

**VIZSGAREMEK**

Készítették:

Ötvös Bence – Fazekas Dominik – Somogyi Bánk

Konzulensek:

Csontos Dénes, Kormos-Tóth László Erik, Fülöp Tibor

Miskolc 2026.

Miskolci SZC Kandó Kálmán Informatikai Technikum

Miskolci Szakképzési Centrum

**Informatikai rendszer- és alkalmazás-üzemeltető technikus**

# Dokumentáció

Ötvös Bence – Fazekas Dominik – Somogyi Bánk

2025-2026

**Vizsgaremek Hálózati Dokumentáció**

# Tartalomjegyzék

[1. Tartalomjegyzék 3](#_Toc221094917)

[2. Bevezetés 4](#_Toc221094918)

[3. Topológia 4](#_Toc221094919)

[4. Megvalósításhoz használt eszközök 5](#_Toc221094920)

[5. IP Táblázat 5](#_Toc221094921)

[6. Telephelyek 6](#_Toc221094922)

[**a) Ruby Telephely (komplex)** 6](#_Toc221094923)

[1) SSH 7](#_Toc221094924)

[2) Vlanok 9](#_Toc221094925)

[3) Etherchannel 12](#_Toc221094926)

[4) Port-Security 14](#_Toc221094927)

[5) GLBP (Gateway Load Balancing Protocol) 15](#_Toc221094928)

[6) Dinamikus Forgalomirányítás (OSPF) 17](#_Toc221094929)

[7) PPP 18](#_Toc221094930)

[8) Tunnel 20](#_Toc221094931)

[9) NAT (Network Address Translation) 22](#_Toc221094932)

[10) ASA Tűzfal 23](#_Toc221094933)

[**b) Weiss Telephely** 25](#_Toc221094936)

[11) Wifi DHCP 26](#_Toc221094937)

[**c) Blake Telephely** 28](#_Toc221094938)

[12) Windows Server 30](#_Toc221094939)

[13) Linux Server 35](#_Toc221094940)

[7. Dokumentáció linkek: 42](#_Toc221094941)

[8. Forrásjegyzék: 42](#_Toc221094942)

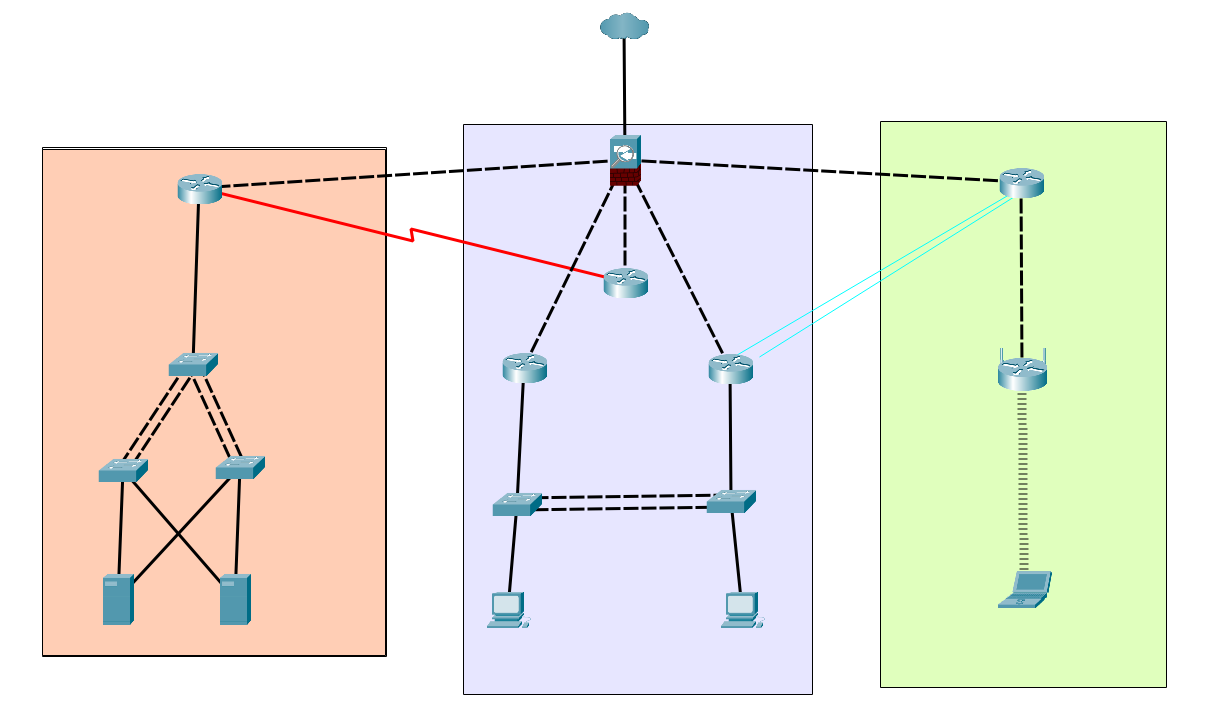
# Bevezetés

A projekt célja egy **biztonságos, skálázható és nagy rendelkezésre állású hálózati infrastruktúra** kialakítása, amely képes kiszolgálni különböző telephelyek (Blake, Ruby és Weiss) eltérő igényeit, miközben egységes címzési, routing- és biztonsági technológiákat alkalmaz.

Ez a dokumentáció bemutatja:

* a hálózat logikai és fizikai felépítését
* az alkalmazott technológiákat és beállításokats
* az IP-címzési tervet
* a telephelyek közötti kapcsolatok megvalósítását

# Topológia



**Weiss**

ASA

**Ruby**

**Blake**

PPP

Tunnel

Etherchannel

Windows

Linux

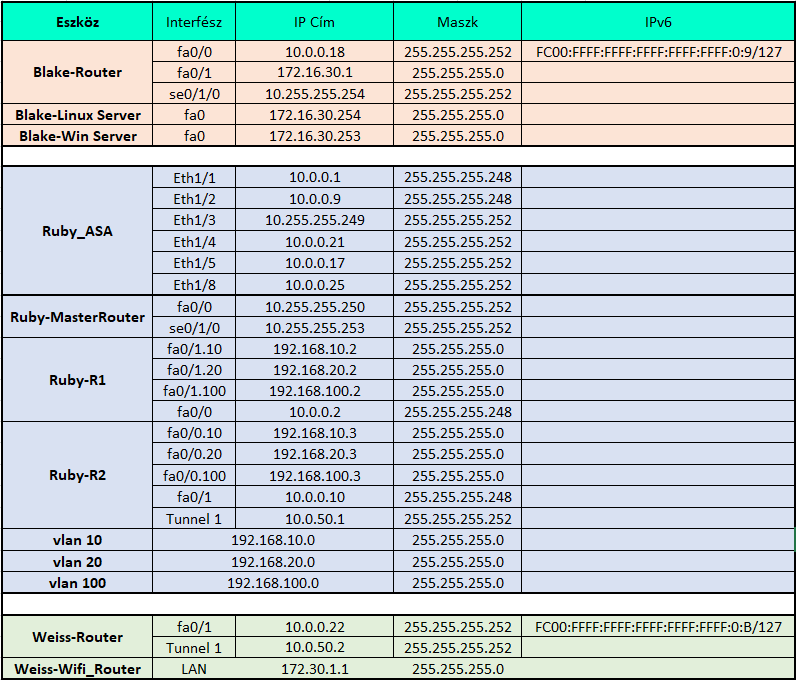
DHCP

Vlan 20

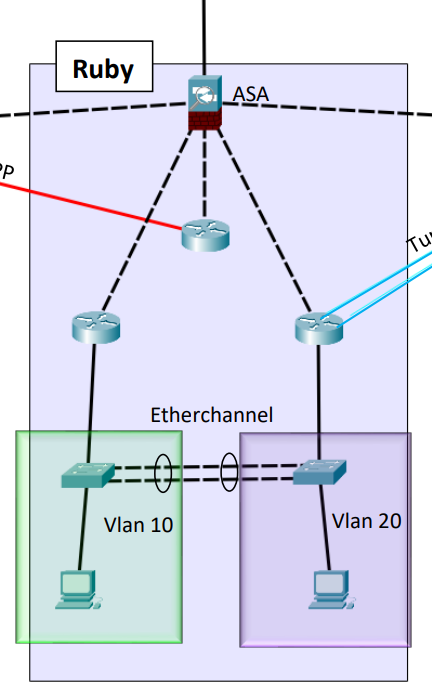
Vlan 10

# Megvalósításhoz használt eszközök

# IP Táblázat



# Telephelyek



## Ruby Telephely (komplex)

Használt technológiák:

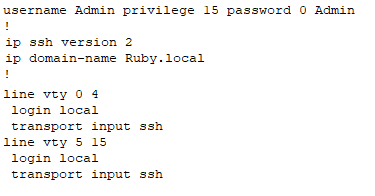
* SSH
* Vlan
* Etherchannel
* Port-security
* GLBP
* OSPF
* PPP
* Tunnel
* NAT

### SSH

Az SSH biztonságos távoli elérést biztosít számítógépek között, lehetővé teszi hogy távolról parancsokat futtassunk vagy fájlokat küldjünk a másik gépnek.

’Admin’ felhasználót és 15-ös privilégiumi szinttű ’Admin’ jelszót állítottunk be. Az ip domain-name ’Ruby.local’ lett, a ’0 4’ és a ’5 15’ vonalon alkalmaztuk az ssh-t. a 15-ös privilégium azonnal „enable” módba helyezi a felhaszálót bejelentkezéskor.

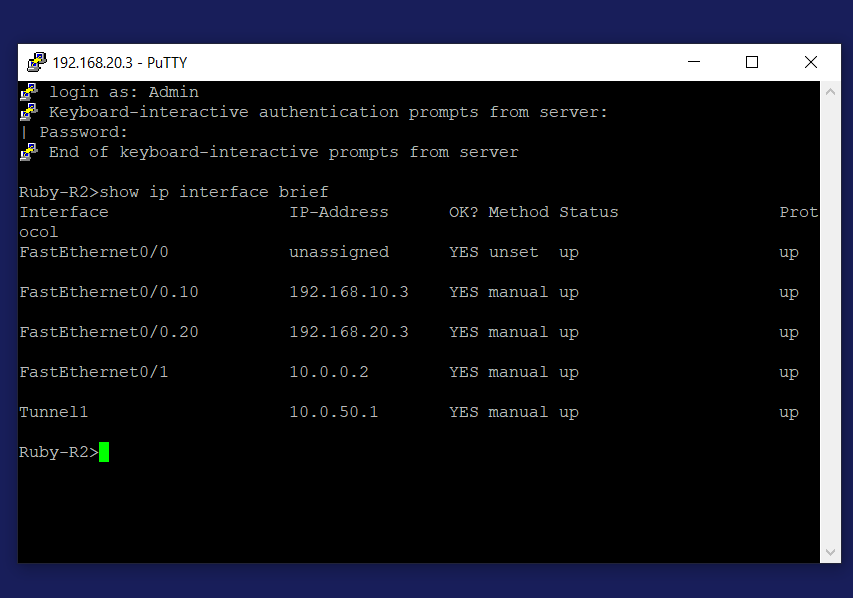
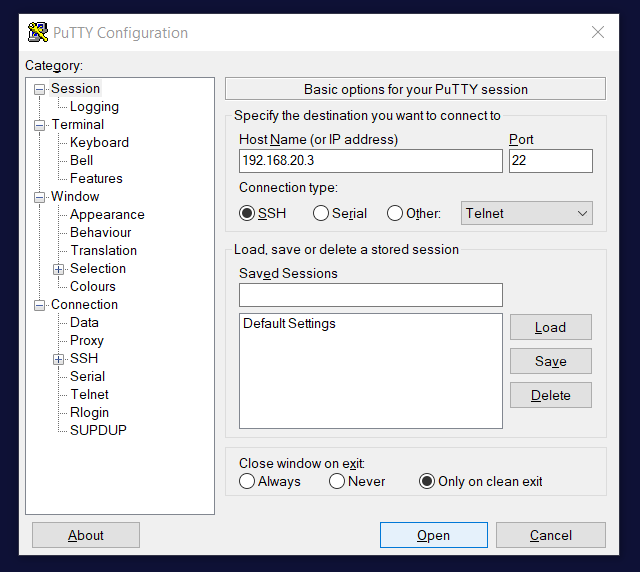
Parancsok SSH konfigurálásához:



Előnyei:

* Biztonságos
* Hitelesítés több módon
* Platformfüggetlen
* Sokoldalú
* Széles körben elterjedt

Router elérése SSH-val Putty-n keresztül:



### Vlanok

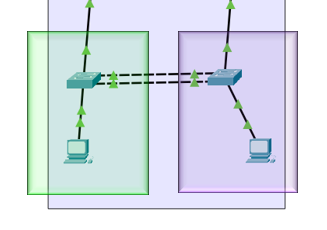
Vlanok segítségével egy fizikai hálózatot logikailag több, egymástól elkülönített hálózatra oszthatunk. Ezzel csökkentjük a szórási tartományt.

