Vizsgaremek Dokumentáció

Tesztelés leírás

Külkereskedelmi Technikum

Informatikai rendszer- és alkalmazás-üzemeltető technikus

Tartalom

[Bevezetés 2](#_Toc128750482)

[Szerver szolgáltatások tesztelése 3](#_Toc128750483)

[DHCP (Dynamic Host Control Protocol) 3](#_Toc128750484)

[Címtár (Active Directory) 5](#_Toc128750485)

[DNS (Domain Name Service) 7](#_Toc128750486)

# Bevezetés

Ez a dokumentáció a tesztelés teljes körű leírását tartalmazza. Itt be lesz mutatva az előírások alapján minden olyan elvárás melyet teljesítenünk kellett. Mint a leírás tartalmazta szükséges volt minimum 1-1 Linux és Windows szerver eszköz alkalmazása. Ezen szerverekkel pedig számos szolgáltatást kellett megvalósítani. A tesztelés folyamán a **GNS3** (**Graphical Network Simulator 3**) nevezetű programot használtuk. A szerverek, illetve kliens eszközök virtuális megvalósítására pedig az Oracle által forgalmazott **VirtualBox** programot alkalmaztuk. Ezt a kettőt sikeresen összekapcsoltuk egymással és így megvalósult számunkra a megfelelő környezet a tesztelésre. A szervereken kívül a hálózati eszközökön szereplő redundáns, illetve más hálózati protokoll megoldásokat is bemutatjuk melyek a gördülékeny működés érdekében voltak elengedhetetlenek. Ehhez a feladathoz szükség volt nem kevés erőforrásra melyet igyekeztünk beszerezni annak érdekében, hogy egyáltalán a tesztelés megszülethessen. Tehát térjünk is rá arra, hogy miket is teszteltünk:

Szervereken szereplő szerepkörök:

* DHCP (Dynamic Host Control Protocol)
* Címtár (Active Directory)
* DNS (Domain Name Service)
* Webserver (HTTP/HTTPS)
* Fájl- és nyomtató megosztás
* Automatizált mentés
* Kliens számítógépekre automatizált szoftvertelepítés
* Programozott hálózatkonfiguráció

Hálózati eszközökön szereplő megoldások:

* HSRP (Hot Standby Routing Protocol)
* Port-Security
* VPN (Virtual Private Network)
* IPv6 Tunnel
* Statikus és dinamikus forgalomirányítás
* Statikus és dinamikus címfordítás
* ACL (Access Control List)
* Hardveres tűzfal

Ezek lennének azok, amelyek le lettek tesztelve és a későbbiekben ezeknek a leírása lesz olvasható. Helyenként képek is segítséget nyújtanak a könnyebb és átláthatóbb felfogás érdekében. A tesztelés a teljesség igénye nélkül folyt le. Ez azt jelenti, hogy a szervereken lévő szerepkörök telepítésébe nem mentem bele hiszen ez alapvető annak érdekében, hogy látható legyen a végeredmény. A hálózati eszköznél pedig a parancsokat melyek segítségével megvalósítottuk az elvárt feladványt nem részleteztem, mivel az **ismertető leírásban** szerepel a szerverek konfigurációs állományaival együtt. Azonban minden olyan észrevételre melyet a tesztelés során fedeztünk fel azt leírtam. A tesztelés során számos olyan probléma került felfedezésre melyek kijavítása elengedhetetlen volt annak érdekében, hogy a hálózatunk kifogástalanul működhessen.

A folytatásban előszőr a szervereken működő szerepkörök tesztelése következik. Itt, mint ahogy említettem képes formában is szerepelni fognak számos dolgok melyek segítségével átláthatóbb és nem mellesleg figyelemfelkeltőbb lesz az összkép. Ezek után a hálózati eszközök szereplő szolgáltatásokat és protokollokat fogom tesztelni. A hálózati eszközöket a **Putty** nevezetű konzol program segítségével tudtam monitorozni és konfigurálni.

# Szerver szolgáltatások tesztelése

## DHCP (Dynamic Host Control Protocol)

Minden hálózati eszköz számára elengedhetetlen, hogy IP címmel rendelkezzen mivel enélkül nem képes részt venni a hálózati kommunikációjában. Ezt a lehetőséget kétféleképpen tudjuk orvosolni: statikusan vagy dinamikusan. Az első megoldás megbukik, ha sok kliens számítógépről beszélünk. A második megoldás, hogy ezt a szerepkört rábízzuk egy **szerverre**, aki mindent megoszt a kliens számítógépekkel dinamikus úton. Mi is így jártunk el a hálózataink megtervezésekor. Erre a feladatra a sokak által megbízhatónak tartott **Linux** operációsrendszert alkalmaztunk. Térjünk is rá hogyan valósítottuk meg:

Ehhez a feladathoz a **Linux** rendszereken jól ismert **isc-dhcp-server** csomagot telepítettük mely rendkívül felhasználóbarát beállításokkal egyszerűen és gyorsan megvalósítható egy hálózatban IP címek, illetve vele járó információk kiosztása. Igen ám, de mi a helyzet akkor amikor Vlan interfészekről beszélünk? A tesztelést követően ezen a területen problémába ütköztünk. Az elképzelés az volt, hogy a Vlan 40-ben szereplő **Linux** **szerver** (melynek IP címe 10.30.40.2) majd **irodai számítógépek** (melyek a Vlan 75-ben foglalnak helyet) és a **vezetéknélküli hálózatra csatlakozó eszközök** (pedig a Vlan 66-ban) számára nyújtana dinamikus címkiosztást. Az elképzelés jó volt, de a kivitelezéskor problémákba ütköztünk. Mivel a szerverünk a Vlan 40-ben „lakik” ezáltal a Cisco kapcsolón azt a portot melyhez a szerver csatlakozik be kellett állítanunk, hogy a port a Vlan 40 számára végezze a csomagvezérlés mely annyit jelent, hogy ezen a porton csak a Vlan 40 forgalma közlekedhet. Mert ha ezt nem tettük meg akkor nem tudott kommunikálni a multilayer kapcsolóval. A következő parancs került kiadásra a kapcsolón:

*interface Ethernet1/0*

*switchport mode access*

*switchport access vlan 40*

Ennélfogva azzal a gyanúperrel élhetünk, hogy így hogyan fogjuk tudni kiszolgálni a másik kettő Vlan számára a címkiosztást, ha azon az egy porton csak a Vlan 40 forgalma közlekedhet? Hiába konfiguráltuk be az **isc-dhcp-server**-t megfelelően és hoztuk létre a megfelelő alinterfészeket a Vlan 66 és 75 eszközei nem jutottak IP címhez. A konfiguráció melyet beírtunk a dhcpd.conf állományba:

*option domain-name "kan.lan";*

*option domain-name-servers 10.30.40.3, 192.168.122.1;*

*default-lease-time 600;*

*max-lease-time 7200;*

*# Vlan 75 Office*

*subnet 10.30.75.0 netmask 255.255.255.0 {*

*range 10.30.75.10 10.30.75.254;*

*option routers 10.30.75.1;*

*}*

*# Vlan 66 Wi-Fi*

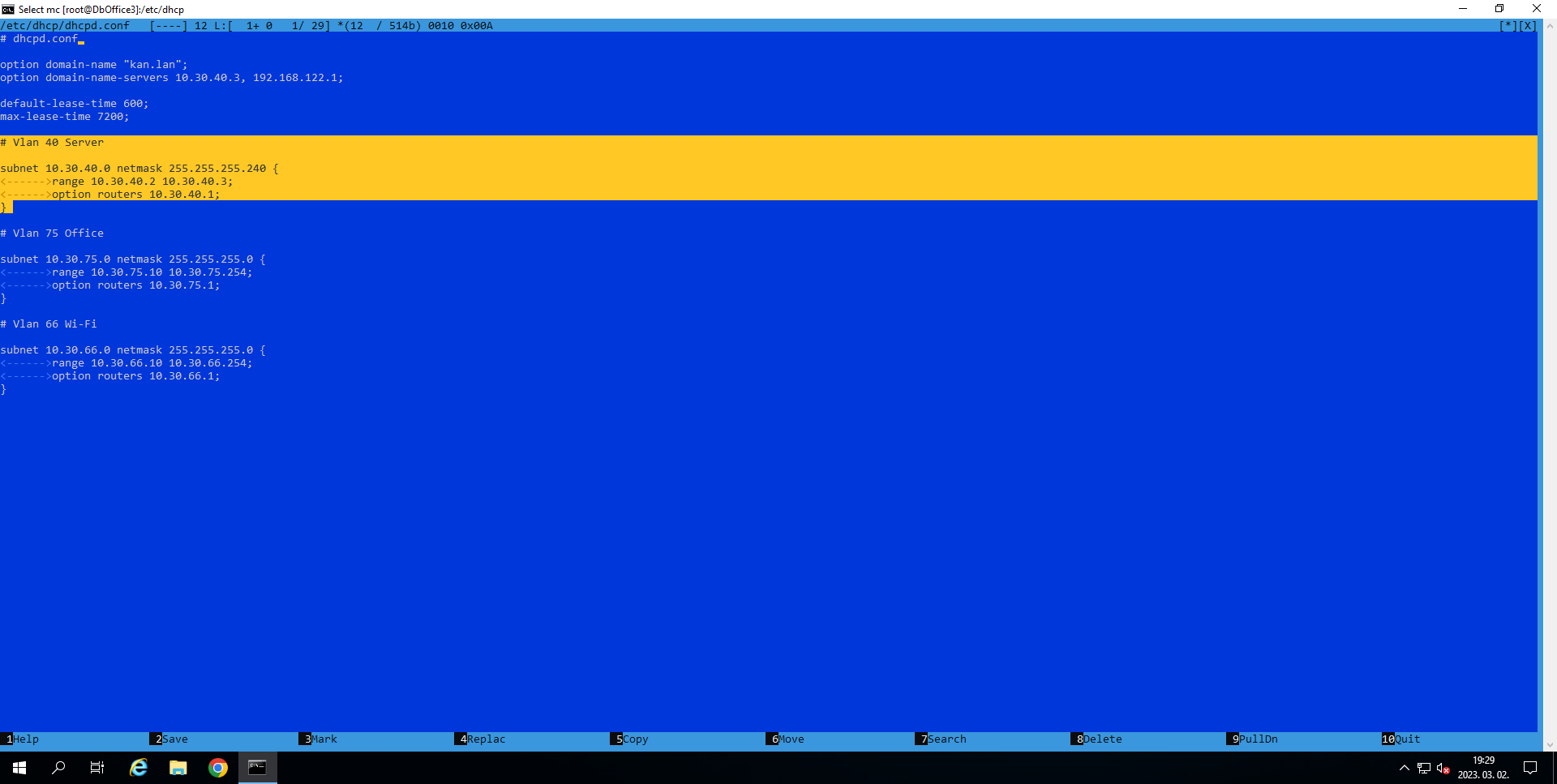
*subnet 10.30.66.0 netmask 255.255.255.0 {*

