**Hyperion**

**Projekt**

**dokumentáció**

**Hálózat**

**És**

**Alkalmazás**

**Készítette:**

**Laczkó Dávid**

**Czipri Gergő Bertalan**

**Kovács Dániel**

**Készült:**

**2024. 12. 1.**

Tartalom

[**Feladat 3**](#_Toc183980782)

[**Használt programok 3**](#_Toc183980783)

[**Hálózat leírása 3**](#_Toc183980784)

[**Leírás 3**](#_Toc183980785)

[**A kód futtatásának hálózati megoldásai 3**](#_Toc183980786)

[**Routerek beállításai 4**](#_Toc183980787)

[**Port és IP cím táblázat 4**](#_Toc183980788)

[**Tesztelés 5**](#_Toc183980789)

[**Alkalmazás leírás 6**](#_Toc183980790)

[**Leírás 6**](#_Toc183980791)

[**Alkalmazás funkciói 6**](#_Toc183980792)

[**Connection oldal 6**](#_Toc183980793)

[**Interface Config oldal 7**](#_Toc183980794)

[**DHCP beállítás oldal 8**](#_Toc183980795)

[**Time Config oldal 8**](#_Toc183980796)

[**Blacklist oldal 9**](#_Toc183980797)

[**Hostname oldal 9**](#_Toc183980798)

[**Syslog szerver 10**](#_Toc183980799)

[**Leírás 10**](#_Toc183980800)

[**Tesztelés 10**](#_Toc183980801)

[**A projekt nehézségei 11**](#_Toc183980802)

[**Hálózati eszközök configjai 11**](#_Toc183980803)

[**R1 11**](#_Toc183980804)

[**R2 12**](#_Toc183980805)

[**R3 13**](#_Toc183980806)

[**R4 14**](#_Toc183980807)

[**R5 15**](#_Toc183980808)

# Feladat

A feladatban egy Graphical user interface (Továbbiakban GUI-t) kellett létrehozni Python kódolásban egy hálózati eszközre (router vagy switch). Továbbá létre kellett hozni egy minimum 6 eszközből álló hálózatot. Ezeken felül létre kellett hozni egy syslog szervert, amin látszódnak a vezérelt eszközöknek a változtatásai.

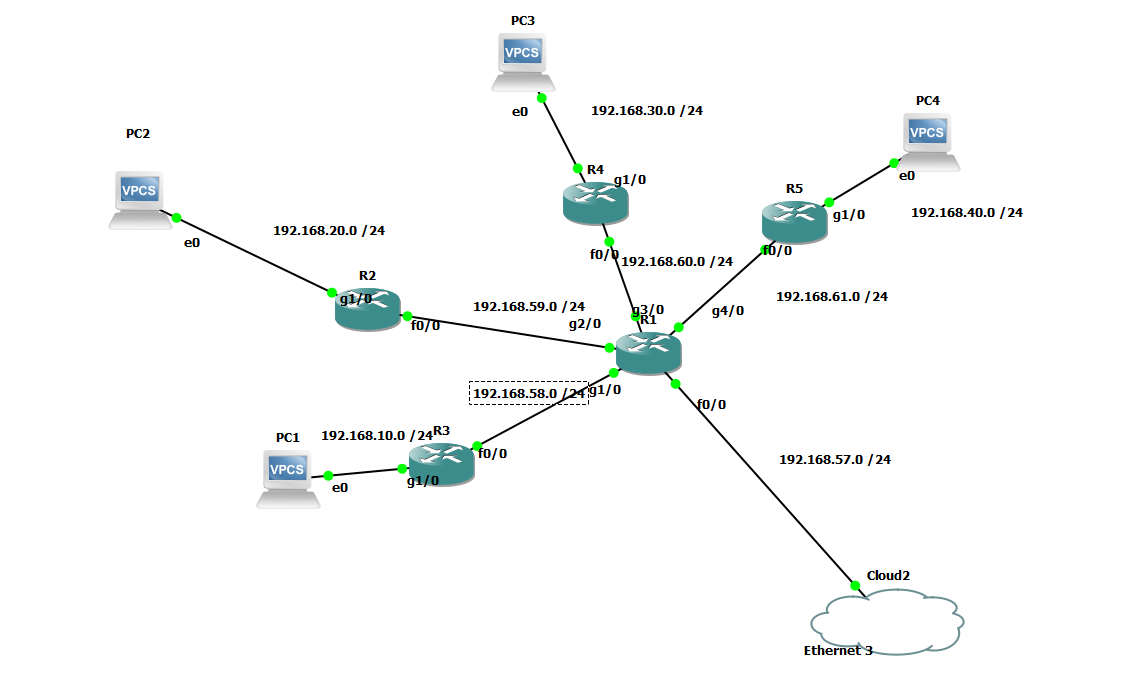
# Használt programok

A hálózat megvalósításához a GNS3 nevezet programot használtuk. A syslog szerverhez a Visual Syslog server nevezetű alkalmazást használtuk. Programozáshoz Visual Studio Code-ot használtunk és a projektünk vezetéséhez és a projekthez tartozó fájlok tárolásához Githubot használtunk.

# Hálózat leírása

## Leírás

A hálózat 6 routerből áll, amihez összesen 5 PC-csatlakozik és egy cloud, ami a saját számítógépünket szimbolizálja. Routereknek Cisco 7200-asakat választottunk.



## A kód futtatásának hálózati megoldásai

A kódot úgy terveztük, hogy saját gépről is lehessen futtatni. A számítógépnek felvettünk egy loopback interface-t, aminek adtunk egy IP (jelen esetben 192.168.57.10) címet, így a saját gépünkről is elérhetőek a virtualizációs program eszközei.