Előnyei:

* Biztonság
* Broadcast forgalom csökkentése
* Rugalmas hálózattervezés
* Költséghatékony
* Jobb menedzselhetőség

Létrehoztuk a Ruby telephelyen lévő switcheken a Vlan10-et és Vlan20-at a hálózatok elkülönítése érdekében.

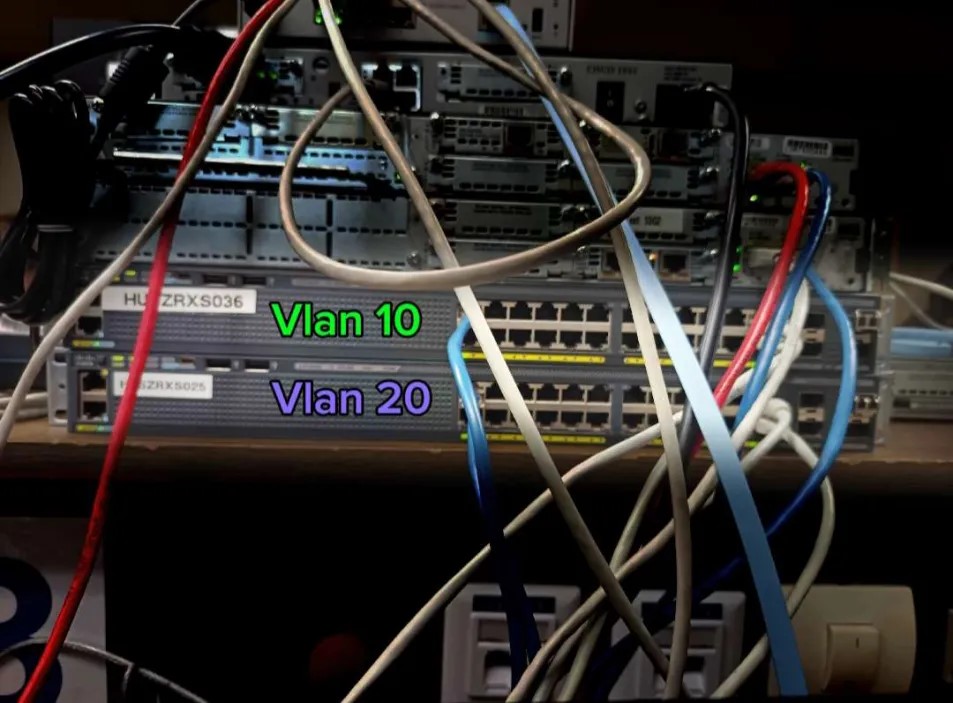
Vlanok a Ruby telephelyen





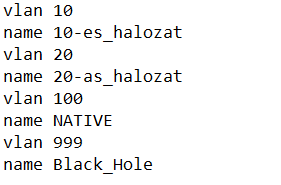
Vlan 20

Vlan 10



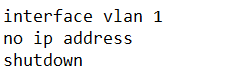
Ezzel növeljük a hálózat biztonságát és javítjuk a teljesítményét mivel elkülönítjük a forgalmat.

Vlanok létrehozása és elnevezése



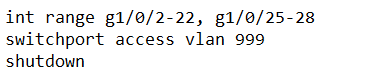
Vlan 1-et megszüntettük, mivel nem volt rá szükség. a Vlan-1 egy biztonsági réssé válhat.

minden switchen alapértelmezett.

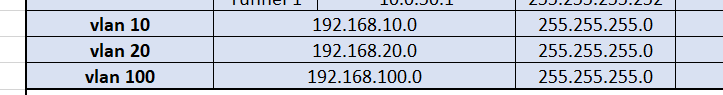


Native Vlan-nak a Vlan 100-at állítottuk be.

Létrehoztuk a vlan 999-et, a ”Black Hole” vlant, amibe bekerült összes nem használt lekapcsolt port, tehát az összes port a g1/0/1, g1/0/23 és a g1/0/24-en kívül.



Vlan 10 kapta a 192.168.10.0-ás hálózatot, vlan 20 a 192.168.20.0-ásat, vlan 100 pedig a 192.168.100.0-ásat.



Ruby-SW1



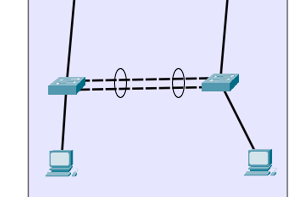
Ruby-SW2



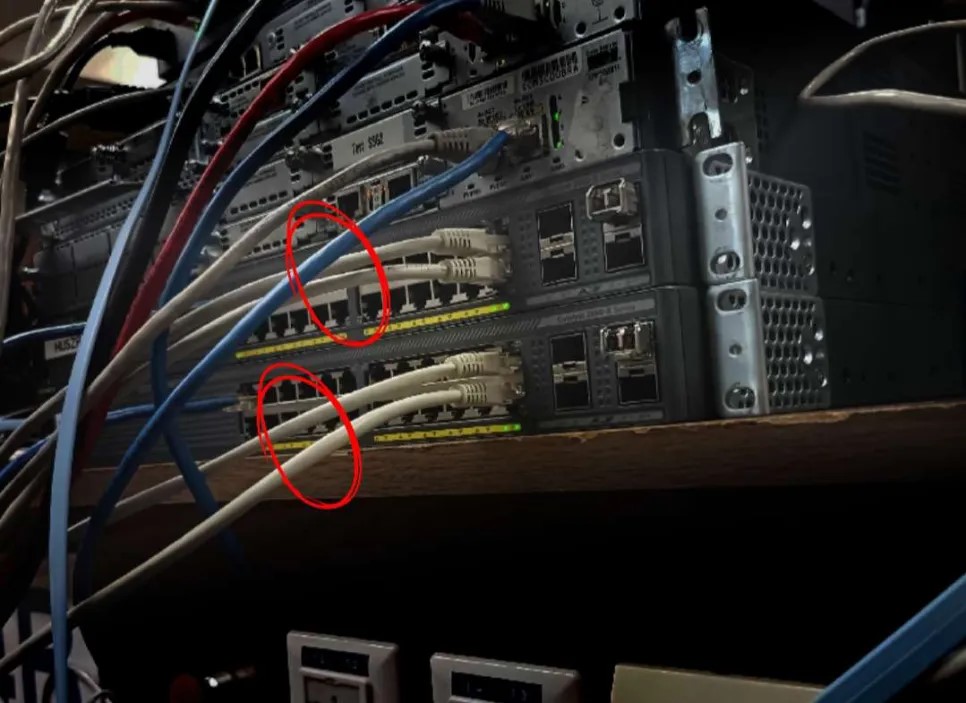
### Etherchannel

Az Etherchannel lehetővé teszi hogy egy switch két portját összefogjuk, így kettő helyett egyetlen összeköttetés jön létre. Az így létrejövő logikai kapcsolat nagyobb sávszélességet, jobb terheléselosztást és magasabb rendelkezésre állást biztosít a hálózatban.

A két Ruby-Switch (Ruby-SW1 és Ruby-SW2) g1/0/23 és g1/0/24-es portjain konfiguráltuk.

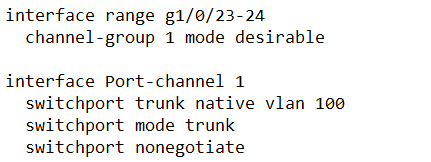


Két port összefogva Ruby-telephelyen



A két port összefogásánál desirable módot konfiguráltunk, ezzel az etherchannel Cisco által fejlesztett ’PAgP’ módját beállítva.

Konfiguráció mindkét switchen



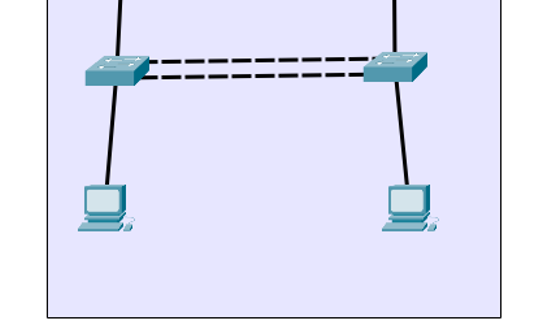
Előnyei:

* Nagyobb sávszélesség
* Redundancia
* Cisco szabadalmaztatott technológia.
* Skálázható
* Hatékonyabb erőforrás-kihasználás

### Port-Security

A Port-Security egy switcheken konfigurálható biztonsági funkció, amely figyeli a portokhoz engedélyezett MAC-címeket, ezzel megakadályozva az jogtalan eszközök csatlakozását a hálózatra.

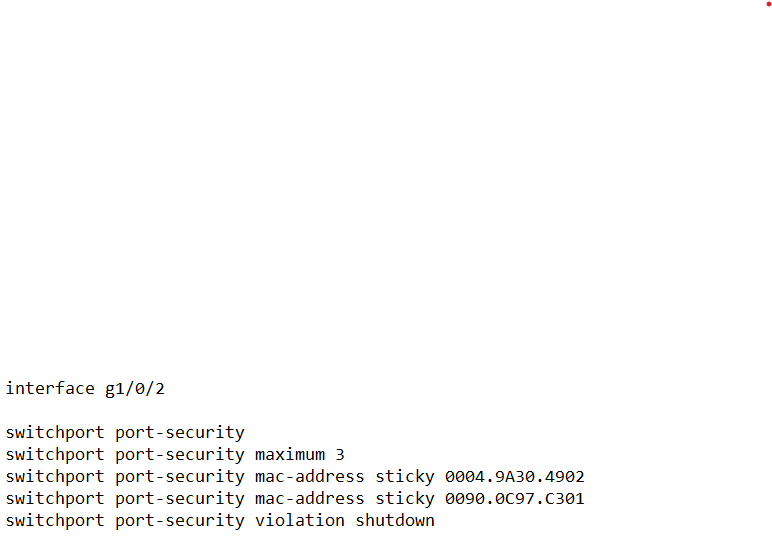
A Ruby Switch1 és Switch2 g1/0/2-es portjain konfiguráltuk.



