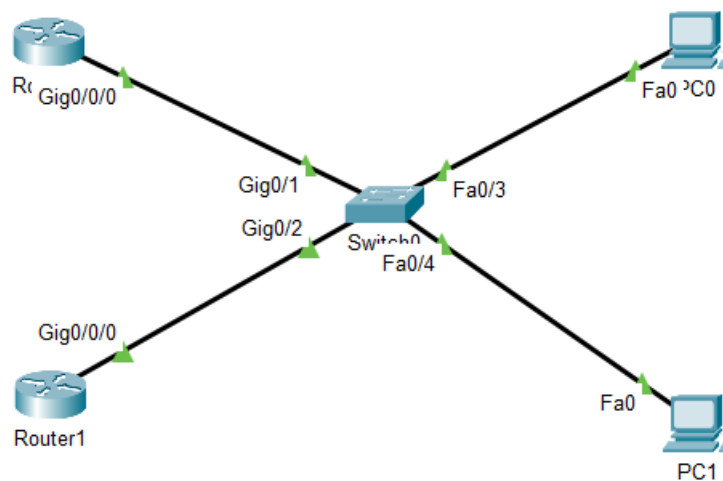
A Cisco Packet Tracer egy szimulációs eszköz, amely lehetővé teszi hálózatok tervezését, konfigurálását és szimulálását. A high availability (nagy rendelkezésre állás) és a load balancing (terheléselosztás) bemutatásához egy olyan hálózatot kell terveznie és konfigurálnia, amely ezeket a funkciókat támogatja.

A HSRP (Hot Standby Router Protocol) egyszerű szemléltetéséhez 2 routerre, 1 switchre és 2 számítógépre van szüksége. A switch lehetővé teszi a két router közötti kommunikációt, és a számítógépek segítségével ellenőrizhető a hálózati rendelkezésre állás.

1.lépés: A hálózati topológia létrehozása:

Az eszközöket az alábbi módon kapcsoltam össze:

- Router0 FastEthernet0/1 --> Switch FastEthernet0/1
- Router1 FastEthernet0/1 --> Switch FastEthernet0/2
- PC0 --> Switch FastEthernet0/3
- PC1 --> Switch FastEthernet0/4



2.lépés: Router és pc ip címének konfigurációja:

A router0 a 193.168.0.1/24-es, a router1-nek pedig a 193.168.0.2/24-es címet osztottam ki.

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int g0/0/0
      ^
% Invalid input detected at '^' marker.

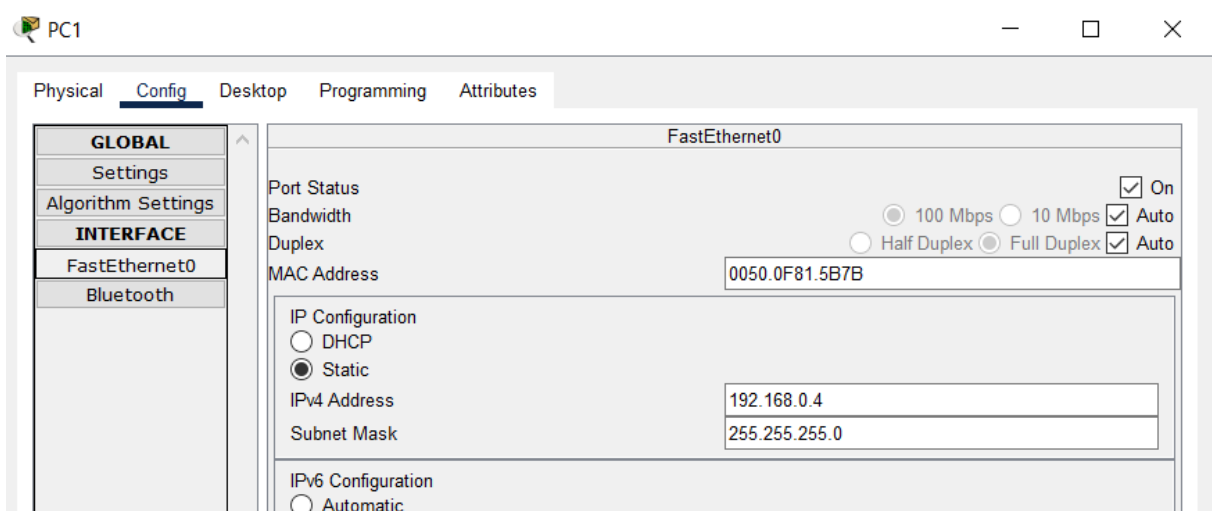
Router(config)#int g0/0/0
Router(config-if)#ip address 193.168.0.2 255.255.255.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/0, changed state to up