*range 10.30.66.10 10.30.66.254;*

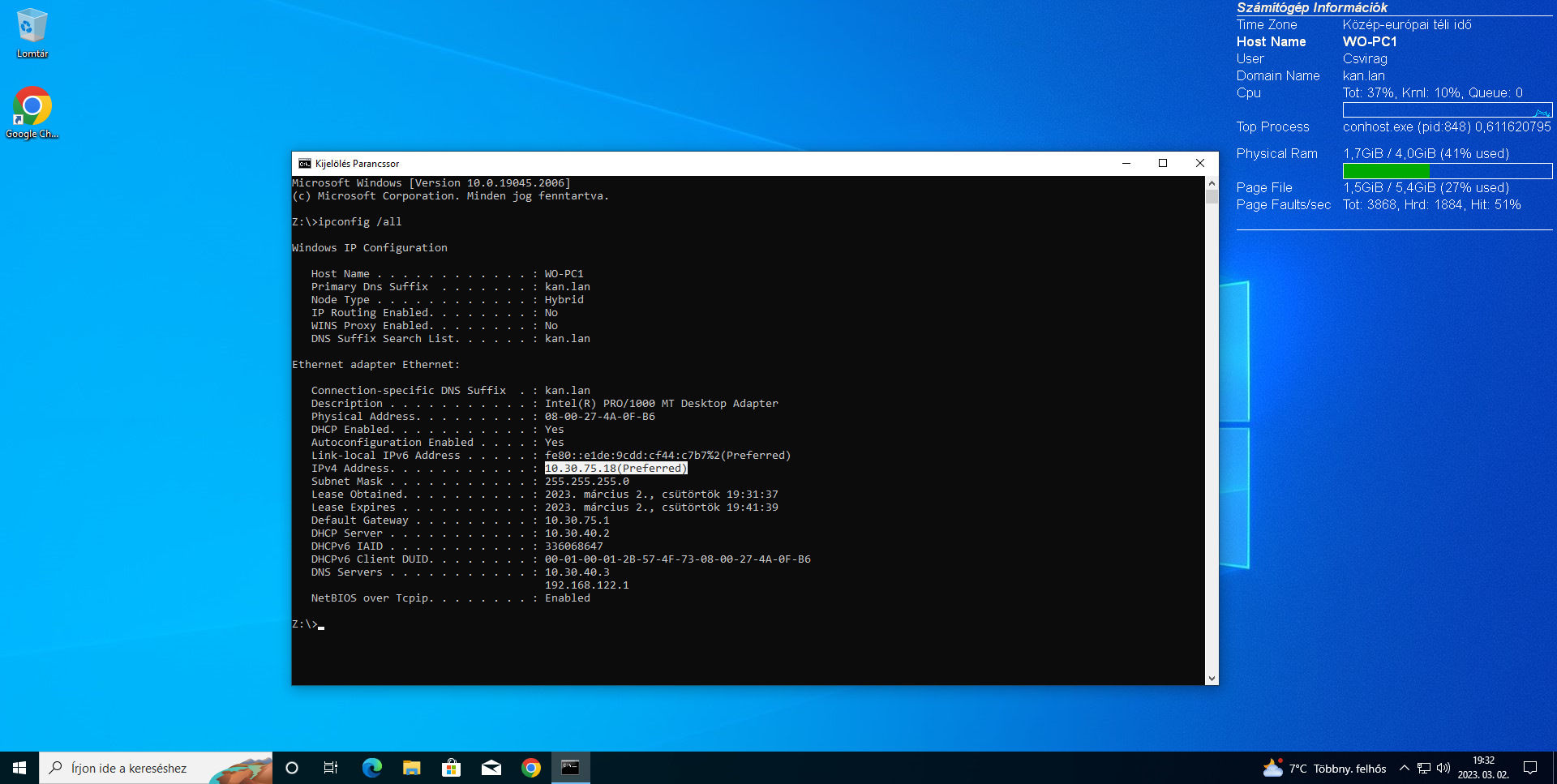
*option routers 10.30.66.1;*

*}*

Elsőkörben utána jártunk, hogy milyen lehetőségekkel tudnánk megoldani ezt a problémát. Először beállítottuk a multilayer kapcsolónkon a Vlan interfészek számára, hogy merre is keressék a DHCP kiszolgálót. A következő parancsot adtuk ki: *ip helper-address 10.30.40.2.* Ez azonban még nem oldotta meg a problémát, de közelebb jutottunk a megoldáshoz. Ezt követően átvizsgáltuk megint a konfigurációnkat a Linux szerveren, hogy hátha mégis valami elírás következtében nem működőképes a szolgáltatás, de nem találatunk hibát. Végül segítségkérés után kiderült miért is nem működött. Ezt egy képpel illusztrálnám:



A Linux szerverre a Windows Server 2019-ről csatlakoztam rá SSH protokoll segítségével parancssort alkalmazva. Amint látható a kijelölt rész hiánya volt az, amely miatt nem működhetett a másik kettő Vlan kiszolgálása. Miután ezt a sort beírtuk ettől fogva a mind a kettő Vlan számára sikeresen nyújt dinamikus IP kiosztási szolgáltatást. Azonban ez a sor azt is jelenti, hogy a 40-es hálózatban szereplő eszközök kaphatnak dinamikusan címet – de ezek az eszközök melyek ebben a *range-*ben szerepelnekstatikus IP címmel rendelkeznek. Érdekesség, hogy ez a jelenség nem fordult elő a Windows Server 2019-en. Példának okáért itt van az egyik kliens, amely megkapta a szervertől az IP címet és a vele járó adatokat:

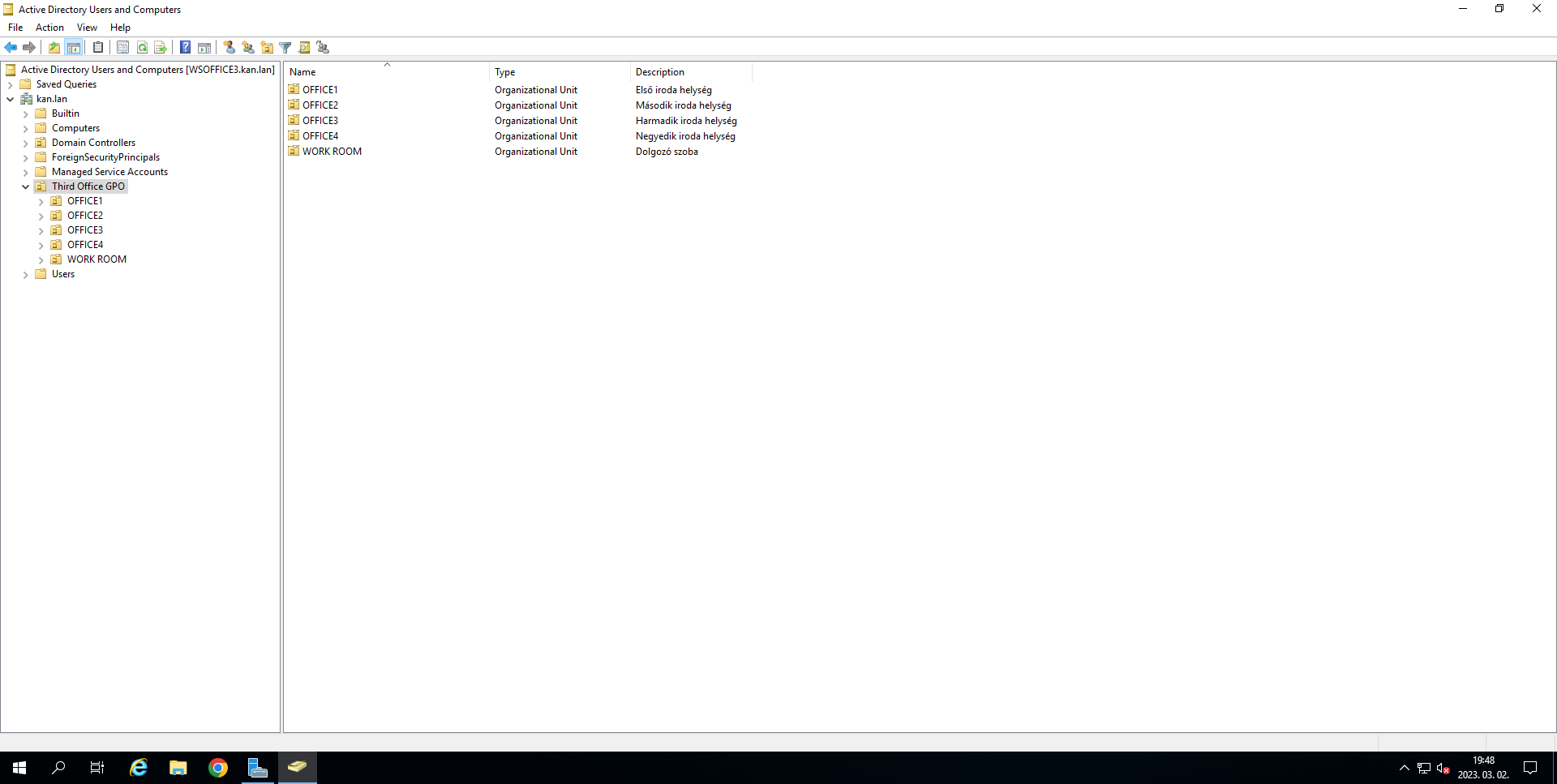


Amint látható ott szerepel a DHCP szerver IP címe, illetve az alapértelmezett átjáró, amely a multilayer kapcsoló. Ezen felül látható még, hogy az eszköz megkapta a DNS kiszolgáló IP címét (elsődlegesként a Windows Server 2019 és másodikként pedig a **GNS3 szerver** által nyújtott NAT felhő IP címe) és a tartomány nevét a kan.lan-t. Tehát végül a DHCP szerver beállítása azon a pár soron múlt melyet nem írtunk be. Végezetül elkönyvelhetjük, hogy korántsem olyan egyszerű az, hogy mit írjunk a konfigurációs állományba. *Ez a példa a harmadik telephelyen szereplő Linux szerverről készült.*