Max 3

Max 3

A portokon maximum 3 MAC-címet engedélyeztünk, ezzel megakadályozva esetleges támadásokat, például MAC-cím túlcsordulás.



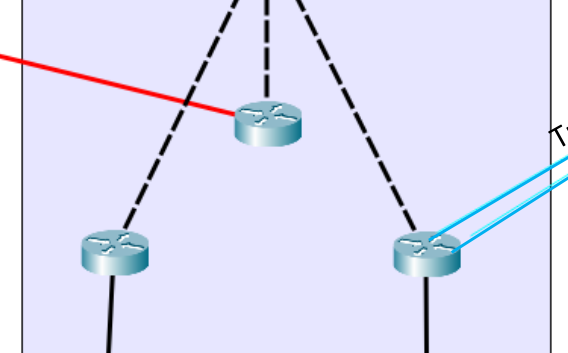
A switchek csak az ismert MAC-címekkel rendelkező eszközöktől származó forgalmat engedik át, ha egy nem ismert eszköz próbál csatlakozni, akkor lekapcsolja a portot.

### GLBP (Gateway Load Balancing Protocol)

A **GLBP** egy Cisco által fejlesztett redundancia- és terheléselosztási protokoll, amely lehetővé teszi több router együttes használatát alapértelmezett átjáróként egy adott hálózati szegmensben, így meghibásodás esetén nem szakad meg a kapcsolat.

A GLBP célja a magas rendelkezésre állás biztosítása mellett a hálózati forgalom hatékony elosztása.

Mi a Ruby telephely routerein alkalmaztuk:



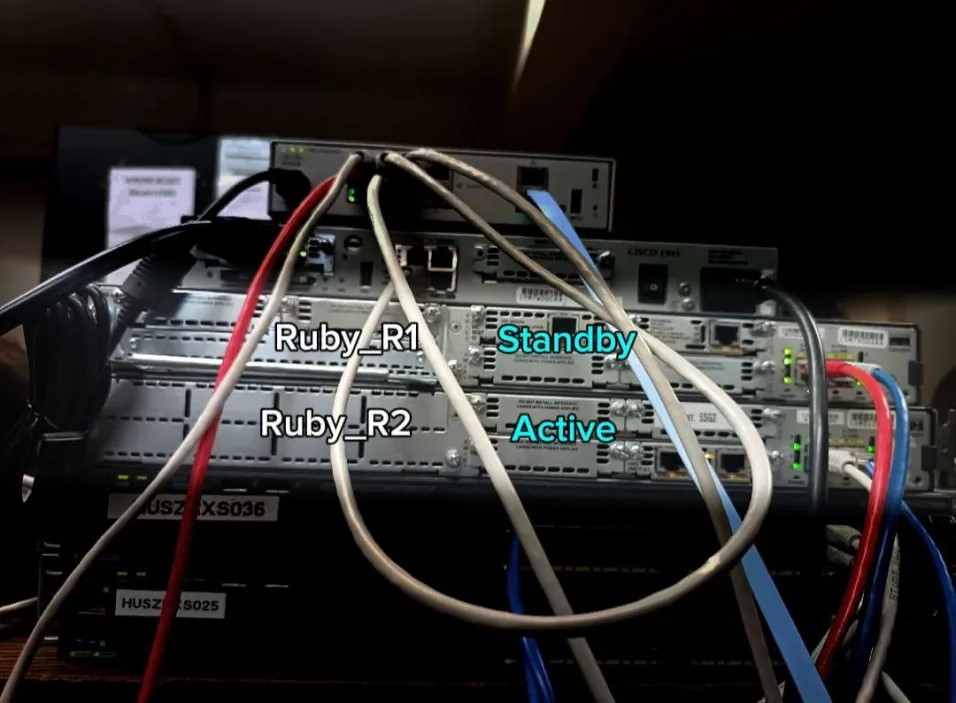
Aktív Router

Készenléti Router

R2

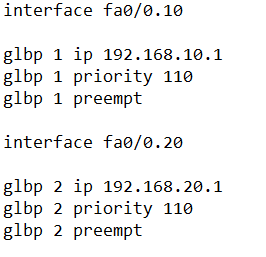
R1

Ha az ’Aktív’ router meghibásodik vagy megszakad a kapcsolat, a készenléti router átveszi a helyét amíg a probléma meg nem oldódik, így nincs leállás.



A két router fa 0/0.10 és fa 0/0.20-as alinterfészein 110-es prioritással konfiguráltuk, és preempt-et is állítottunk .

Mindkét router



Előnyei:

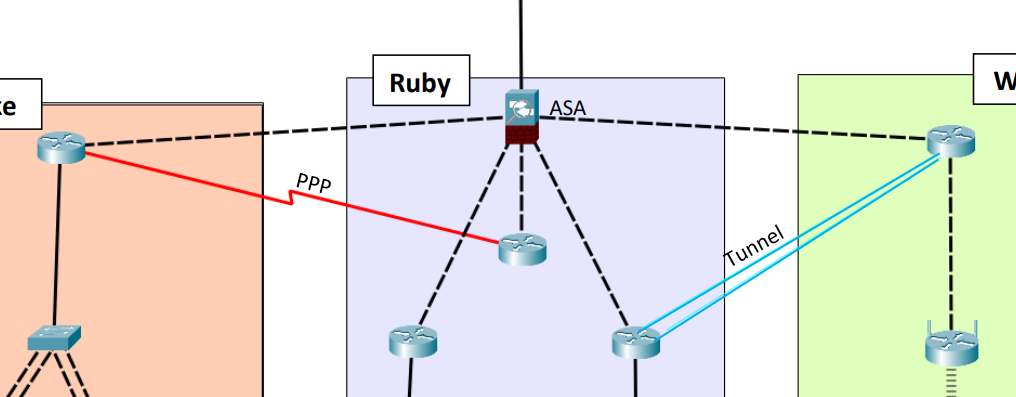
* Magas rendelkezésre állás
* Automatikus átváltás
* Egyszerű kliensoldalon
* Preemption támogatás

### Dinamikus Forgalomirányítás (OSPF)

A dinamikus forgalomirányítás automatikusan kiválasztja a legjobb útvonalat a forgalom elosztása érdekében.

Mi az OSPF útválasztási protokoll-t használtuk.

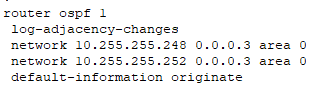
OSPF-el konfigurált eszközök



Az **OSPF (Open Shortest Path First)** egy link-state típusú dinamikus útválasztó protokoll, amelyet IP-alapú hálózatokban alkalmaznak. Célja, hogy a hálózaton belül a lehető legrövidebb és legköltséghatékonyabb útvonalat válassza ki az csomagok továbbításához, magas megbízhatóságot biztosít.

Az OSPF-et 1-es folyamatazonosítóval (router ospf 1) és 0-ás területtel (area 0) konfiguráltuk és beállítottuk, hogy az alapértelmezett útvonalat (0.0.0.0/0) is hirdesse (default-information originate).

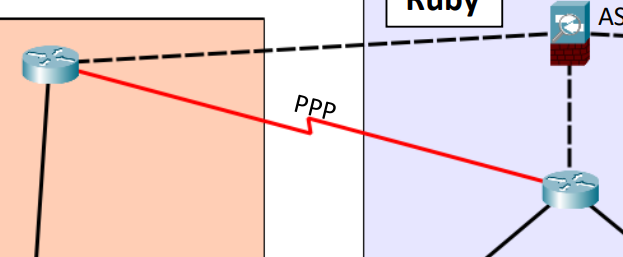
ALTER-ROUTER



### PPP

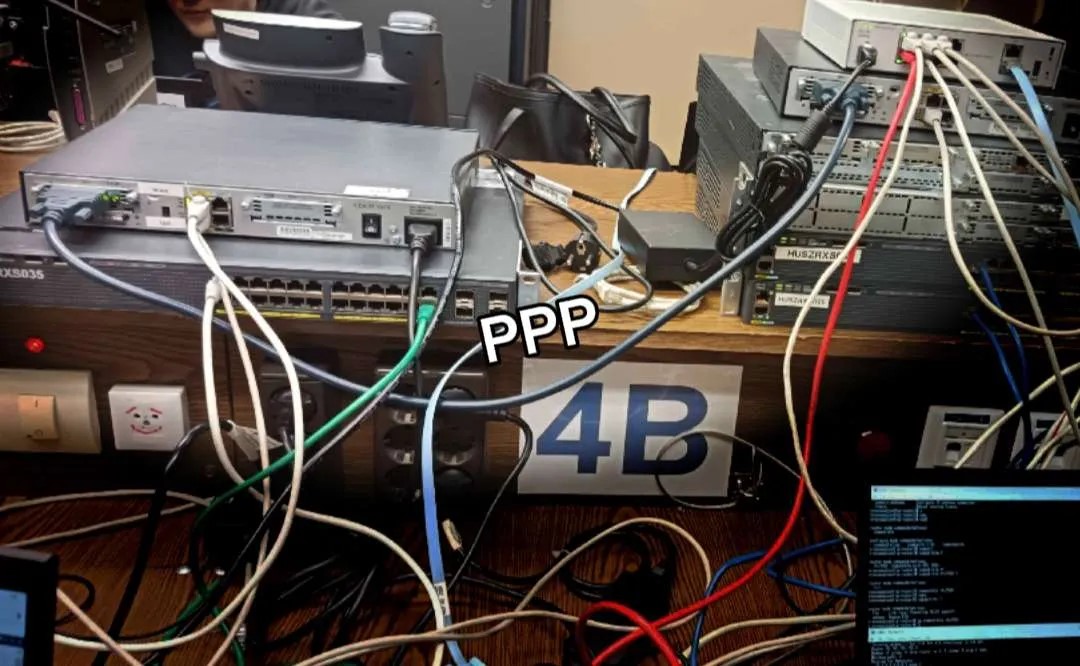
A PPP (Point-to-Point Protocoll) két hálózati eszköz között biztonságos, szabályozott és megbízható adatkapcsolat létrehozására szolgál. Általában router-router között, soros vonalokon alkalmazzák.

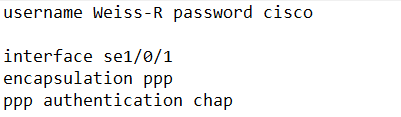
Mi ALTER-ROUTER és Weiss\_R1 közötti serial kapcsolaton alkalmaztuk:



**Blake\_Router**

**ALTER-ROUTER**





ALTER-ROUTER

PPP konfigurálásnál autentikációt állítunk be, amivel a hitelesíti egymást a két eszköz.

Két autentikációs módszer alkalmazható: PAP vagy CHAP, mi CHAP-ot alkalmaztunk mivel ez biztonsági szempontból jobban ajánlott a kettő közül.