## Routerek beállításai

A routerekben a következő beállítások találhatóak meg:

* Jelszó: admin
* 1-1 DHCP pool az adott hálózatnak megfelelően
* Domain lookup kikapcsolása
* Sysloggal kapcsolatos beállítások
* SSh
  + Felhasználónév: admin
  + Jelszó: admin
  + RSA kulcs: 2048
  + Verzió 2
  + Vty 0 15-ön csak SSH és lokális bejelentkezés
* RIP
  + Auto-summary kikapcsolása
  + Verzió 2
  + Az adott hálózatok beírása a routerekbe

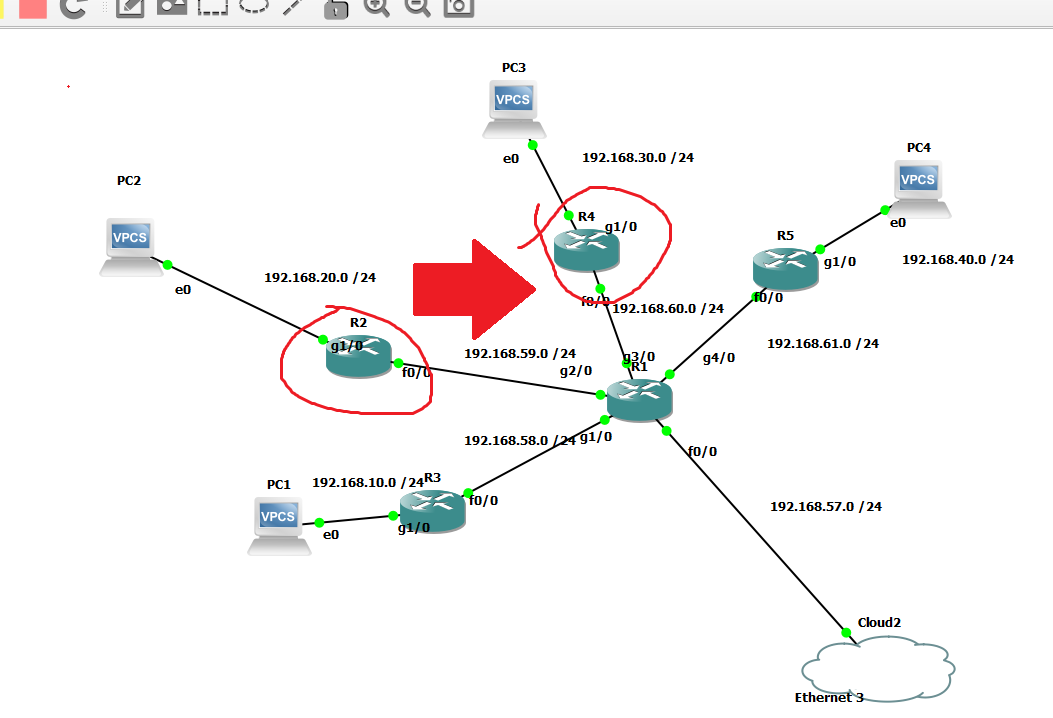
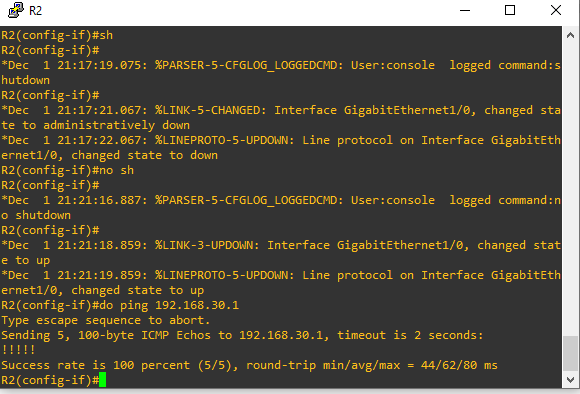
## Port és IP cím táblázat

|  |  |  |  |
| --- | --- | --- | --- |
| **Eszköz** | **Port** | **Hova csatlakozik** | **IP-cím** |
| R1 | F0/0 | Cloud2  Ethernet3 | 192.168.57.11 /24 |
| G1/0 | R3  f0/0 | 192.168.58.11 /24 |
| G2/0 | R2  f0/0 | 192.168.59.11 /24 |
| G3/0 | R4  f0/0 | 192.168.60.11 /24 |
| G4/0 | R5  f0/0 | 192.168.61.11 /24 |
| R2 | F0/0 | R1  G2/0 | 192.168.59.10 /24 |
| G1/0 | PC2  e0 | 192.168.20.1 /24 |