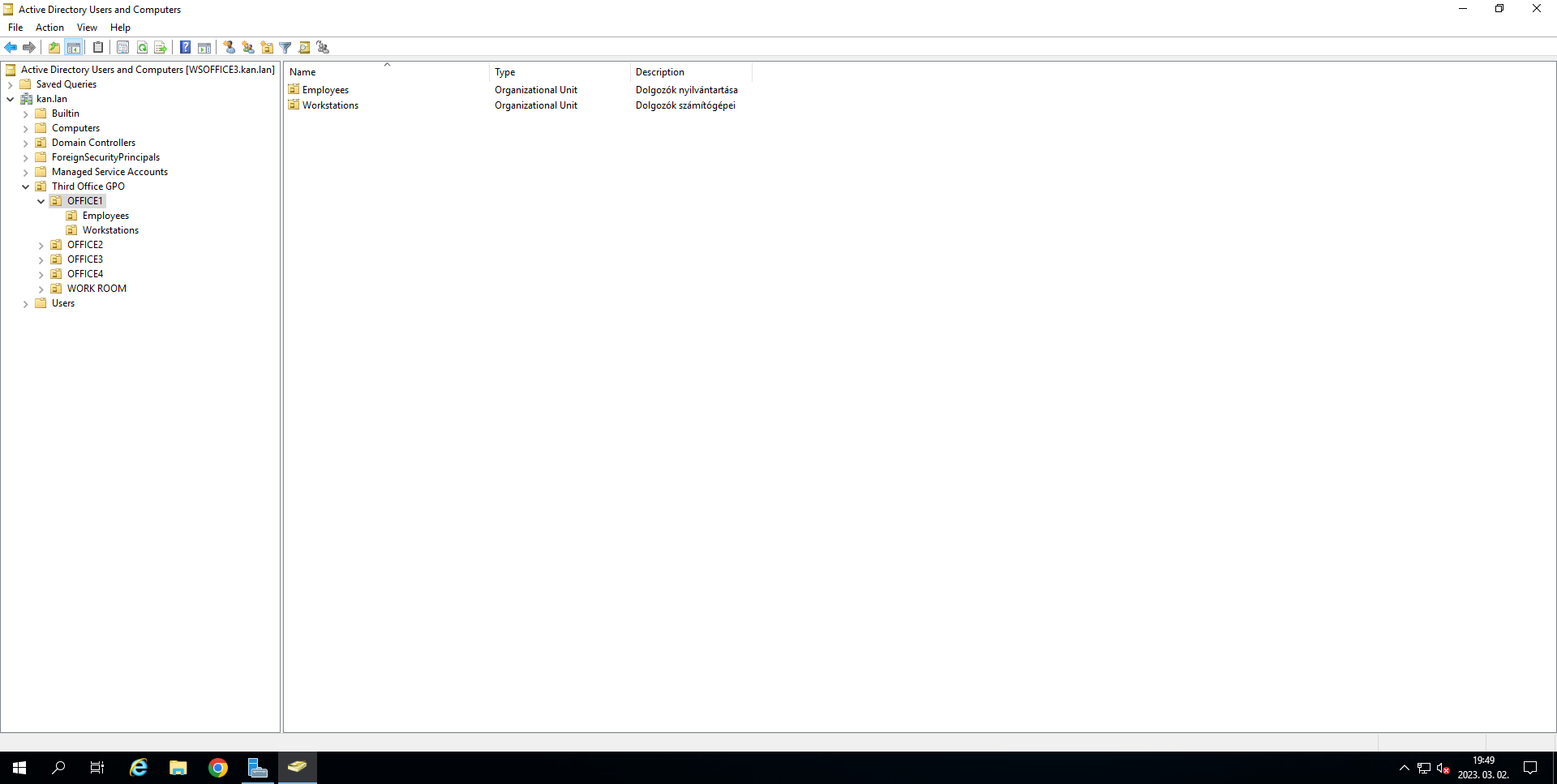
## Címtár (Active Directory)

Ezt a feladványt a **Microsoft** által megvalósított **Active Directory** (továbbiakban **AD**) szolgáltatással valósítottuk meg mivel egyszerűbb és átláthatóbb a kezelése, mint más konkurens megoldásnak például a **Linux** operációs rendszeren belül lévő **Samba**-nak. Nem mellesleg a **Microsoft** által megvalósított címtár sokkal elterjedtebb, sőt a kliens számítógépek legjava is valamely **Windows** operációs rendszer verzióját futtatja, ennélfogva jobb választásnak bizonyult egy olyan platform, amely közös gyártón osztozik. Na de lássunk is hozzá a teszteléshez:

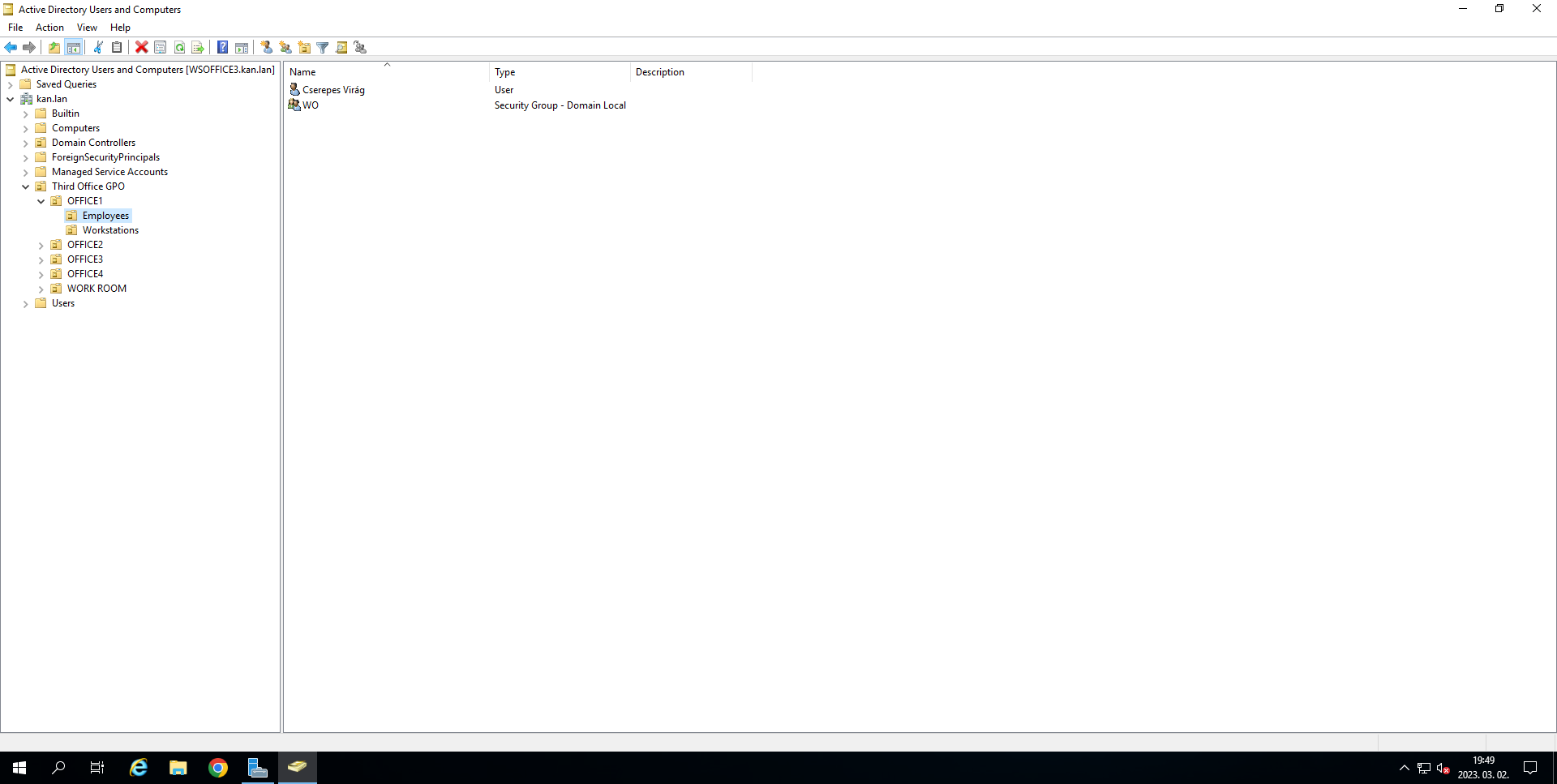
Tehát mint említettem a címtár megvalósítását **Windows Server 2019**-en valósítottuk meg. Mindenekelőtt hangsúlyt fektettünk arra is, hogy milyen SMB protokollt használjunk. A **Server Message Block** elengedhetetlen az **AD**-ba tartozó klienseknek, hogy kommunikálhassanak a szerverrel. Az **SMBv1** protokoll letiltásával kezdtük mivel a biztonság az első faktor egy ilyen szolgáltatás esetén. Az **SMBv1** protokoll nagyon elavult és biztonságát tekintve alul marad újabb társaitól (az **SMBv2** és **SMBv3**-tól például). PowerShell segítségével letiltottuk és engedélyeztük az **SMBv2**-őt. Ezt megtettük a kliens számítógépeken is annak érdekében, hogy elkerüljük a kommunikációs problémákat. Arra is felfigyeltünk, hogy az újabb Windows csomagokban (például a 21H2-ben) már alapértelmezetten tiltva van ez a protokoll, de jobbnak tartottuk ezt saját kezűleg is megvalósítani. Ezen felül az IPv6-ot is letiltottuk mivel nem kívántuk használni egy olyan környezetben, ahol csak IPv4 szerepel. A szerverünk, mint az előző DHCP teszt során látható volt a **10.30.40.3**-as statikus IP címmel rendelkezik. Az **AD** szolgáltatás telepítésével együtt a **DNS** szerepkör is települ mivel függőségként tekinthető az **AD** számára. Miután megtörtént a telepítés felvettük az „új erdőt” amely neve a **kan.lan**. A telepítés végét követően megkezdtük a könyvtárt szerkezet kiépítését a telepehelyek számára. A későbbiekben látott fotókon a **harmadik telepehelyen** lévő címtár szervert fogjuk megtekinteni és annak könyvtár szerkezetét.



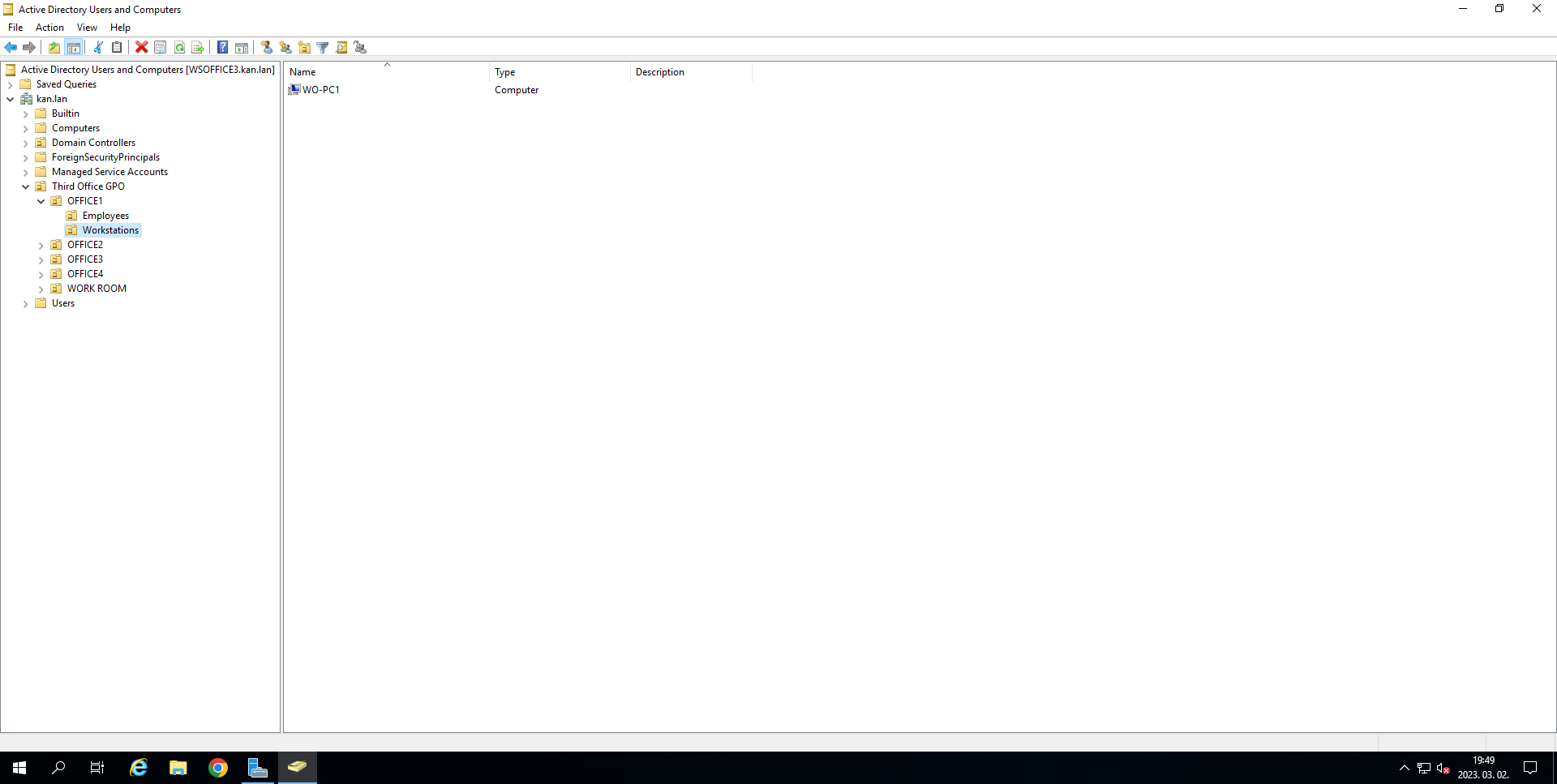
Amint látható a képen létrehoztunk egy **Third Office GPO** nevezetű szervezeti egységet, amely alatt a többi hozzá tartozó szervezeti egység szerepel. Mivel a harmadik telephelyen **4 darab** **irodahelyiséget** és **egy dolgozó szobát** foglal magában ezért szét szedtük külön szervezeti egységekre annak érekében, hogy az ezekben lévő számítógépeket és felhasználókat külön lehessen kezelni, ha esteleg alkalmazás telepítés vagy akár fájl és nyomtató megosztást szeretnénk megvalósítani. Példának okáért a harmadik telephelyünkön minden helységben külön nyomtatók találhatók ezért volt értelme külön bontani őket egymástól.