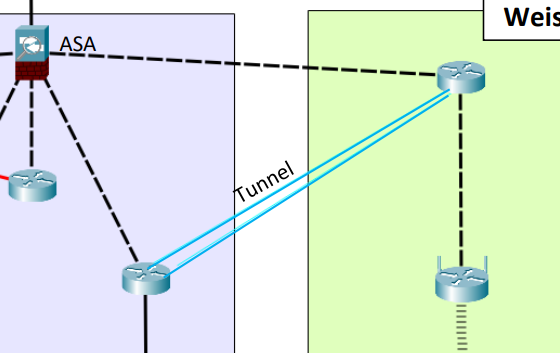
Előnyei:

* Egyszerű és stabil
* Hitelesítést támogat
* Több hálózati protokoll
* Széles körben támogatott

### Tunnel

A tunnel segítségével egy hálózati protokoll csomagjait egy másik, idegen protokollon keresztül továbbítják, mintha egy „alagúton” utaznának, így biztonságosan vagy nem kompatibilis hálózatok között képesek kommunikálni.

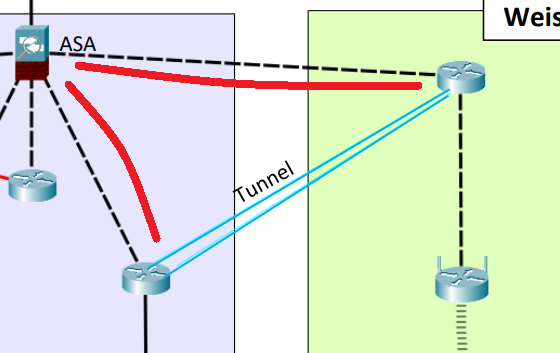
Ruby\_R2 és Weiss\_R1 között alkalmaztuk:

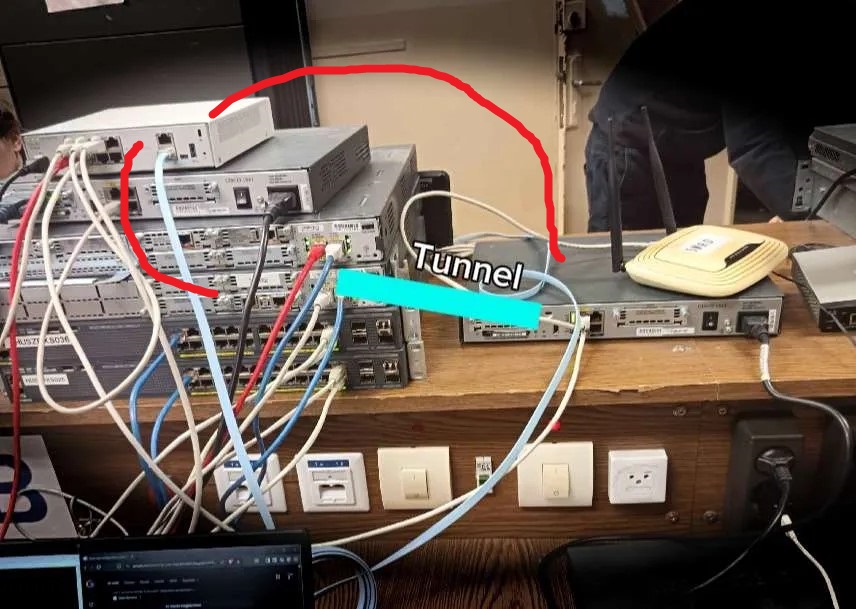


**WeissR1**

**RubyR2**

Tunnel ’igazi’ útvonala (pirossal):





Ruby-R2 tunnel IP-nek 10.0.50.1-et állítottunk, Weiss-R-en pedig 10.0.50.2-t.

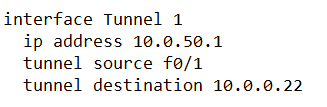
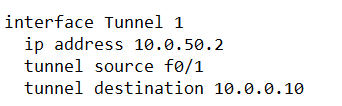


Ruby-R2

Weiss-R



Konfigurálásnál megadtuk a router kimenő interfészét (tunnel source), és a célt, a másik router IP címét (tunnel destination).



Ruby R2

Weiss-router

### NAT (Network Address Translation)

A **NAT** egy hálózati technológia, amely lehetővé teszi a privát IP-címmel rendelkező belső hálózati eszközök és az internet közötti kommunikációt.

A NAT feladata a belső és külső IP-címek közötti címfordítás, ezáltal növeli a hálózat biztonságát és csökkenti a nyilvános IP-címek felhasználásának igényét.

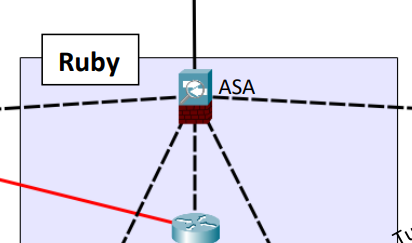
Előnyei:

* IPv4 címmegtakarítás
* Alap biztonság
* Egyszerű belső címzés
* Széles körben támogatott

### ASA Tűzfal

Az **ASA Tűzfal** egy hálózati biztonsági eszköz, amely tűzfal- és hozzáférés-szabályozási funkciókat biztosít.

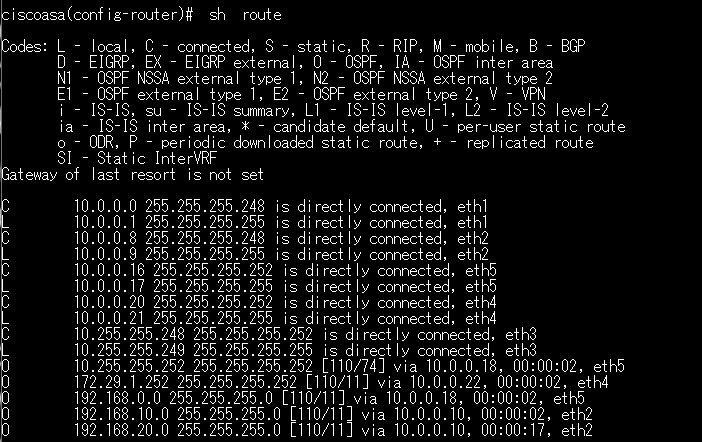
Az ASA tűzfal feladata a belső hálózat védelme, a bejövő és kimenő forgalom ellenőrzése, valamint a jogosulatlan hozzáférések megakadályozása.





Az ASA tűzfal a hálózatot különböző **biztonsági zónákra** bontja, amelyekhez biztonsági szintek tartoznak.

A magasabb biztonsági szintről az alacsonyabb felé irányuló forgalom alapértelmezetten engedélyezett, míg az ellenkező irányú forgalom tiltott, kivéve, ha azt kifejezetten engedélyezzük.



## 

## 

## Weiss Telephely

Használt technológiák:

* Tunnel
* Wifi DHCP

### Wifi DHCP

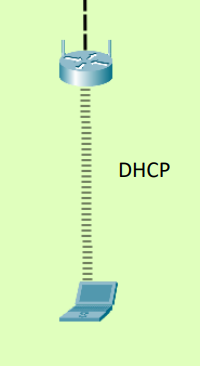
DHCP (Dynamic Host Configuration Protocoll) segítségével automatikusan osszon ki hálózati beállításokat az eszközöknek.

A DHCP segítségével a kliens automatikusan megkapja:

* IP-címét
* Alhálózati maszkot
* Alapértelmezett átjárót (gateway)
* DNS szerver címét

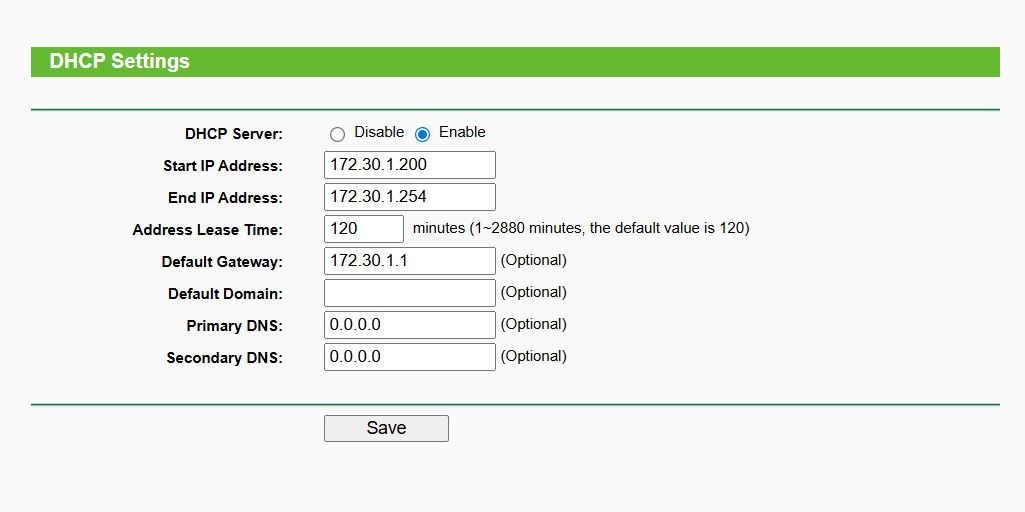
Így nem kell kézzel konfigurálni minden egyes eszközt.

A Weiss telephelyen a kliensek Wifi-Router-től kapnak címet DHCP-n keresztül



A wifi router webes felületén végeztük el a szükséges beállításokat.

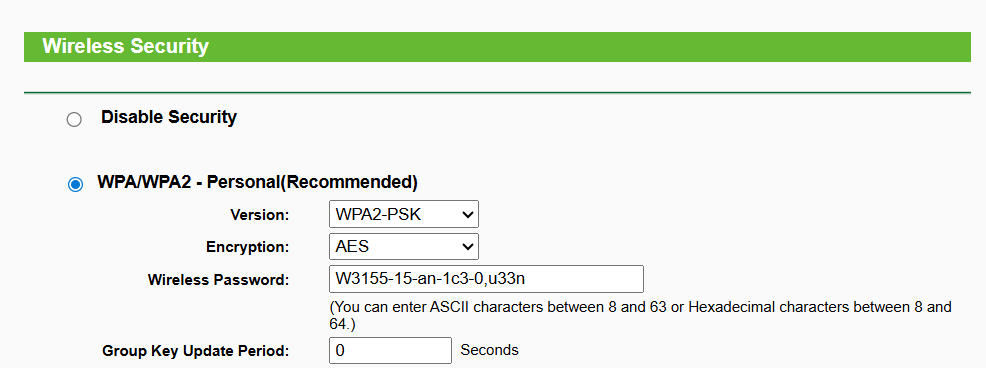
A DHCP 172.30.1.200 – 172.30.1.254-ig oszt címeket.

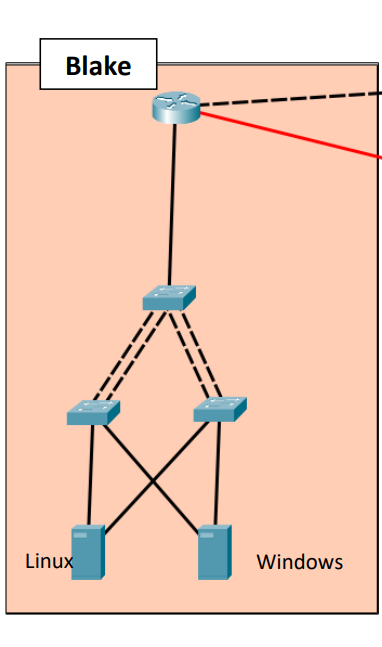


DHCP beállításai

A Wifi SSID-nak ’Weiss\_WLAN’-t állítottunk, ez a wifi neve.

A wifi-re csatlakozás titkosított, a verzója WPA2-PSK, a titkosítás AES.