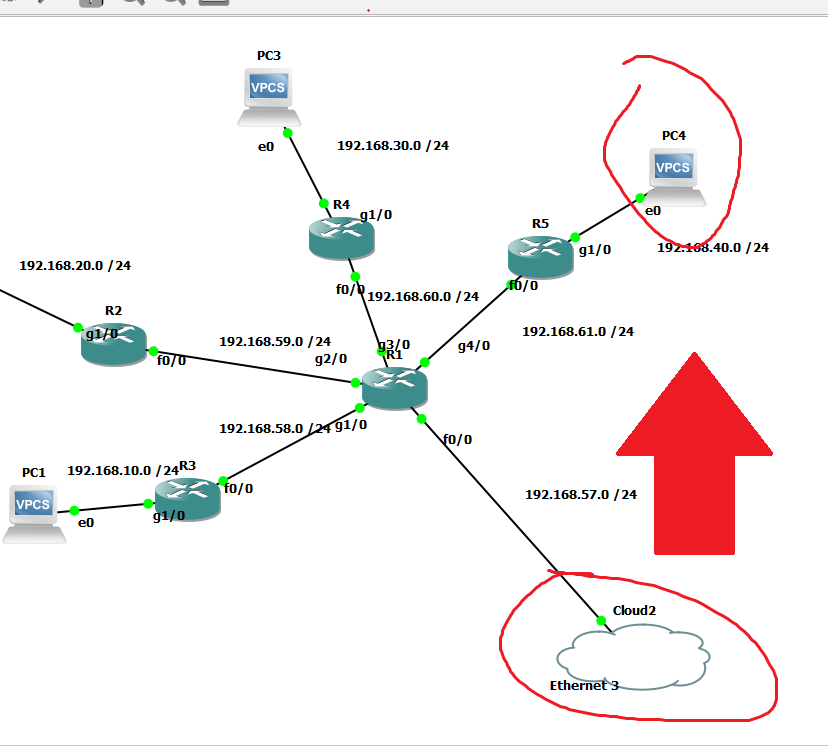
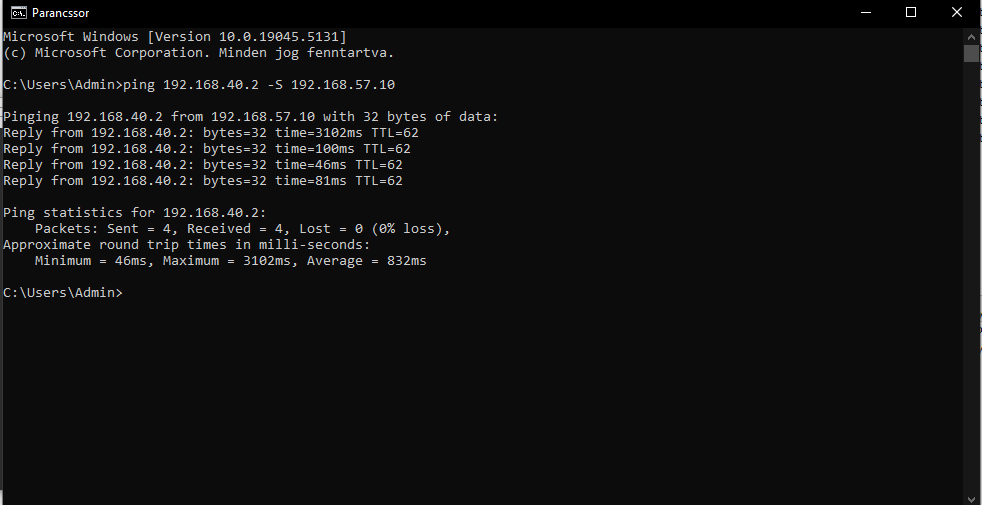
|  |  |  |  |
| --- | --- | --- | --- |
| R3 | F0/0 | R1  G1/0 | 192.168.58.10 /24 |
| G1/0 | PC  e0 | 192.168.10.1 /24 |
| R4 | F0/0 | R1  G3/0 | 192.168.60.10 /24 |
| G1/0 | PC  e0 | 192.168.30.1 /24 |
| R5 | F0/0 | R1  G4/0 | 192.168.61.10 /24 |
| G1/0 | PC  e0 | 192.168.40.1 /24 |
| PC1-PC4 | e0 | R\*(2-5) | DHCP |

## Tesztelés

R2-ről megpingeljük sikeresen az R4-et



Cloud2-ről (saját gép) megpingeljük sikeresen a PC4-et



# Alkalmazás leírás

## Leírás

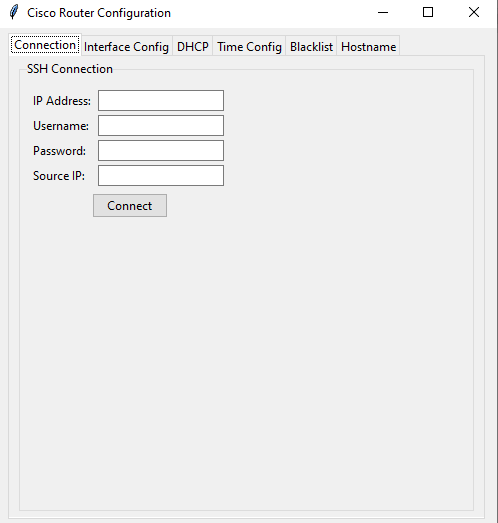
Csapatunk a Cisco router-ekhez írt egy GUI kódot Python kódolásban (Hálózat része: netmiko, GUI része: tkinter). 5 funkicót kellett beleépíteni a programkba, amik nálunk a következőek:

* Interfészek konfigurálása
* DHCP konfigurálás
* Idő beállítása
* Feketelista
* Hosztnév beállítása

## Alkalmazás funkciói

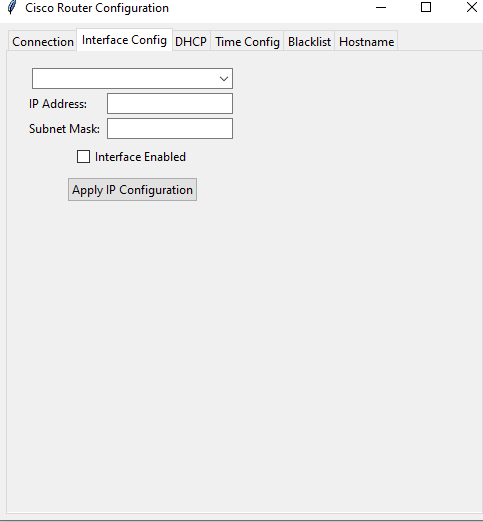
### Connection oldal

Ezen az oldalon lehet csatlakozni a kívánt eszközhöz. Meg kell adni a kívánt eszköz IP-címét, annak SSH felhasználónevét és jelszavát. Továbbá lehetséges megadni egy forrás IP címet, amennyiben problémákba ütközik a kapcsolódás, amiatt, mivel az alapértelmezett útvonal egy másik portra van beállítva, ami ezzel a megoldással kiküszöbölhető. A csatlakozást a connect gombbal lehet elérni.