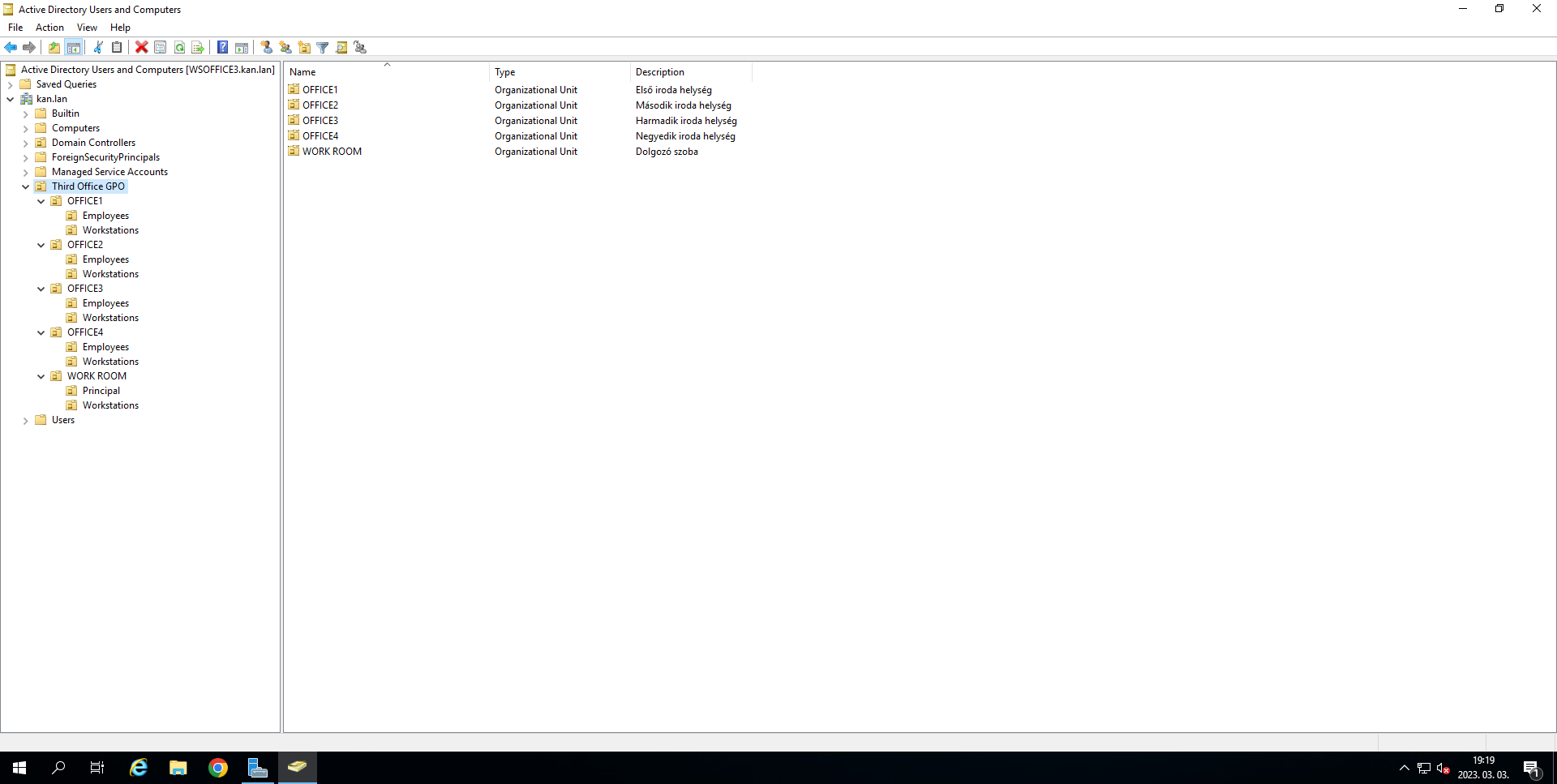
Amint a következő képen látható, hogy az **OFFICE1** iroda számára létrehozott szervezeti egység mit tartalmaz. Ez a szervezeti egység tartalmaz egy *Employees* és egy *Workstation* elnevezésű szervezeti egységet. Elsőben az irodában dolgozók profiljait tartjuk nyilván a másodikban pedig az adott irodához tartozó számítógépeket. Eme séma mentén van a többi szervezeti egység is megvalósítva kivéve a *WORK ROOM* esetében mivel ott *Employees* helyett *Principal* szervezeti egység helyezkedik el.



Belépve az *Employees* szervezeti egységbe láthatóvá válik az egyik tartományi felhasználó (Cserepes Virág) melyet a tesztelés érdekében hoztunk létre. Azonban szerepel ott egy *WO* nevezetű biztonsági csoport, amely azért hoztuk létre, hogy a dolgozókat biztonsági csoportok szerint el tudjuk szeparálni egymástól. Erre azért volt szükség mivel a mappamegosztásoknál fontos, hogy kinek milyen jogot vagy engedélyt adunk meg az adott könyvtár vonatkozásában és ezt egyszerűbb úgy megvalósítani, hogy csoportba rendezzük a dolgozókat és egy tömbként kezeljük őket, mintsem egyesével felvesszük őket mivel az elég hosszadalmas és ezért releváns megoldás.