## 

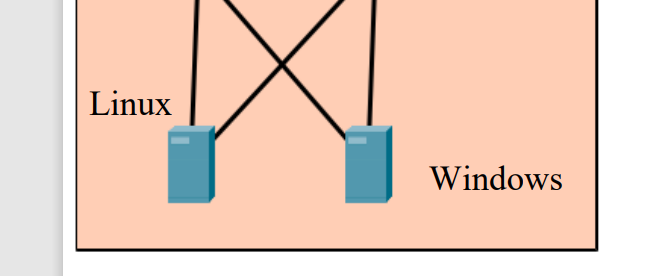
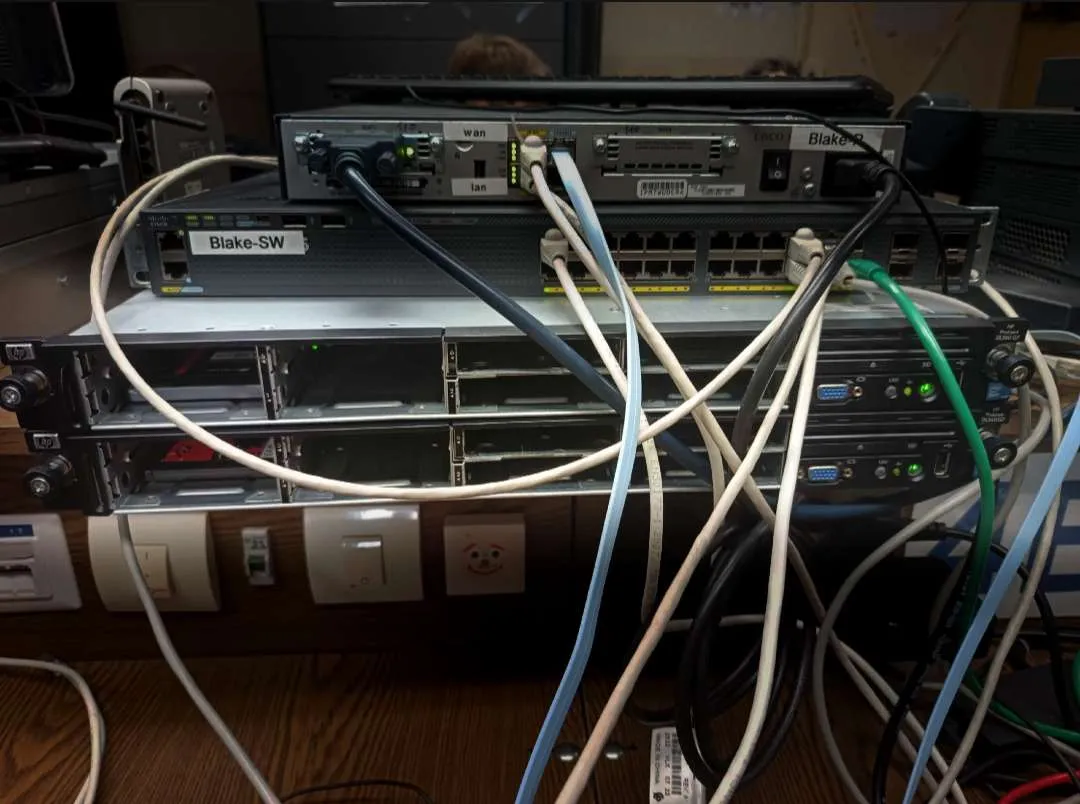
## Blake Telephely

Használt technológiák:

* PPP
* Linux server
* Windows server

**Szerverek**

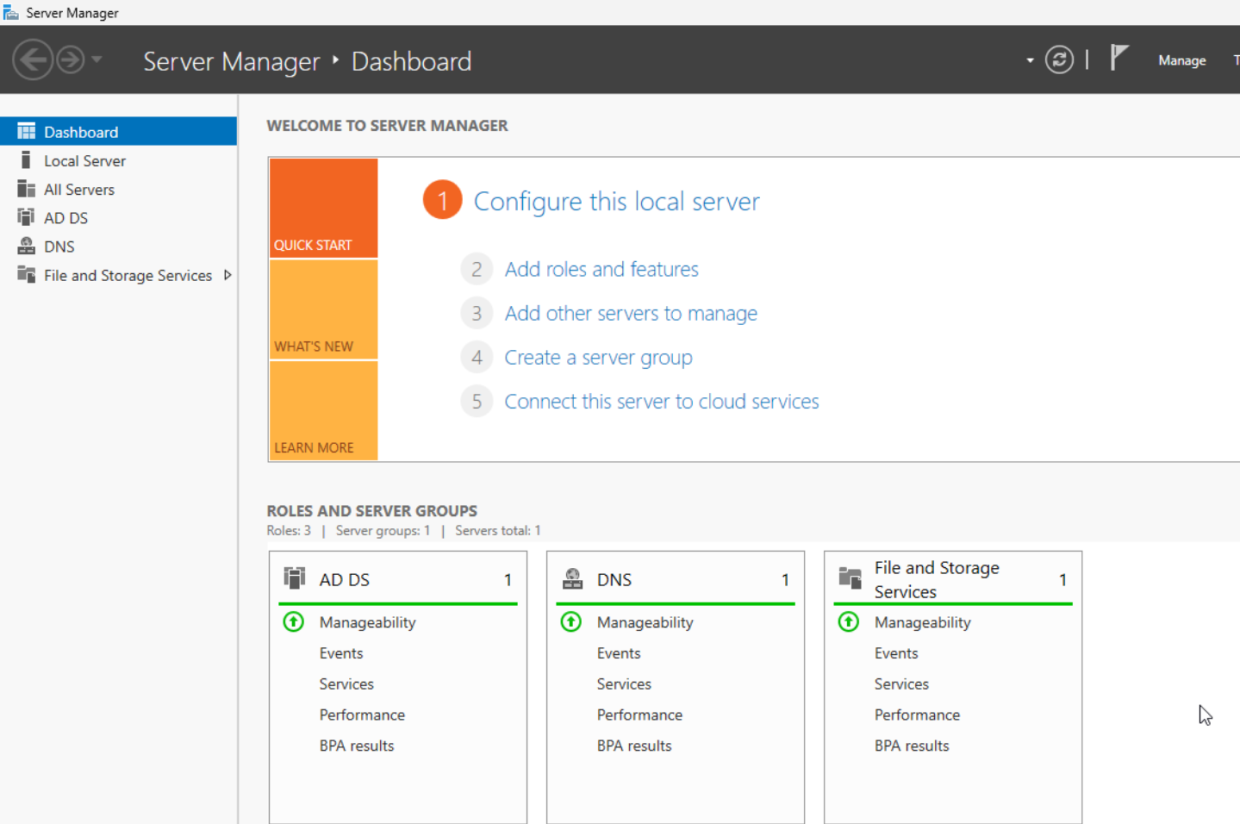
A **szerverek** fontos szerepet töltenek be a hálózat működésében, mivel ezek biztosítják a kliensek számára a szükséges szolgáltatásokat. A szerverek feladata a hálózati erőforrások kiszolgálása, kezelése és védelme, valamint a folyamatos és megbízható működés biztosítása.

Szerverek a Blake telephelyen:

### Windows Server

A Windows Server alkalmazása a hálózati projektben lehetővé teszi a központi menedzsmentet, a biztonságos felhasználókezelést és hálózati szolgáltatások (pl. DHCP, DNS, FTP) megbízható működtetését.

Mi Windows Server 2025-ös verzión végeztük el a feladatokat, a Server Manager konzolban.



A Windows szerveren telepített szolgáltatások:

* Active Directory
* DNS szerver
* IIS Webszerver
* Windows Server Backup

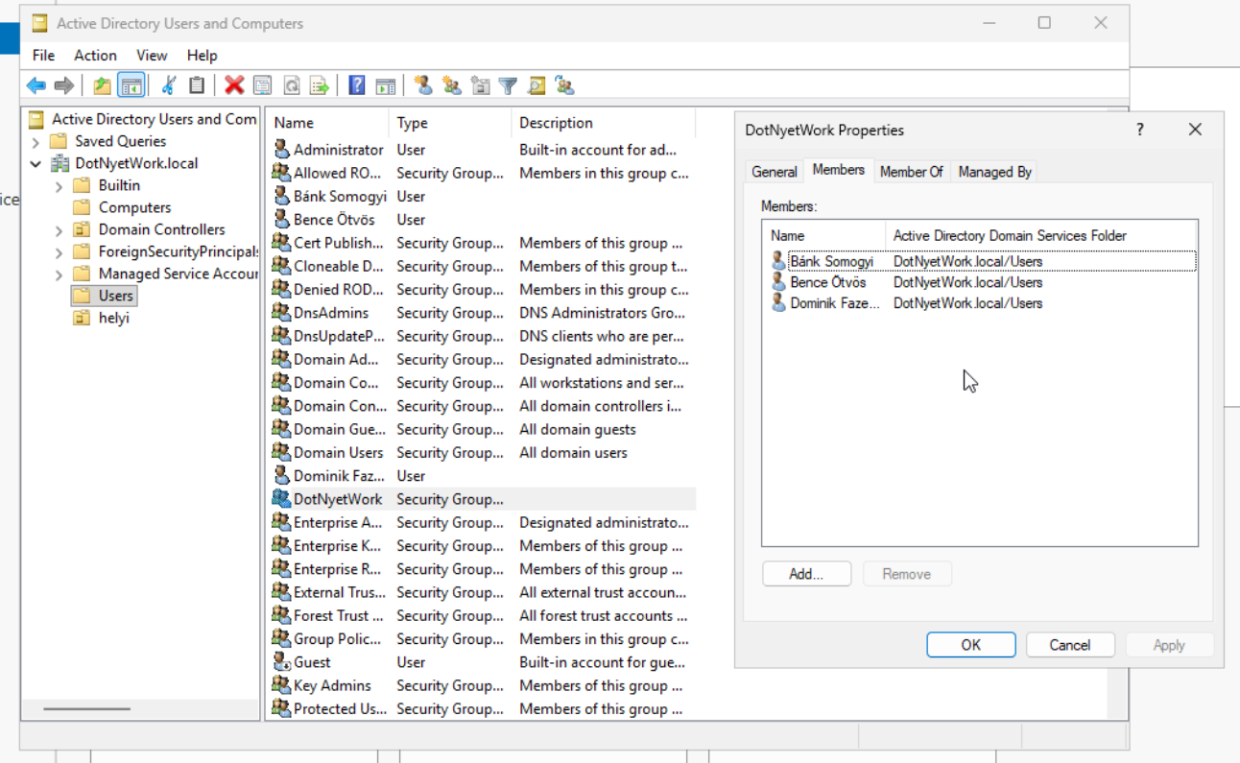
#### Active Directory Domain Services (AD DS)