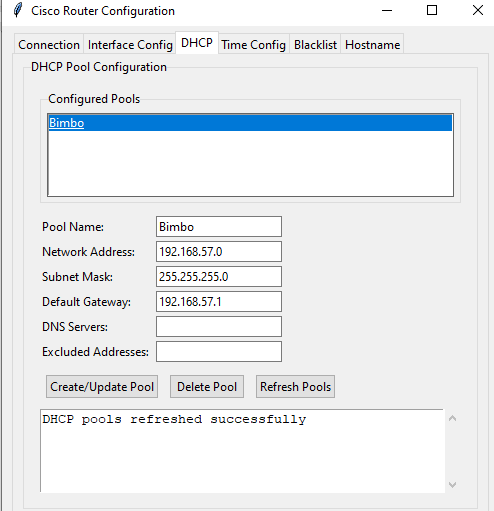
### Interface Config oldal

Ezen az oldalon az interfészeket lehet kiválasztani (ezek dinamikusan frissülnek), majd azoknak lehet megadni ip-címet és alhálózati maszkot, ezeket az Apply IP Configuration gombbal lehet véghez vinni. Továbba található egy checkbox, amivel az interfészt lehet fel-le kapcsolni (Ha pipa akkor be van kapcsolva, ha üres, akkor ki van kapcsolva), ez egyből a megnyomása után frissíti, ezzel együtt változtatja az állapotát.



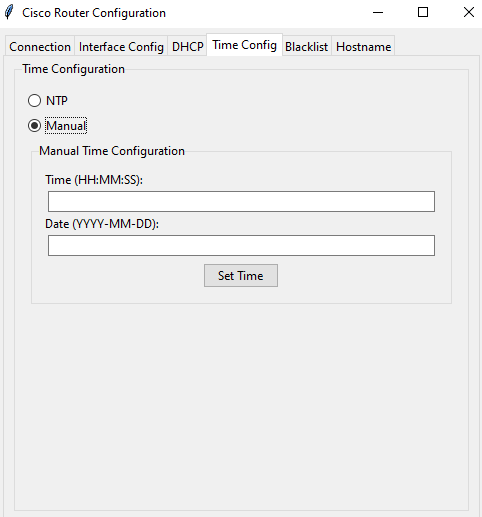
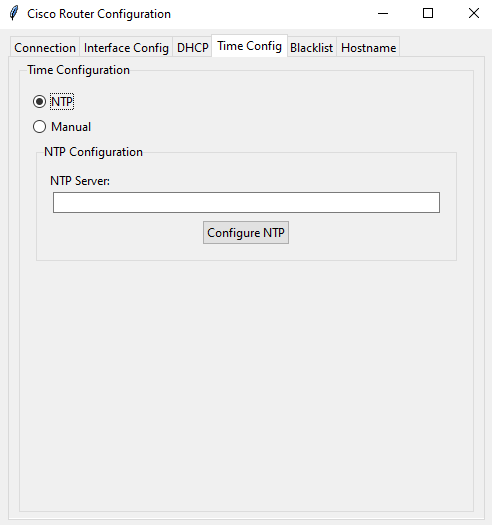
### DHCP beállítás oldal

Ezen az oldalon a DHCP poolokat lehet beállítani a router-eken. Meg lehet adni a nevét, a hálózaticímét, az alapértelmezett átjárót, az alhálózatot, a DNS címet (opcionális) és azokat a címeket, amiket ne osszon ki (opcionális). Továbbá alul 3 gomb látható, ami közül az elsővel létrehozhatjuk/frissíthetjük a DHCP poolt (Create/Update Pool), a másodikkal törölhetjük a kiválasztott poolt, míg a harmadikkal csak szimplán a listára frissíthetünk rá a listára.



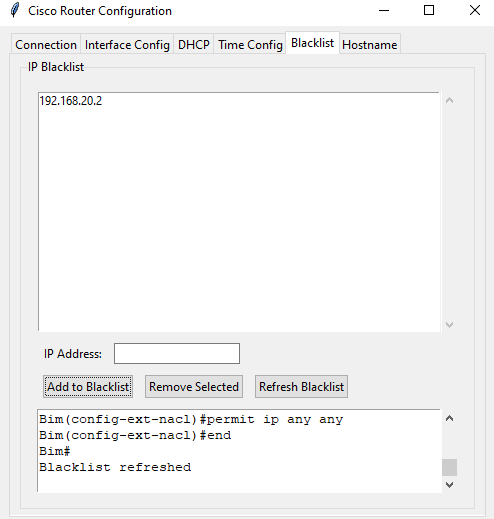
### Time Config oldal

Ezen az oldalon az időt lehet beállítani. Lehetőséget ad a program, hogy NTP szervert írjunk be, vagy pedig, hogy manuálisan állítsunk be időt, illetve dátumot . (NTP-nél Configure NTP, Manuálisnál Set Time gombbal)



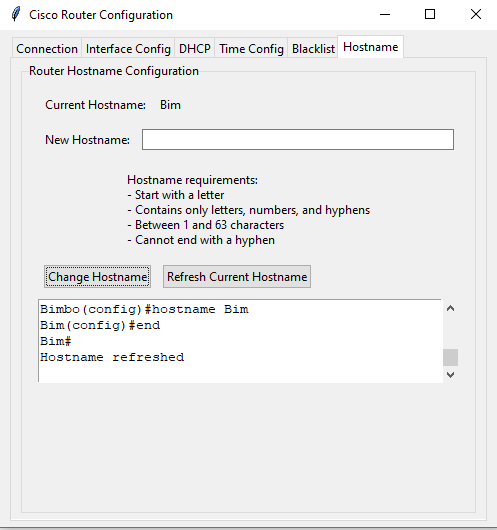
### Blacklist oldal

A Blacklistnek saját magunk írhatunk be IP címeket, majd adhatjuk hozzá az „Add to Blacklist” gombbal. A kiválasztott elemeket a „Remove Selected” gombbal törölhetjük. A listát a „Refresh Blacklist” gombbal frissíthetjük. Megoldásilag egy hozzáférési listét (Acces-list) hozunk létre, ahol az összes megadott címet elutasítjuk (deny) az összes porton.