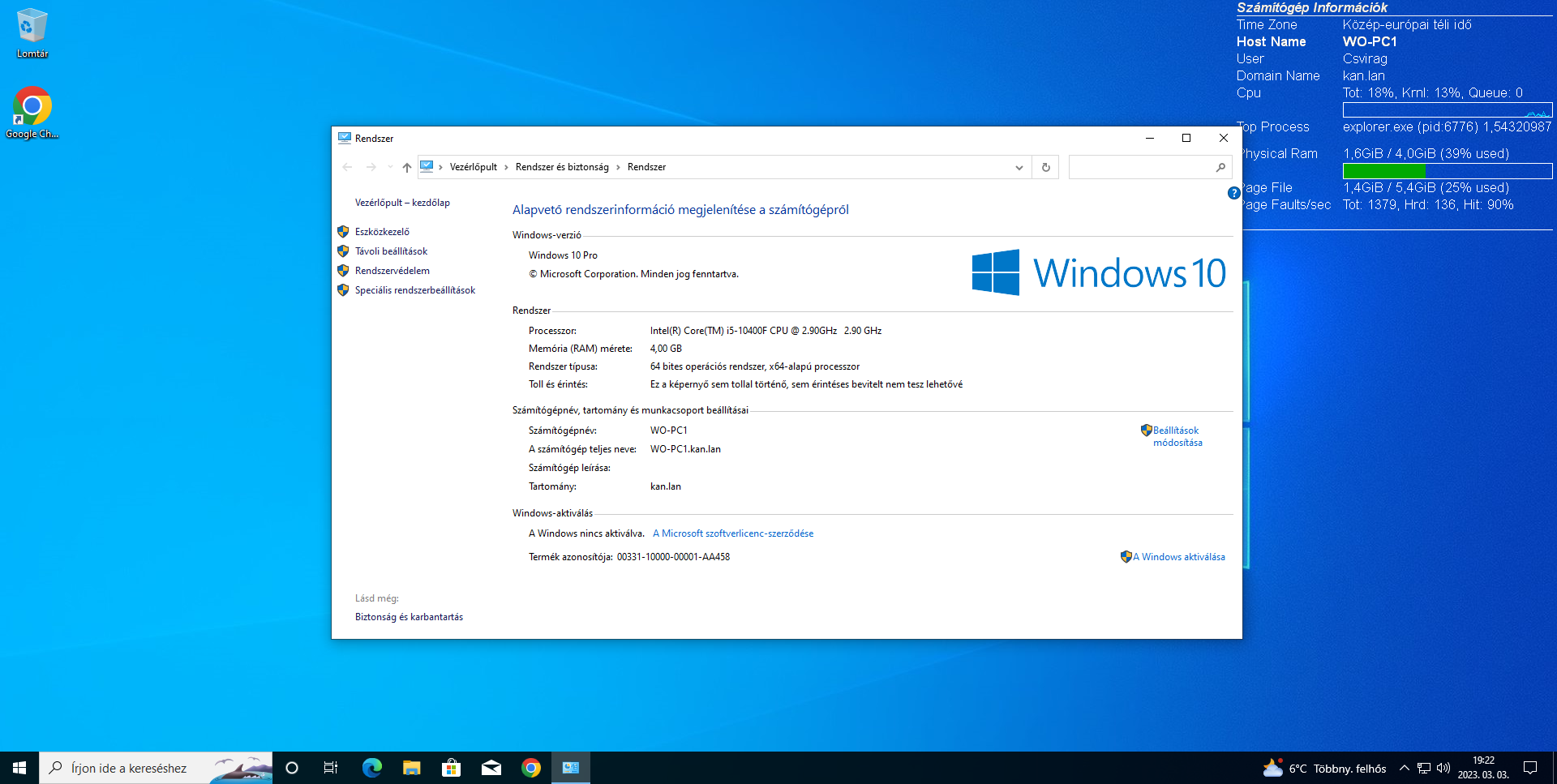
Átlépve a *Workstation* megnevezésű szervezeti egységbe láthatóvá vált a számítógép melyet felvettünk a tartományunkba. A gépeket úgy neveztük el, hogy tudjuk hová is tartoznak felhasználói szinten. A „WO” elnevezést a *Workers*-t rövidíti és ezt az eszközt az előbb mutatott tartományi felhasználó (Cserepes Virág) használja. Azonban e mentén például a HR-PC1 az a Human Resources és CEO-PC1 a Principal számára nyújt részvételt a tartományba. Végül itt van teljes könyvtárszerkezet kinyitva:



Ezek után jöhetett a tesztelés a kliens számítógépen. Elsősorban parancssort használva egyszerűbb parancsok kiadásával leteszteltük, hogy az eszközünk egyáltalán kapcsolatban e áll a Windows szerverünkkel. Amint a képen látható elsőként *ping* parancsot alkalmaztunk annak érdekében, hogy lássuk válaszol e nekünk a szerver. IP cím és tartománynév alapján is megkíséreltük a kapcsolatfelvételt. Azonban nem kaptunk választ. Ez azért történhetett mert az **AD** szerver tűzfalán nem engedélyeztük az ICMP csomagot ki és be történő áramlását. Miután ezt beállítottuk sikeresen tudtuk *pingelni* a szervert. Ezt követően még alkalmaztuk az *nslookup* nevezetű felderítő parancsot, amely tartománynév megadását követően kiírja a hozzá tartozó szerver elérhetőségét, azaz IP címét. Miután ezzel is sikerrel jártunk következhetett a tartományba való beléptetés. A következő képen látható, hogy a kliens a tartomány része:



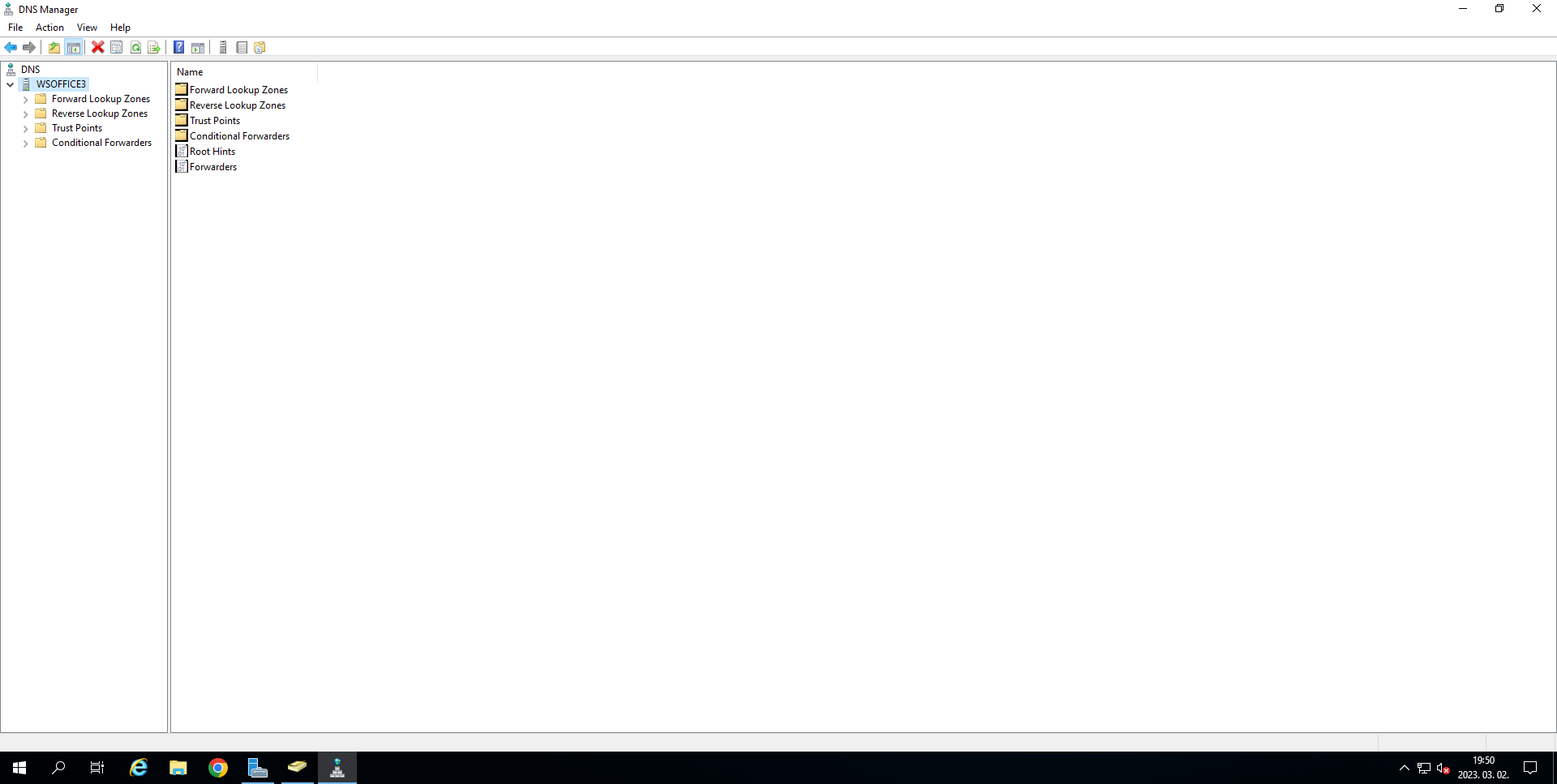
A bejelentkezési képernyőn találkozhatunk az előbbiekből jól ismert felhasználóval.