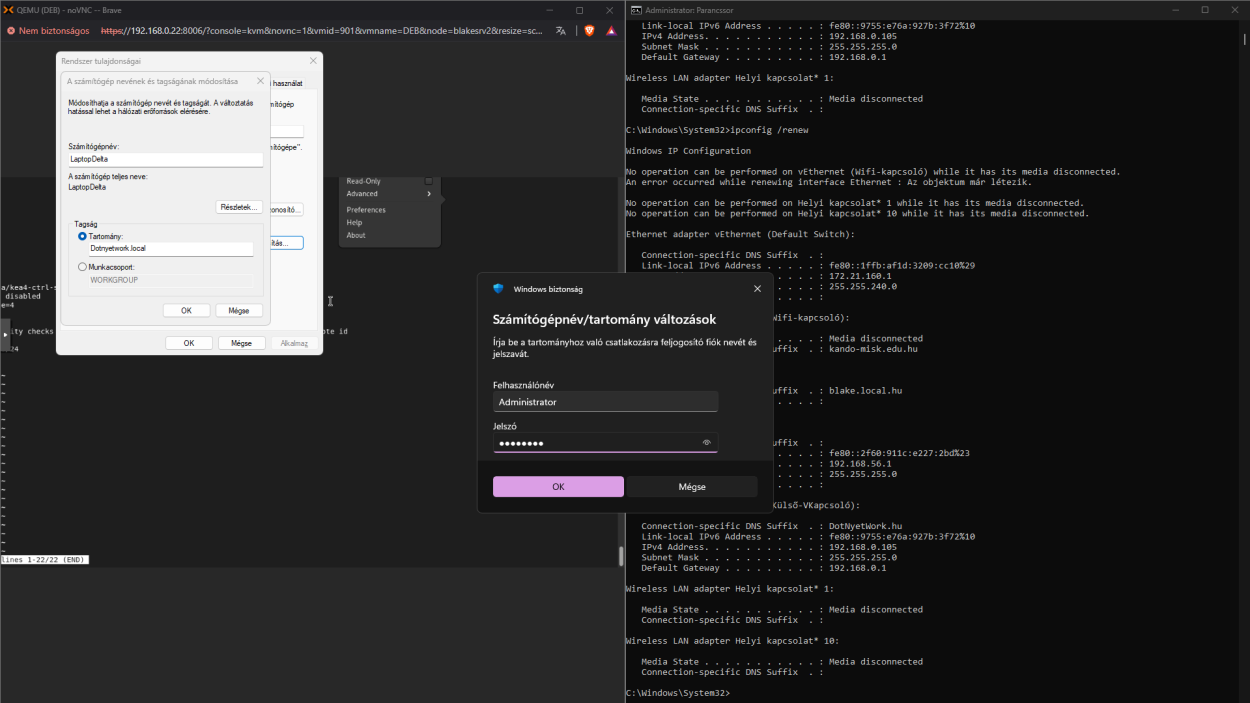
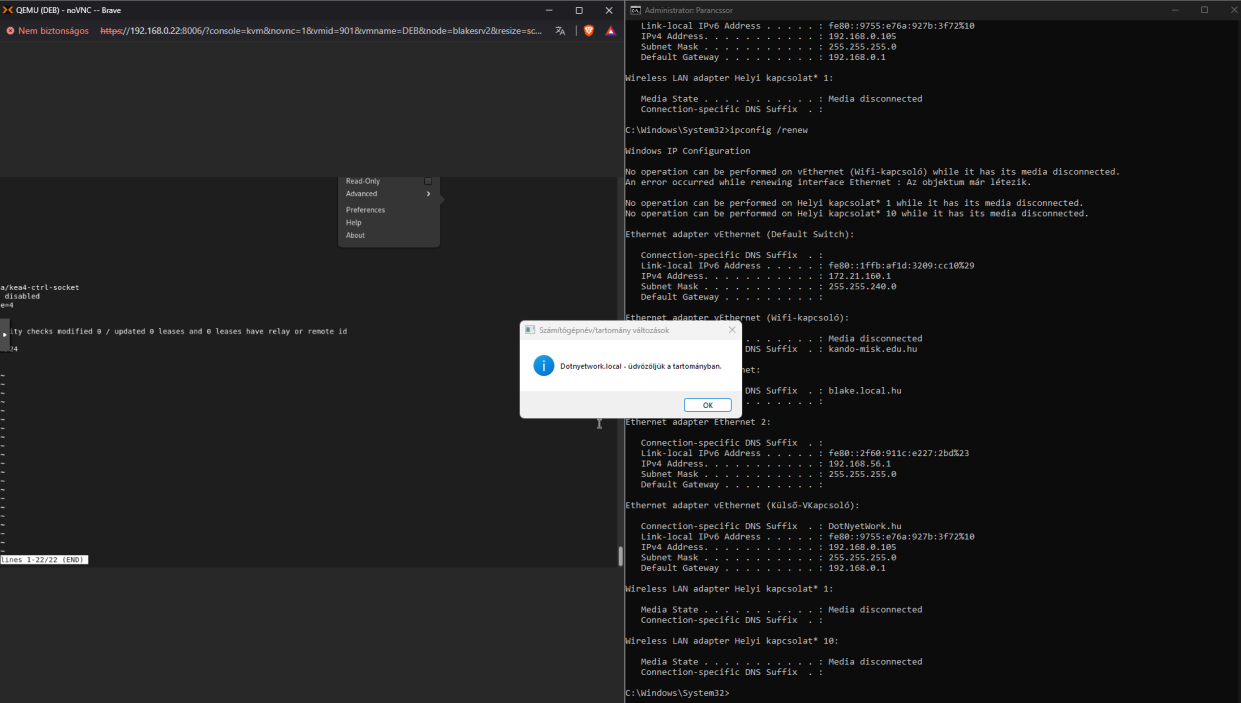
A Windows server egyik fontos eleme az **AD DS**, amely a tartományalapú hitelesítést és jogosultságkezelést valósítja meg.

Az AD DS segítségével központilag kezelhetők a felhasználói fiókok és csoportok és biztonsági szabályok alkalmazhatóak a tartományban lévő számítógépeken.

A tartománynak DotNyetWork.local-t állítottunk be.

A projekt tagjai létrehozva felhasználókként a DotNyetWork csoportban:



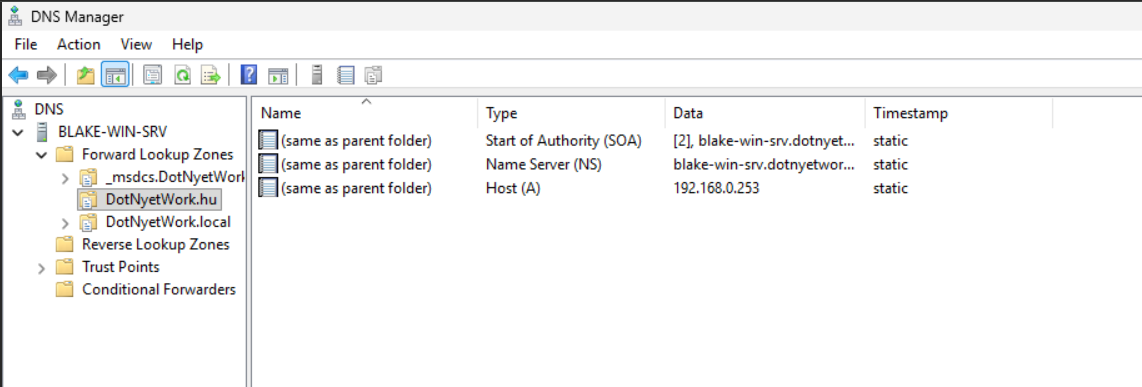
Egy laptop beléptetése a DotNyetWork tartományba:

#### Domain Name System (DNS)

A **DNS** szolgáltatás biztosítja a hálózati eszközök névfeloldását: A domain neveket (pl. www.DotNyetWork.com) a számítógépek számára értelmezhető IP-cmekké fordítja le.

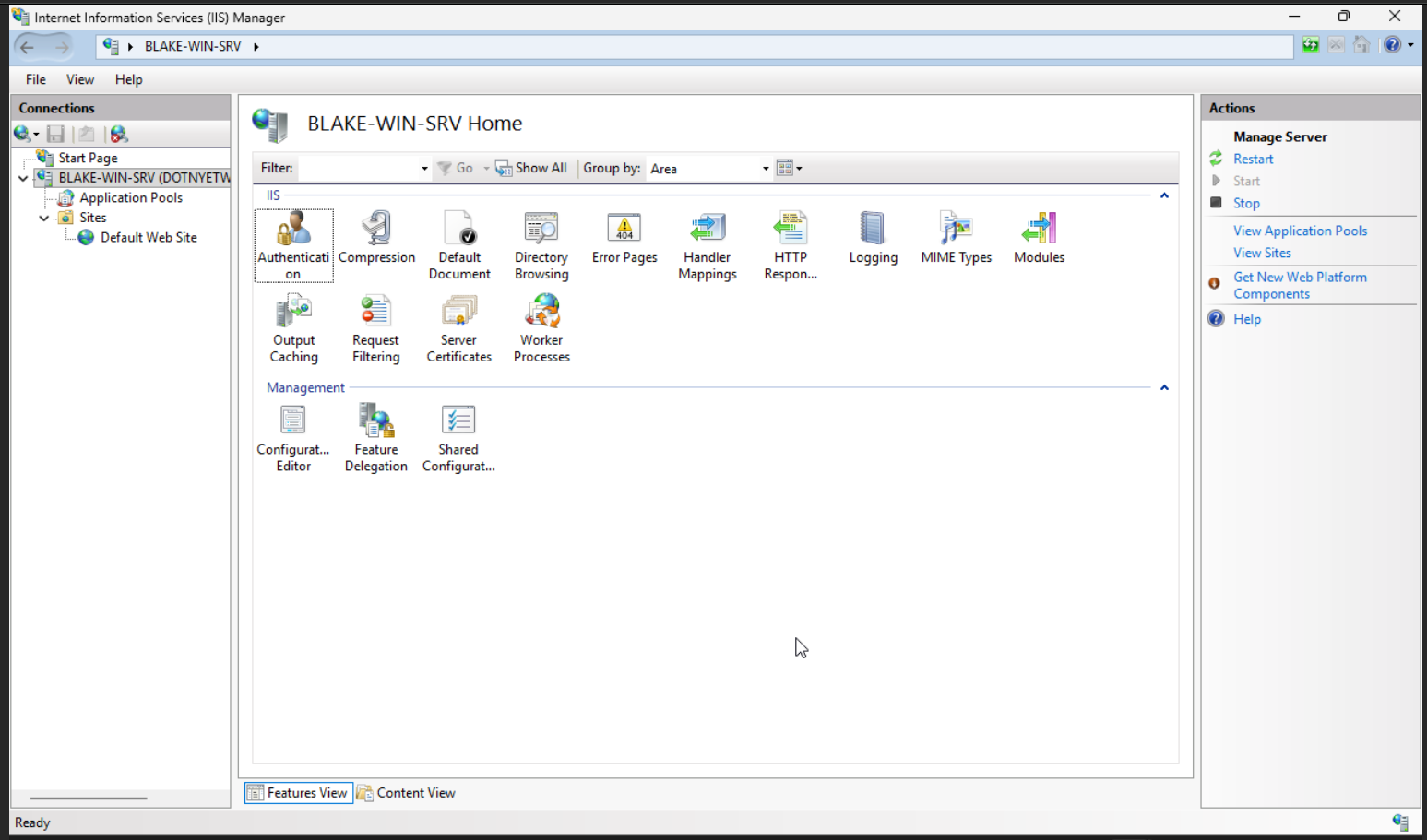
A Windows Server DNS-szervere lehetővé teszi a belső hálózati nevek hatékony kezelését és integráltan működik az AD DS-el.