### Hostname oldal

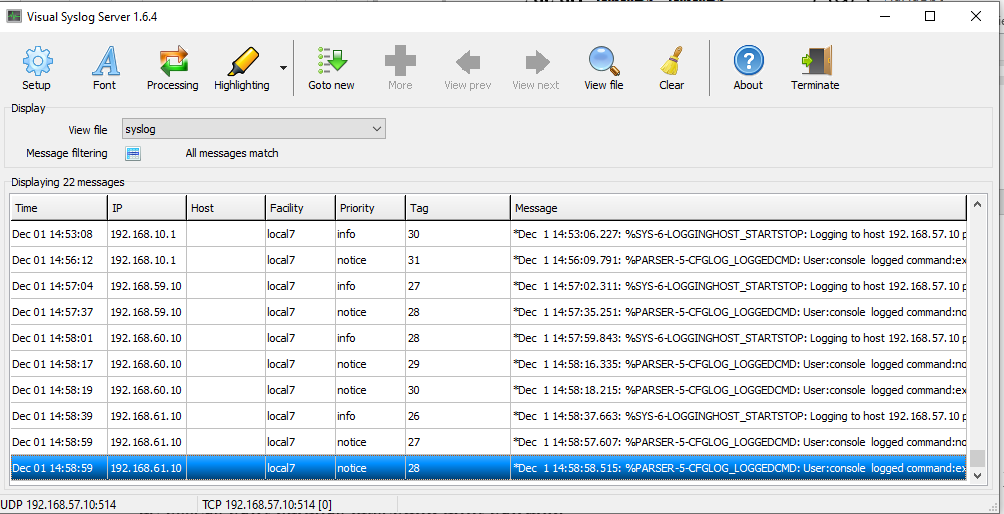
Ezen az oldalon hostnamet tudunk állítani, amit a „New Hostname” mezőbe tudunk beírni és a „Change Hostname” gombbal tudjuk átváltoztatni.



# Syslog szerver

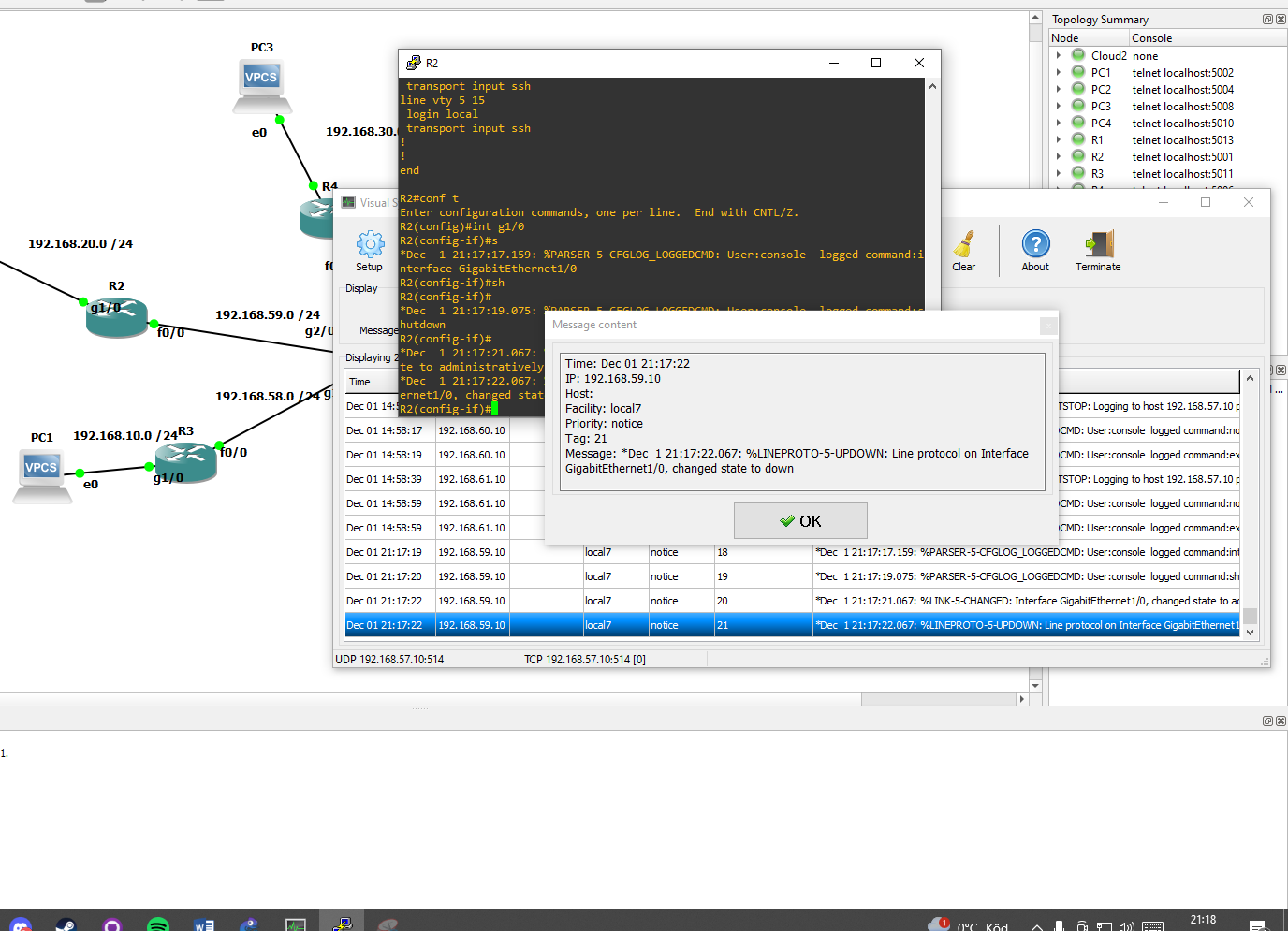
## Leírás

Mint említettük a Visual Syslog Server alkalmazást választottuk. Ez a szerver a 192.168.57.10-es íp címen található, ami az összes üzenetet megkapja mind TCP és mind UDP protokollból. Az alábbi képen is jól látható, hogy a routereken az összes beírt parancs látszódik, amit beírunk.