Router(config-if)#|
```

A PC0 192.168.0.3/24, a PC1 192.168.0.4/24-es ip címet kapott



3.lépés: Tesztelés:

```

Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.0.1

Pinging 192.168.0.1 with 32 bytes of data:

Reply from 192.168.0.1: bytes=32 time<lms TTL=255
Reply from 192.168.0.1: bytes=32 time<lms TTL=255
Reply from 192.168.0.1: bytes=32 time<lms TTL=255
Reply from 192.168.0.1: bytes=32 time<lms TTL=255

Ping statistics for 192.168.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.0.2

Pinging 192.168.0.2 with 32 bytes of data:

Reply from 192.168.0.2: bytes=32 time<lms TTL=255
Reply from 192.168.0.2: bytes=32 time<lms TTL=255
Reply from 192.168.0.2: bytes=32 time<lms TTL=255
Reply from 192.168.0.2: bytes=32 time<lms TTL=255

Ping statistics for 192.168.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

Miután a számítógépeket beállítottam, megpróbáltam pingelni a két router IP-címét (192.168.0.1 és 192.168.0.2), és ellenőrizni, hogy válasz érkezik-e. Ez segített ellenőrizni, hogy a routerek megfelelően vannak-e konfigurálva.

4.lépés: Load balancing tesztelése:

A Load Balancing teszteléséhez küldjünk kéréseket (pl: HTTP kéréseket) a Load Balancer címére vagy IP-címére. A Load Balancernek képesnek kell lennie arra, hogy a kéréseket a hátul lévő szerverek között eloszlassa. A válaszokat mindkét routerre kell elküldenie, és látni fogja, hogy az egyes router válaszol a kérésekre.

5.lépés: High Availability tesztelése:

A High Availability teszteléséhez leállíthatjuk az egyik router-t vagy az ahhoz vezető összeköttetést, és ellenőrizheti, hogy a másik router átveszi-e a forgalmat. Próbálja meg újra pingelni az alapértelmezett átjárót, és nézze meg, hogy a válasz megérkezik-e. Ezzel ellenőrizheti a nagy rendelkezésre állás működését.

6.lépés HSRP (Hot Standby Router Protocol) konfiguráció:

Beállítjuk a HSRP-t a Router1-en, a következőképp:

```

Router(config-if)#standby 1 ip 192.168.0.1
Router(config-if)#do sh standby br
                P indicates configured to preempt.
                |
Interface    Grp  Pri  P State      Active      Standby      Virtual IP
Gig0/0/0     1    100   Speak    unknown    unknown     193.168.0.1
Router(config-if)#
%HSRP-6-STATECHANGE: GigabitEthernet0/0/0 Grp 1 state Speak -> Standby

%HSRP-6-STATECHANGE: GigabitEthernet0/0/0 Grp 1 state Standby -> Active
do sh standby br
                P indicates configured to preempt.
                |
Interface    Grp  Pri  P State      Active      Standby      Virtual IP
Gig0/0/0     1    100   Active   local      unknown     192.168.0.1

```

Először speaking status-ban van, itt megnézi a hálózat kiépítését és rájön, hogy aktív tud lenni, ez mellett van egy virtuális IP is. Majd aktív módba áll.