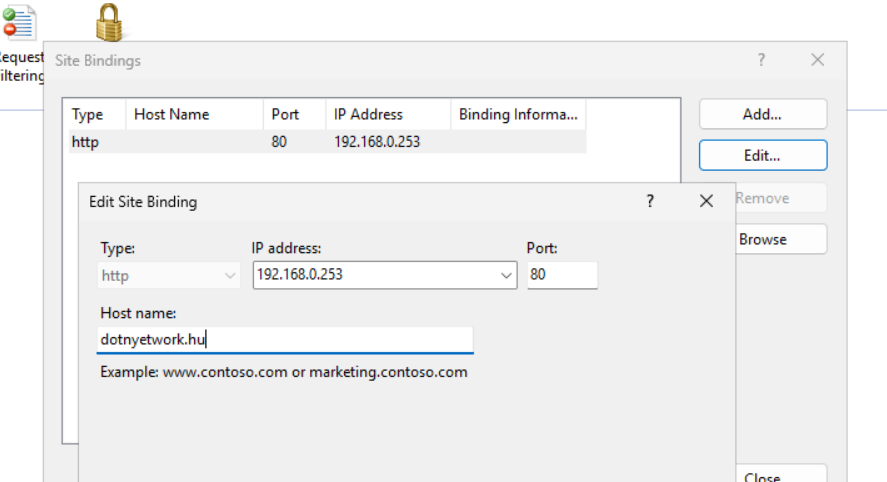


A dotnyetwork.hu címen elérhető a weboldal, beállítás az alábbi képen látható.

#### IIS Webszerver

A **IIS (Internet Information Services)** egy webszerver szolgáltatás, amely Windows Server környezetben biztosít weboldalak és webes alkalmazások futtatását. A hálózati projekt során az IIS feladata a webes tartalmak kiszolgálása HTTP és HTTPS protokollokon keresztül.

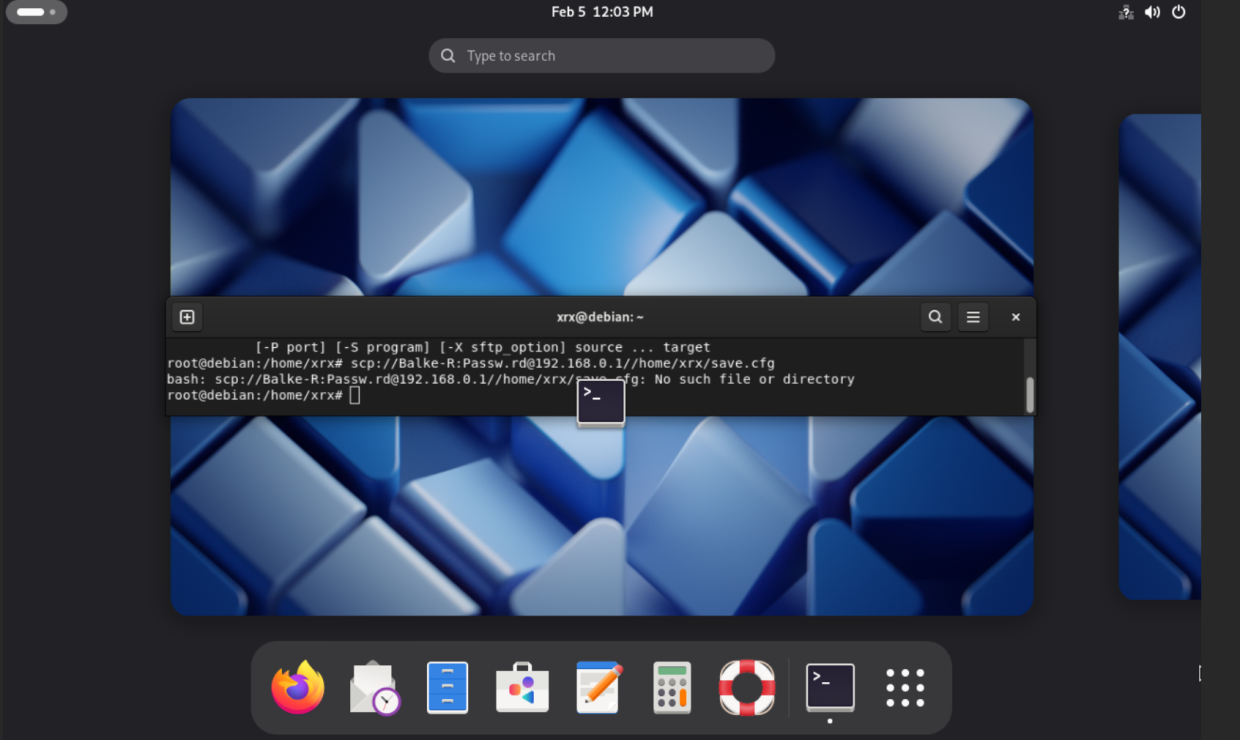
Az IIS konzolja, ahol a webszerver beállításait konfiguráltuk:

A webszerver IP címe 192.168.0.253, ezen az IP-n érjük el a weboldalt http-vel (DNS szerverrel a dotnyetwork.hu címen is elérhető).

### Linux Server

A **Linux szerver** egy nyílt forráskódú operációs rendszerre épülő szervermegoldás, amely nagyfokú stabilitást, biztonságot és rugalmasságot biztosít a hálózatban. A Linux szerver feladata különböző szolgáltatások megbízható és hatékony biztosítása, valamint a rendszer erőforrásainak optimális kihasználása.

Linuxon Debian környezetben végeztük el a feladatokat parancsokkal a terminálban.



A Linux szerveren telepített szolgáltatások:

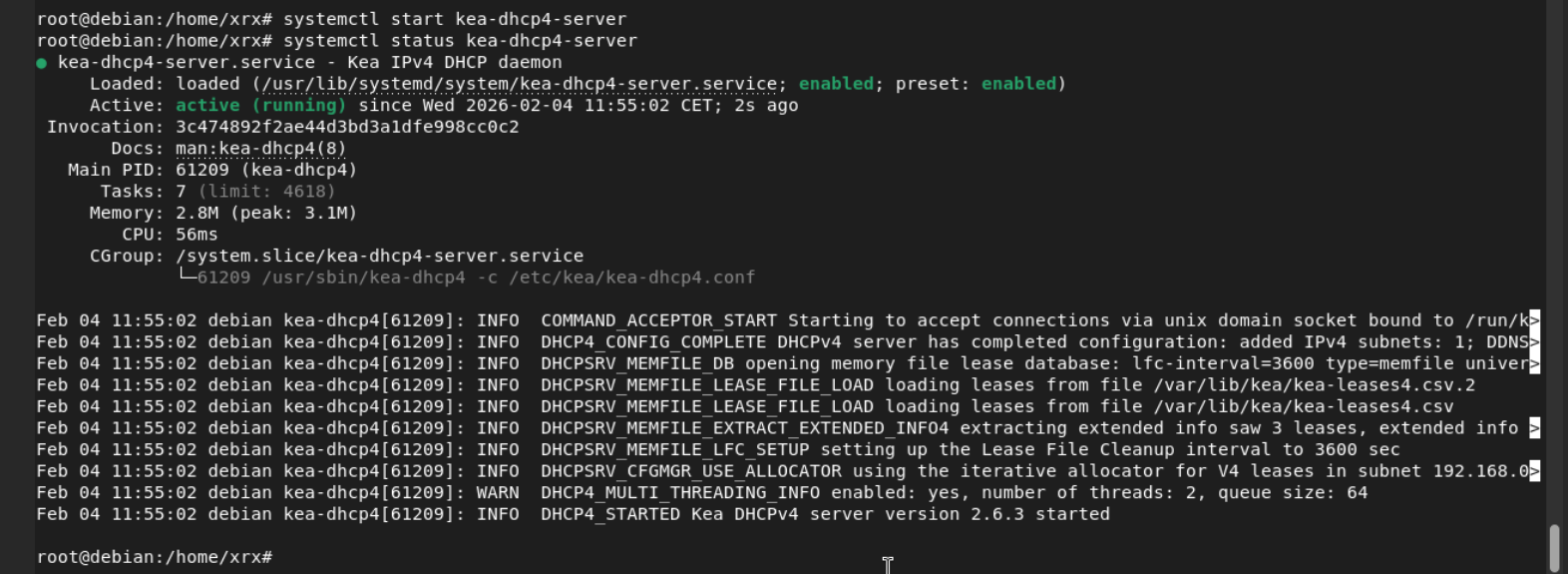
* DHCP szerver (Kea)
* FTP (vftpd)
* Samba

#### DHCP (Kea)

A **DHCP** szolgáltatás feladata az IP-címek és egyéb hálózati adatok automatikus kiosztása a kliensek számára.

A Linux szerveren futó DHCP-szolgáltatás biztosítja, hogy a hálózatra csatlakozó eszközök kézi konfiguráció nélkül, gyorsan és hibamentesen kapják meg a szükséges beállításokat, mint például az IP-címet, a Deafult-gatweay-t és a DNS-t.

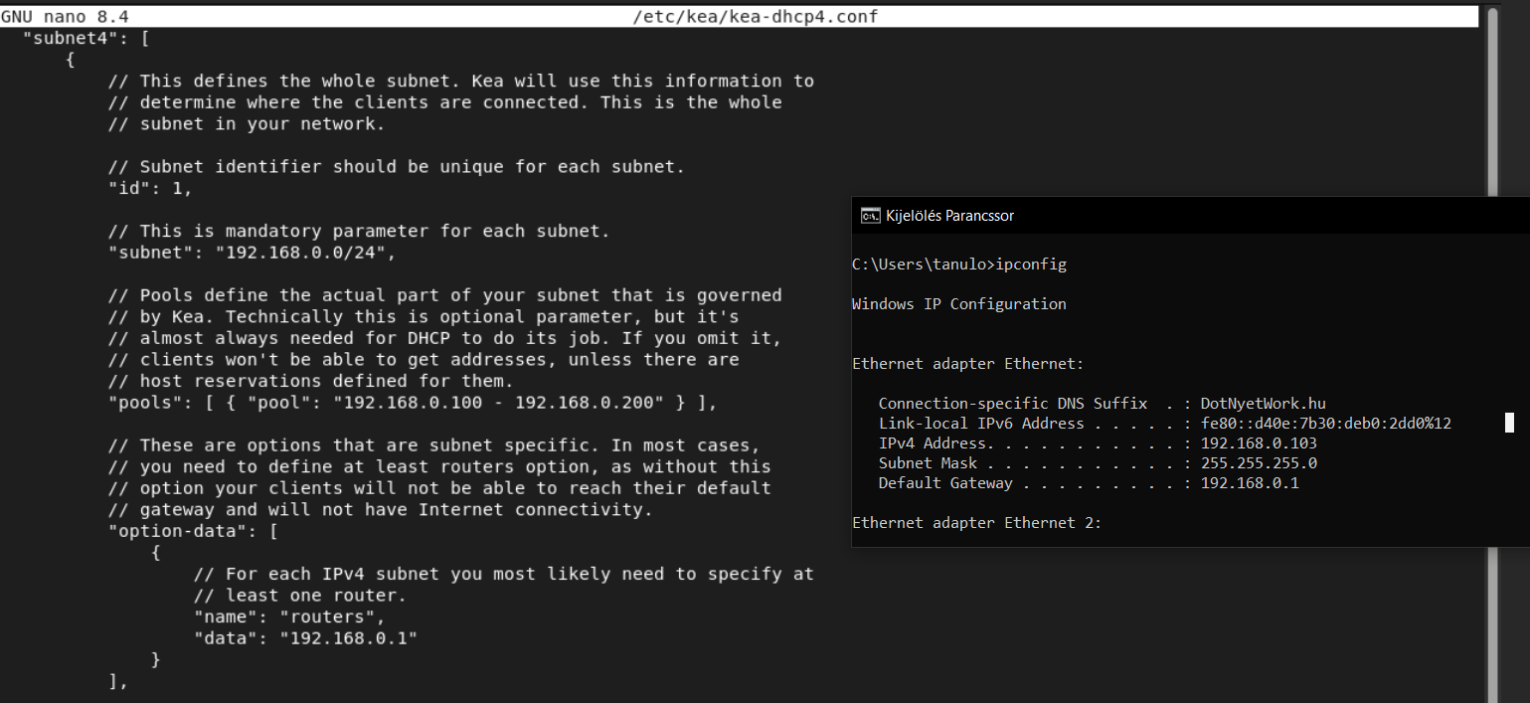
Linux szerveren a Kea-Dhcp szolgáltatást telepítettük a DHCP konfigurálásához (apt install kea-dhcp4-server).