## Tesztelés

A R2-be beleírunk egy shutdown parancsot a G1/0 portjára 21:17:10-kor. És mint látható a syslog szerver megkapja a változást.

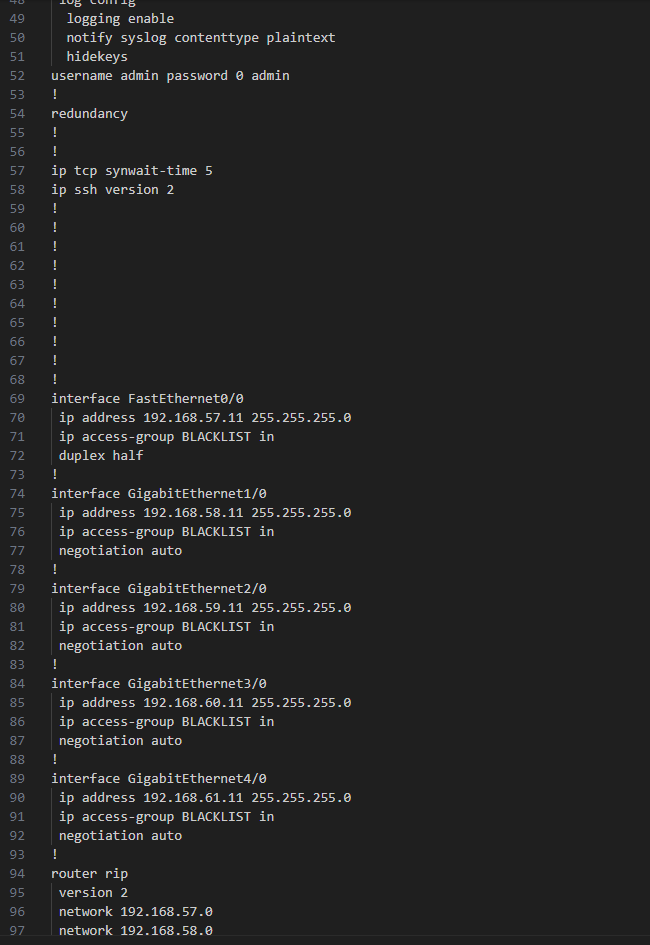


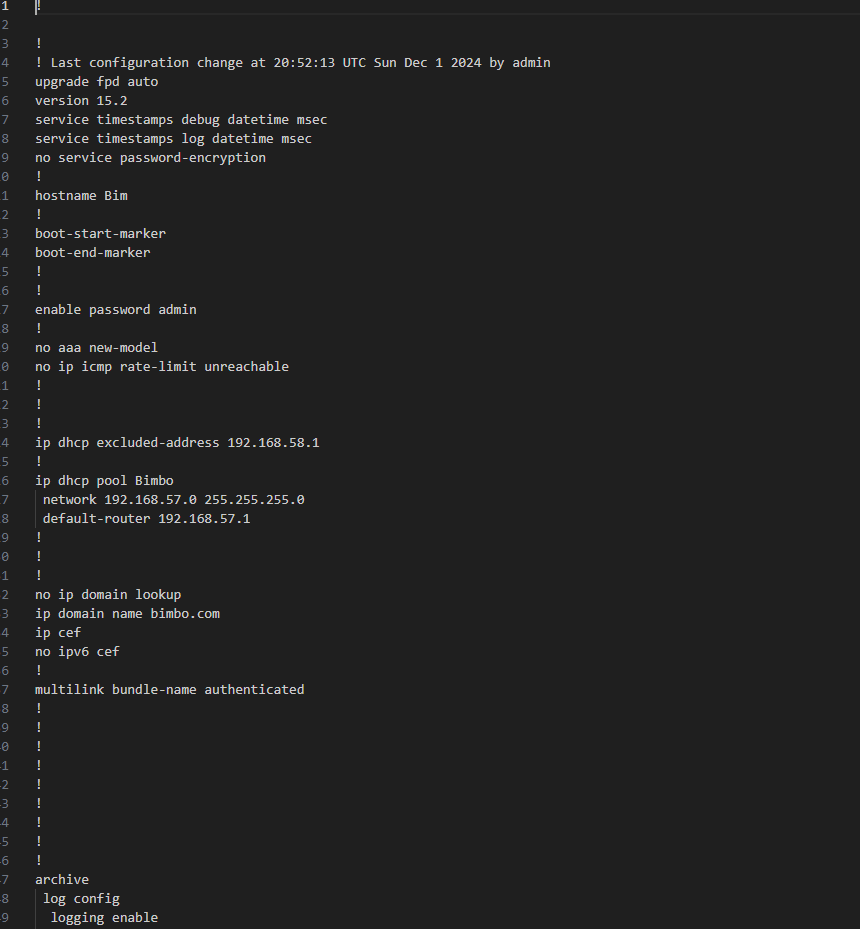
# A projekt nehézségei

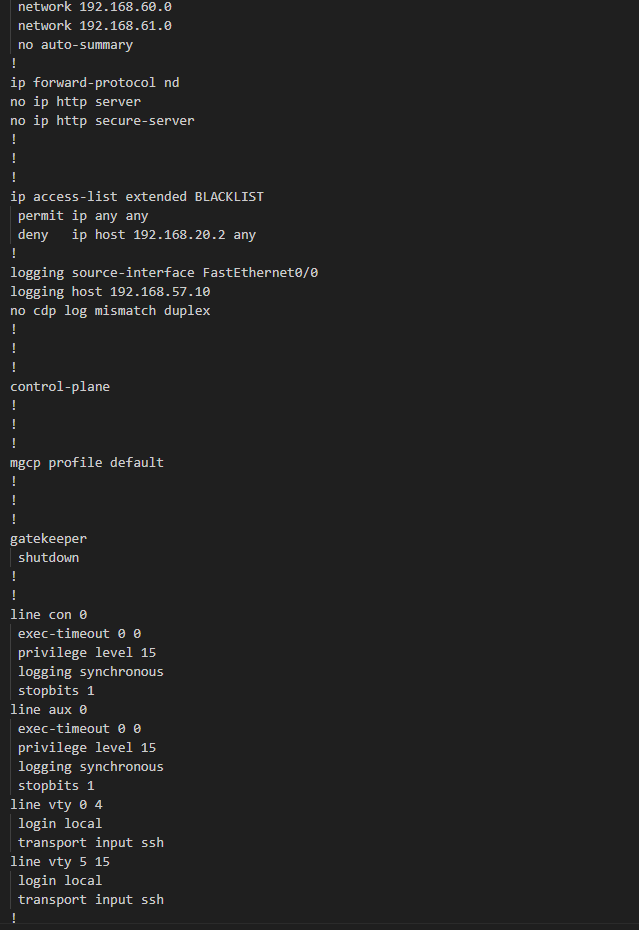
Mindegyikünk foglakozott már hálózattal és egyéb megoldásokkal, azonban a GNS3 mindhármunknak új volt. Pythonnal is mindegyikünk foglalkozott már, azonban egy sokkal kisebb szinten. Először is meg kellett tanulni a GNS3 használatát, aminél az első nehézséget a különböző eszközöknek a fájljainak megszerzése okozta, majd ezek beimportálása a szimulációs programba. A következő probléma, hogy hogyan szeretnénk futtatni a hálózatunkon a Python kódot. Először virtuális géppel próbálkoztunk, majd utána rájöttünk, hogy csak szimplán rá kéne kötni a gépünket a hálózatra. Létrehoztuk az adaptert és… semmi. Végezetül mint kiderült az NPCAP nem volt megfelelő állapotban ezért azt kellett frissíteni. Azonban ez sem működött az összes csapattagunknak, ezért nekik az NPCAP-ot le kellett törölni és helyette WINPCAP-ot kellett telepíteni. A következő nehézség a Python kód megírása volt, mivel tényleg senki sem értett hozzá, se GUI-s szinten, se Netmikos (hálózati megoldás) szinten.