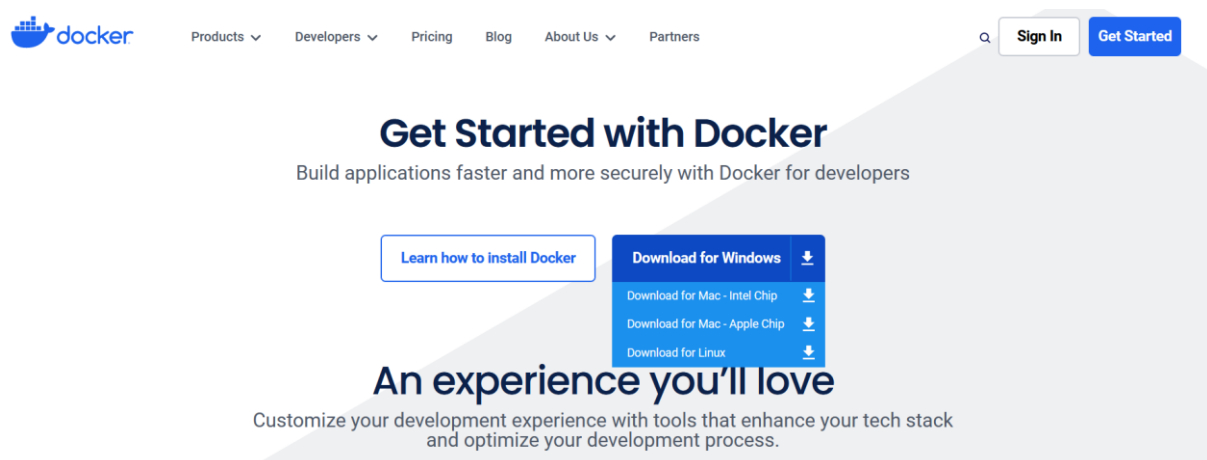
Ugyanezt megcsináljuk a Router0-val is! Ebben az esetben a Router0 standby állapotba került.

Így a két router között tudunk váltani, hogy melyik legyen az aktív és a standby. A preempt bekapcsolásával és a prioritás állításával megoldható, hogy az egyik router beavatkozzon (interrupt) és ő legyen az aktív router. Ebben az esetben másik router standby állapotba fog kerülni.

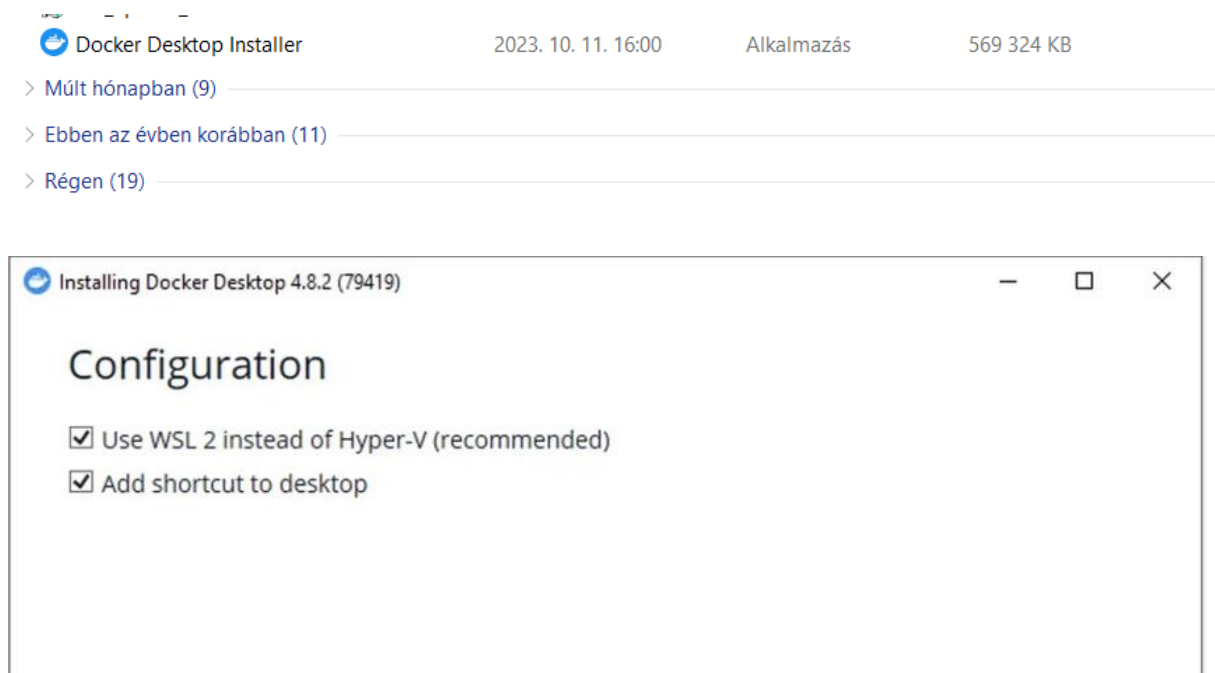
2.Feladat:

Docker letöltése:

Felmegyünk a <https://www.docker.com/get-started/> oldalra és letöltjük az szoftvert a megfelelő operációs rendszeren:



A letöltés után telepítjük a programot:



Ezután pedig letöltjük mellé a WSL2 Linux kernell-t a <https://learn.microsoft.com/hu-hu/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package> oldalról:

Step 4 - Download the Linux kernel update package

The Linux kernel update package installs the most recent version of the [WSL 2 Linux kernel](#) for running WSL inside the Windows operating system image. (To run [WSL from the Microsoft Store](#), with more frequently pushed updates, use `wsl.exe --install` or `wsl.exe --update`.).

1. Download the latest package:

- [WSL2 Linux kernel update package for x64 machines](#)

① Megjegyzés

If you're using an ARM64 machine, please download the [ARM64 package](#) instead. If you're not sure what kind of machine you have, open Command Prompt or PowerShell and enter: `systeminfo | find "System Type"`. **Caveat:** On non-English Windows versions, you might have to modify the search text, translating the "System Type" string. You may also need to escape the quotations for the find command. For example, in German `systeminfo | find '"Systemtyp"`.

2. Run the update package downloaded in the previous step. (Double-click to run - you will be prompted for elevated permissions, select 'yes' to approve this installation.)

MySql :

A docker letöltése után bejelentkezünk rendszergazdai jogosultsággal parancssoron keresztül a docker-be.

```
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. Minden jog fenntartva.

C:\Windows\system32>docker login
Authenticating with existing credentials...
Login Succeeded
```

Majd a következő parancs segítségével letöltjük és futtatjuk a mysql image file-t

```
C:\Windows\system32>docker run --name some-mysql -e MYSQL_ROOT_PASSWORD=my-secret-pw -d mysql
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
8e0176adc18c: Pull complete
2d2c52718f65: Pull complete
d88d03ce139b: Pull complete
4a7d7f11aa1e: Pull complete
ce5949193e4c: Pull complete
f7f024dfb329: Pull complete
5fc3c840facc: Pull complete
509068e49488: Pull complete
cbc847bab598: Pull complete
942bef62a146: Pull complete
Digest: sha256:1773f3c7aa9522f0014d0ad2bbdaf597ea3b1643c64c8ccc2123c64afd8b82b1
Status: Downloaded newer image for mysql:latest
9d74e60295e077dd5d71f18af10d957dd577ea948720c9e326b075c35261d694
```

Apache webserver:

Az Apache image letöltéséhez és futtatásához a következő parancsot használjuk:

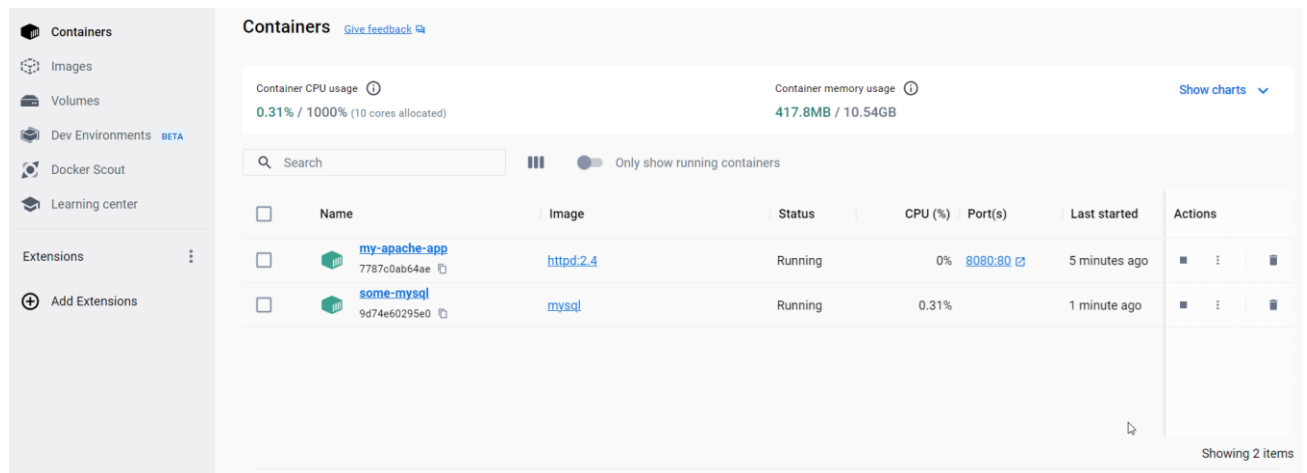
```
C:\Windows\system32>docker run -dit --name my-apache-app -p 8080:80 httpd:2.4
Unable to find image 'httpd:2.4' locally
2.4: Pulling from library/httpd
578acb154839: Pull complete
c1a8c8567b78: Pull complete
10b9ab03bf45: Pull complete
74dbedf7ddc0: Pull complete
6a3b76b70f73: Pull complete
Digest: sha256:4e24356b4b0aa7a961e7dfb9e1e5025ca3874c532fa5d999f13f8fc33c09d1b7
Status: Downloaded newer image for httpd:2.4
7787c0ab64ae8fd6e6e8ce5dedcaa98b27eab04c904b2cd074bcdedc9b8c66fd
```

Ezután, ha mindezzel készen vagyunk leellenőrizhetjük a docker ps parancs segítségével a meglévő image file-okat:

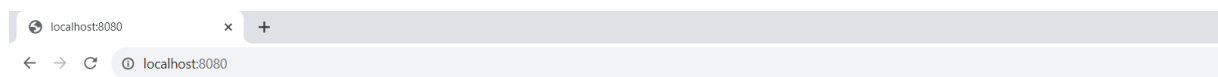
```
C:\Windows\system32>docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
9d74e60295e0	mysql	"docker-entrypoint.s..."	41 seconds ago	Up 40 seconds	3306/tcp, 33060/tcp	some-mysql
7787c0ab64ae	httpd:2.4	"httpd-foreground"	4 minutes ago	Up 4 minutes	0.0.0.0:8080->80/tcp	my-apache-ap

Ez a letöltött docker desktop-ban a következőképpen fog kinézni:



Fennt látható példaként az apache server localhost címe. Ezt beütve a böngészőbe az 'It works!' felirat fog minket fogadni, feltéve ha nem gépeltük el a címet:



It works!