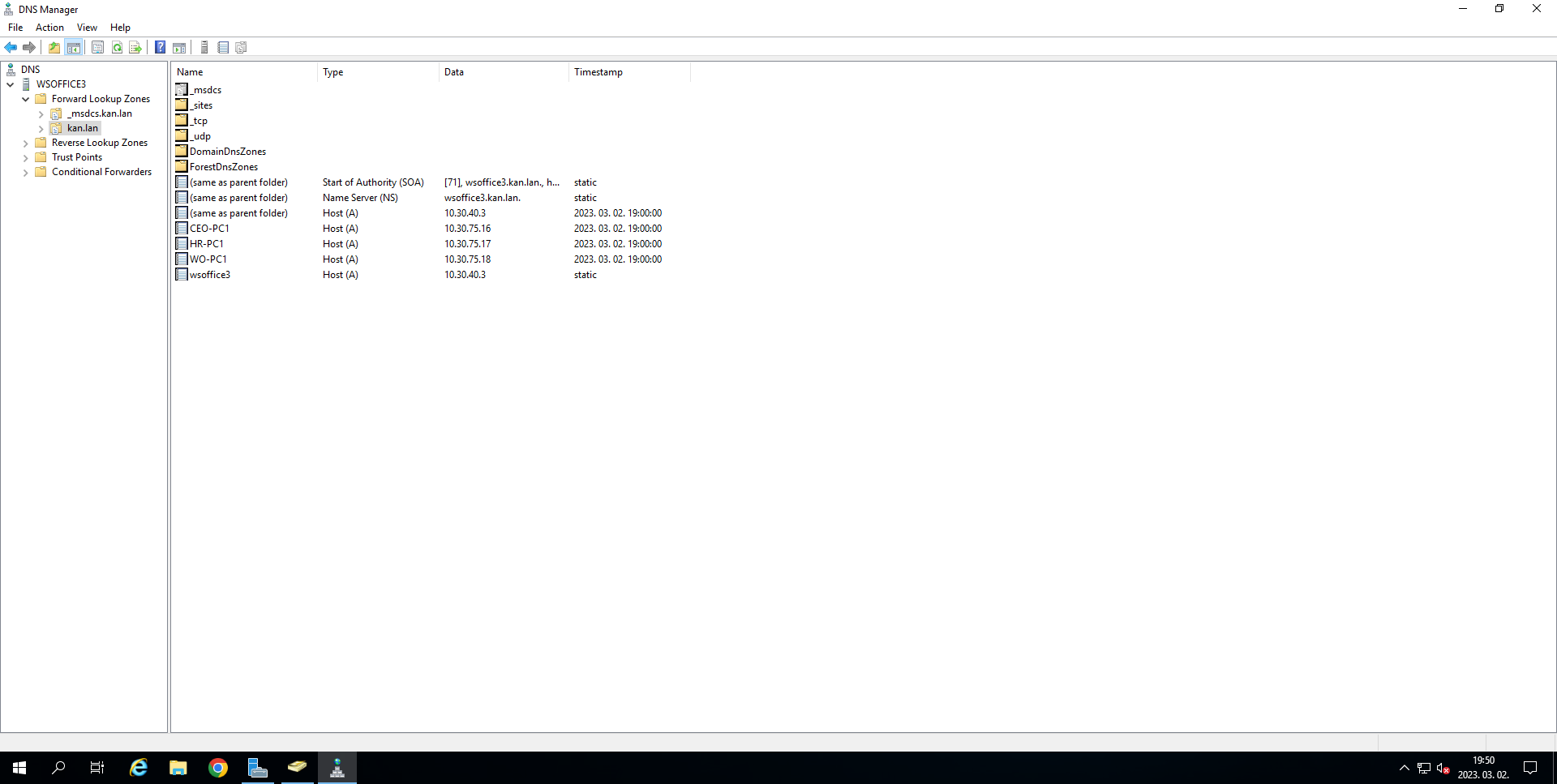
Bejelentkezést követően a vezérlőpultban le ellenőrizhető a tartomány megléte és amint látható ott szerepel a tartománynevünk a *kan.lan*. Innentől kezdve tudjuk kezelni az eszköz minden egyes beállítását távolról. *Ez a példa a harmadik telephelyen szereplő Windows Server 2019 szerverről készült.*

## DNS (Domain Name Service)

Mint az előző teszt demonstrálta a **DNS** szolgáltatás elengedhetetlen a címtár működése érdekében ennélfogva mindenféleképpen szükséges a használata. Telepítése a címtárral együtt történt a mi esetünkben, azonban külön is telepíthető. Nem mellesleg a **DNS** szolgáltatás annak érdekében is fontos mivel a kliensek számára szükséges egy olyan kiszolgáló, amely a névfeloldás teszi számukra lehetővé. Röviden a névfeloldás azért szükséges, hogy kiváltsuk az IP címek megjegyzését és inkább szöveg formátumban keressünk rá weboldalakra. Mint a **DHCP** szolgáltatás tesztelésekor látott parancssorban ott szerepelt a **DNS** kiszolgálók között a mi szerverünk a **10.30.40.3** IP címmel. A kliensek tehát ehhez a szerverhez fognak fordulni a névfeloldás megvalósítása érdekében. A következő képen a **DNS** szolgáltatás beállításai láthatóak:



Amint látható szerverünk neve **WSOFFICE3** elnevezéssel rendelkezik. Kinyitva őt láthatjuk a **DNS** könyvtár szerkezetét. Itt tulajdonképpen a **DNS** szolgáltatás által alapértelmezetten meghatározott könyvtár szerkezetet figyelhetjük meg.



Tovább lépve a **Forward Lookup Zones** könyvtárba és ott a mi általunk létrehozott tartománynak a DNS szolgáltatás létrehozott számos bejegyzéseket. Fentről lefelé haladva látható a **SOA** (Start of Authority) rekord, amely a zóna kezdő rekordot jelenti, enélkül nincs **DNS** szolgáltatás mivel olyan információkat tartalmaz melyek elengedhetetlen a működés érdekében. Ezután következik a **NS** (Name Server) rekord mely megnevezi a szerverünket az elsődleges szervernek. Ezt követően jönnek az **A** (Address) rekordok melyek az IPv4-es címeket tartják nyilván. Amint látható a három kliens számítógép is szerepel ebben a felsorolásban.