# Hálózati eszközök configjai

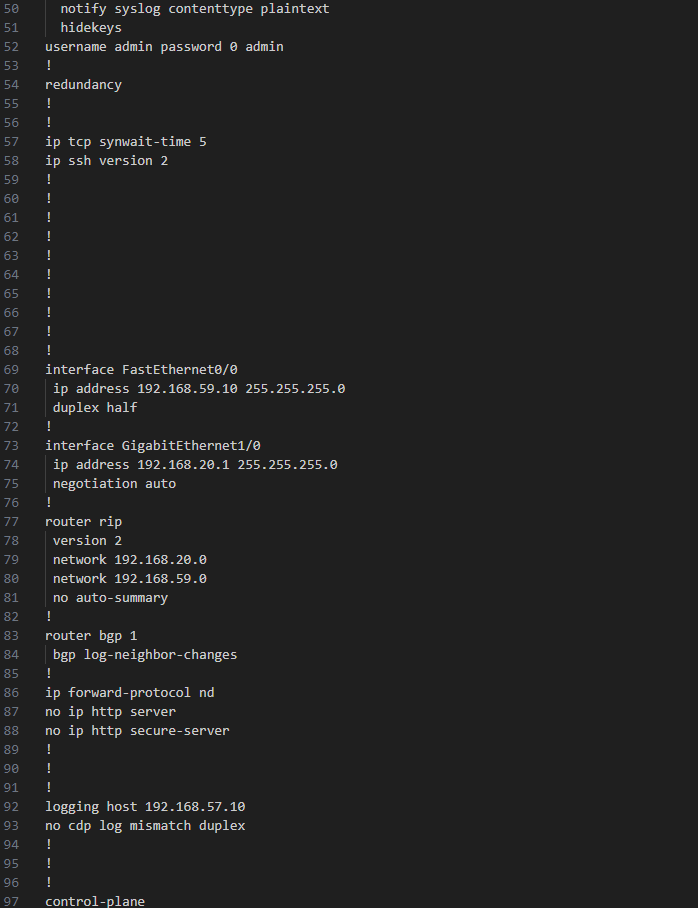
## R1



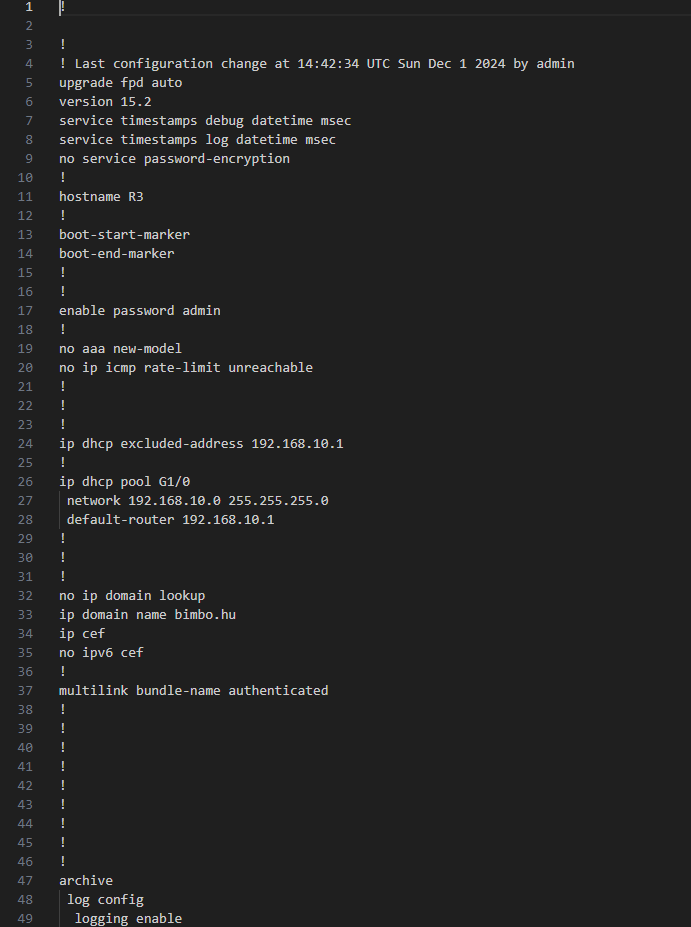