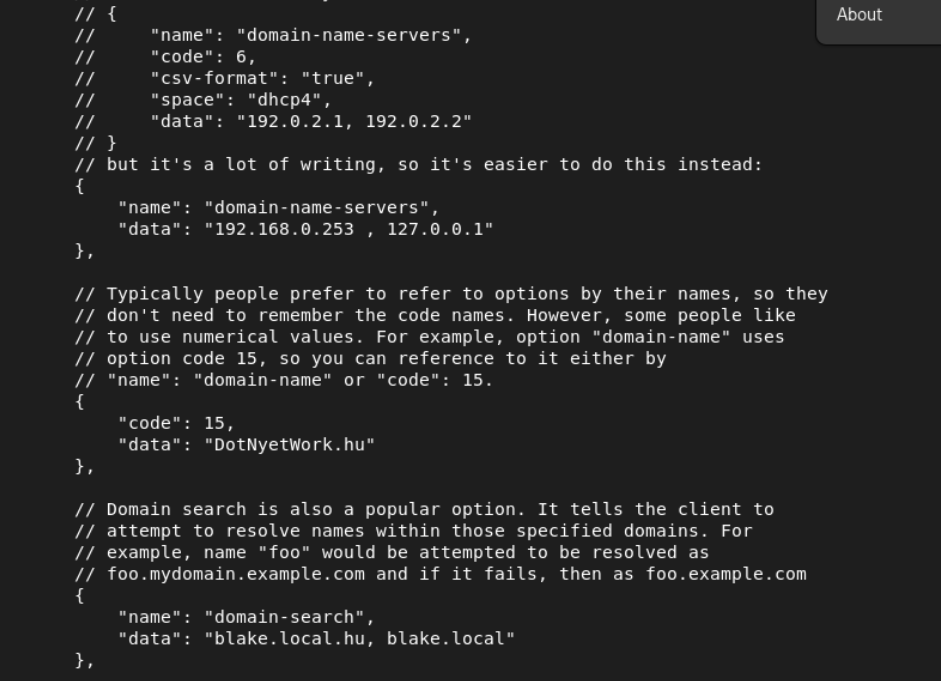


A kea-dhcp4.conf fájlban végeztük el a beállításokat. (nano /etc/kea/kea-dhcp4.conf)

A tartomány 192.168.0.100-tól 192.168.0.200-ig oszt IP címeket /24-es maszkkal.

A default gateaway Blake R IP címe (192.168.0.1).



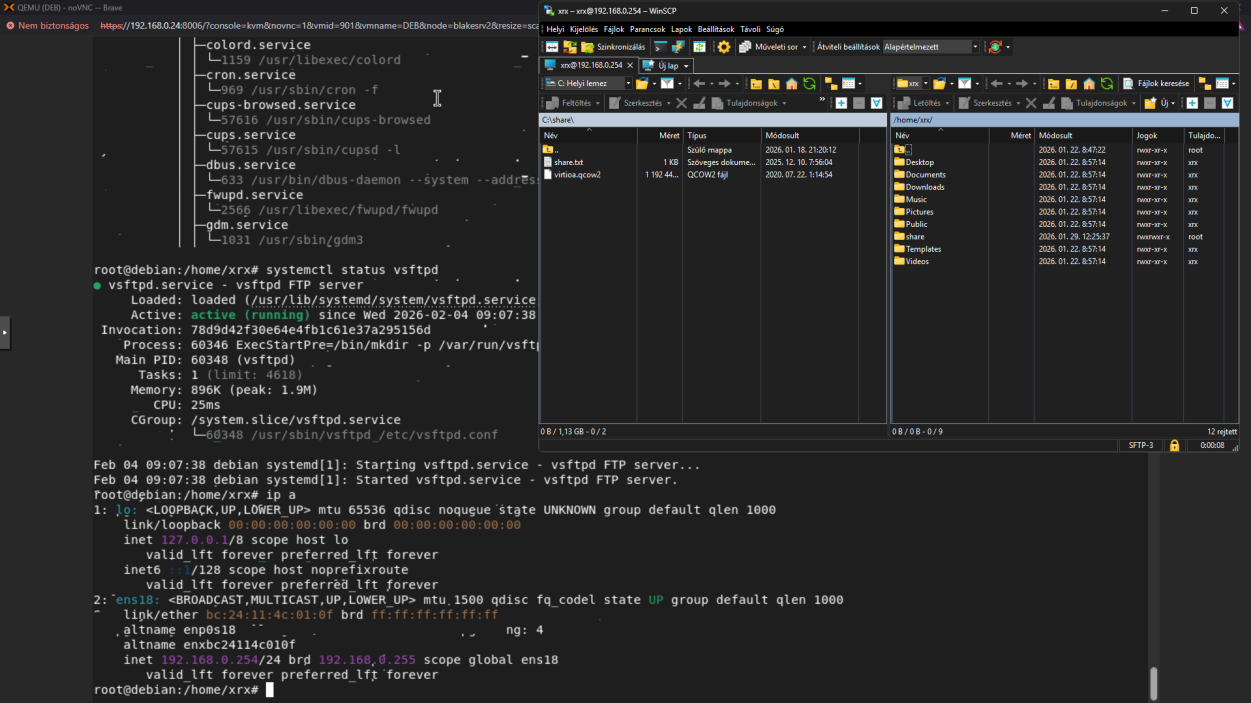


#### File Transfer Protocol (FTP)

Az **FTP** szolgáltatás fájlok feltöltésére és letöltésére szolgál a szerver és a kliensek között. Linuxon gyakran alkalmazott megoldás például a vsftpd vagy ProFTPD, amelyek biztonságos és hatékony fájlátvitelt tesznek lehetővé.

A jogosultságkezelés segítségével meghatározható, hogy mely felhasználók férhetnek hozzá az FTP-szerverhez, illetve milyen műveleteket hajthatnak végre.

FTP szervernek vsftpd-t telepítettünk (apt install vsftpd).



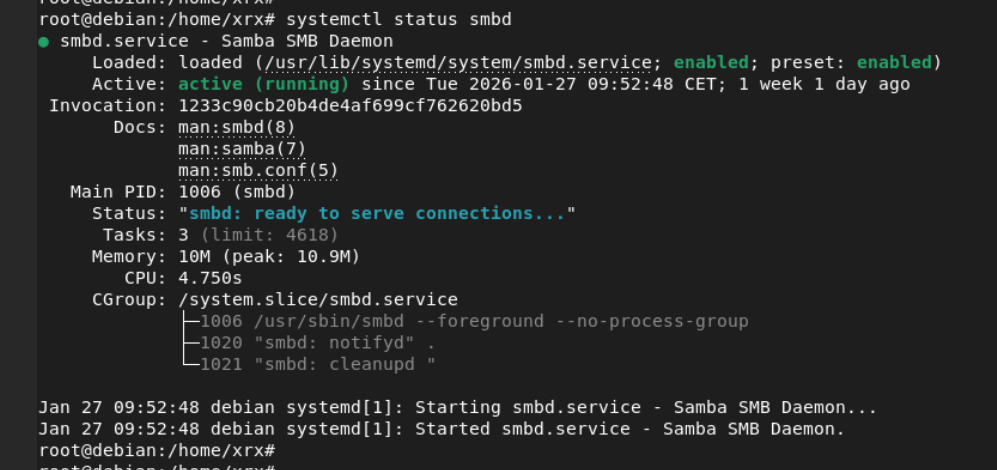
Az FTP szerver konfigurációit a vsftpd.conf-ban végeztük el (nano /etc/vsftpd.conf), az alábbi képen látható a konfigurációs fájl.



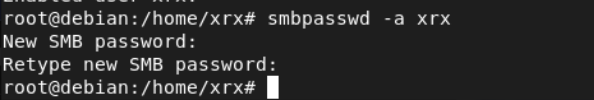
#### Samba (Fájlmegosztás)

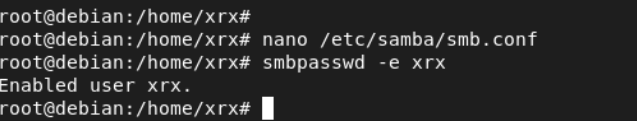
A Samba szolgáltatás lehetővé teszi a fájlmegosztást Linux és Windows rendszerek között. A Samba segítségével a Linux szerver Windows hálózatban is natív módon jelenik meg, így a felhasználók egyszerűen érhetik el a megosztott mappákat.

A hozzáférés felhasználónként vagy csoportonként szabályozható, ezáltal biztosítható az adatok védelme és az elkülöníített hozzáférés.

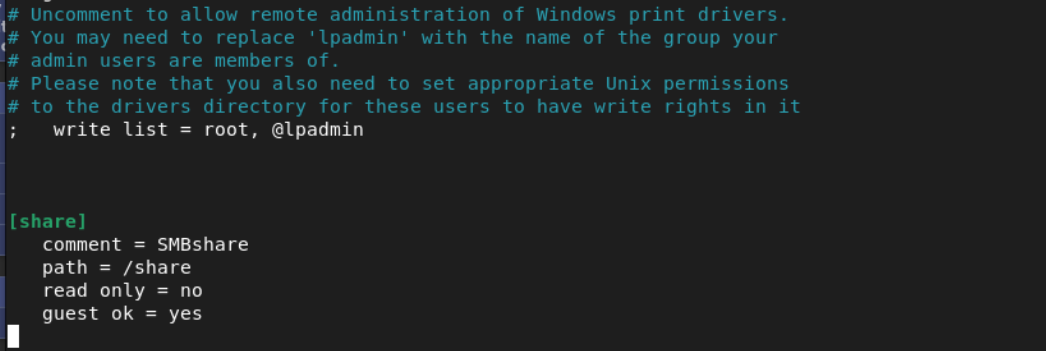
Linuxon telepítettük a Samba szolgáltatást (apt install samba).

Létrehoztunk egy Samba felhasználót xrx néven.

Jelszó beállítása a felhasználóhoz:

Samba felhasználó engedélyezése:

Létrehoztunk egy share nevű megosztott mappát, amit hozzáadtunk a samba konfigjához (nano /etc/samba/smb.conf).



# Dokumentáció linkek:

Prezentáció: [Cím nélküli prezentáció (5).pptx](Cím%20nélküli%20prezentáció%20(5).pptx)

Hálózat szimuláció:

Videó:

Tesztelési Dokumentáció: <Projekt_Word_Tesztelés.docx>

# Forrásjegyzék:

[Netacad](https://www.netacad.com/)

[ChatGPT](https://chatgpt.com/)