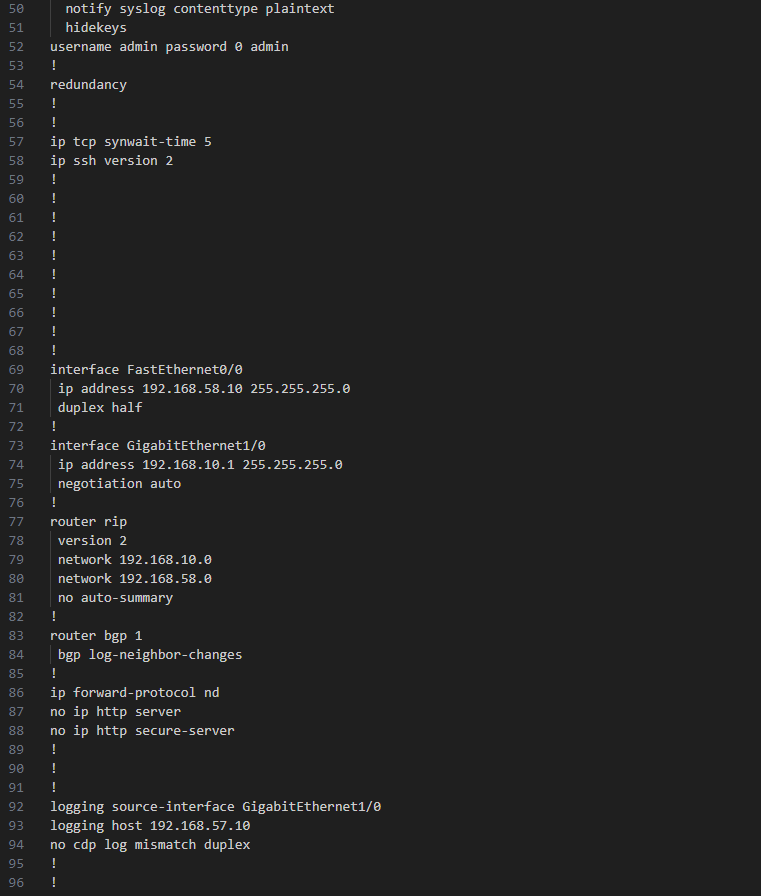
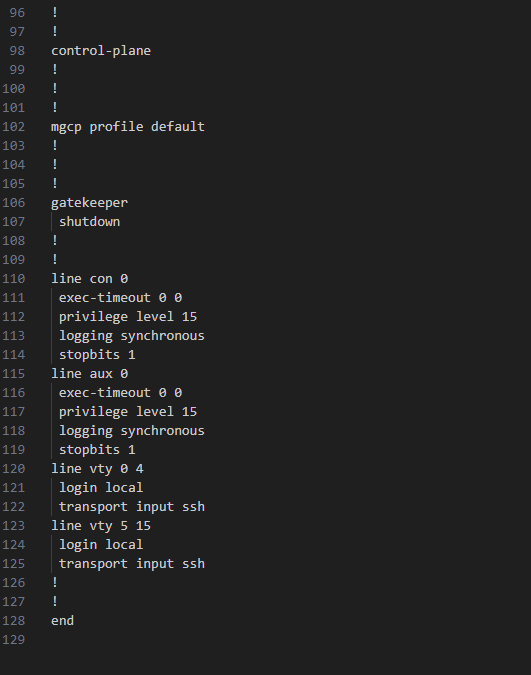




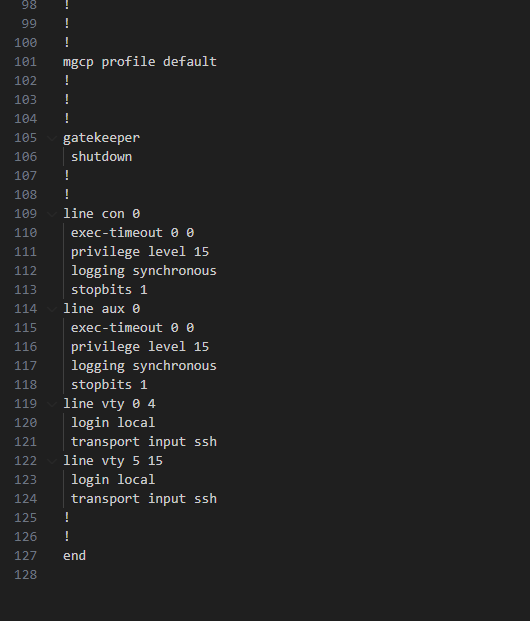
## R2



## R3



## R4



## R5

