KKK hálózati dokumentációja

Készítette:

Csete Ádám

Nagy Barnabás

Téringer Gergő

Tartalomjegyzék

[1. Projekt ismertetése 1](#_Toc194859585)

[1.1 A projekt célja 1](#_Toc194859586)

[1.2 A Kihagyhatatlan Kincsek Kiadó 1](#_Toc194859587)

[1.3 A cég felépítése 1](#_Toc194859588)

[1.3.1 Központ – Budapest 1](#_Toc194859589)

[1.3.2 Fiók – Debrecen 2](#_Toc194859590)

[1.3.3 Fiók – Szeged 3](#_Toc194859591)

[1.4 A cég igénye 3](#_Toc194859592)

[1.5 Csapatunk 4](#_Toc194859593)

[1.6 Csapatmunka 5](#_Toc194859594)

[2. Hálózat 6](#_Toc194859595)

[2.1 LAN 6](#_Toc194859596)

[2.1.1 Logikai topológia 6](#_Toc194859597)

[2.1.2 Fizikai topológia 9](#_Toc194859598)

[2.1.2.1 Épületek paraméterei 9](#_Toc194859599)

[2.1.2.2 Topológia 10](#_Toc194859600)

[2.1.3 VLAN-ok 15](#_Toc194859601)

[2.1.4 Kapcsolók 16](#_Toc194859602)

[2.1.5 IPv4 és IPv6 címzés 17](#_Toc194859603)

[2.1.6 VTP 27](#_Toc194859604)

[2.1.7 STP 27](#_Toc194859605)

[2.1.8 Második rétegbeli redundancia 28](#_Toc194859606)

[2.1.8.1 LACP 28](#_Toc194859607)

[2.1.9 Harmadik rétegbeli redundancia 29](#_Toc194859608)

[2.1.9.1 Redundáns kapcsolatok 29](#_Toc194859609)

[2.1.9.2 HSRP 29](#_Toc194859610)

[2.1.9.3 Harmadik rétegbeli EtherChannel 30](#_Toc194859611)

[2.1.10 OSPFv2, OSPFv3 30](#_Toc194859612)

[2.1.11 Biztonság 31](#_Toc194859613)

[2.1.11.1 Fizikai biztonság 31](#_Toc194859614)

[2.1.11.2 Hitelesítés 31](#_Toc194859615)

[2.1.11.3 Belépés próbálkozások száma 32](#_Toc194859616)

[2.1.11.4 Inaktív időkorlát 32](#_Toc194859617)

[2.1.11.5 Webes felület 32](#_Toc194859618)

[2.1.11.6 SSH 33](#_Toc194859619)

[2.1.11.7 Management VLAN 34](#_Toc194859620)

[2.1.11.8 VLAN 35](#_Toc194859621)

[2.1.11.9 Nem használt portok 35](#_Toc194859622)

[2.1.11.10 Portbiztonság 35](#_Toc194859623)

[2.1.11.11 Trunk portok 35](#_Toc194859624)

[2.1.11.12 DHCP Snooping 36](#_Toc194859625)

[2.1.11.13 Dinamikus ARP-ellenőrzés 36](#_Toc194859626)

[2.1.11.14 BPDU Guard 36](#_Toc194859627)

[2.1.11.15 Root guard 36](#_Toc194859628)

[2.1.11.16 CDP letiltása 37](#_Toc194859629)

[2.1.11.17 ACL-ek 37](#_Toc194859630)

[2.1.11.18 Unicast Reverse Path Forwarding (uRPF) 40](#_Toc194859631)

[2.1.11.19 WLAN 40](#_Toc194859632)

[2.1.11.20 OSPFv2 és OSPFv3 hitelesítés 40](#_Toc194859633)

[2.1.11.21 EIGRP hitelesítés 42](#_Toc194859634)

[2.1.11.22 BGP hitelesítés 42](#_Toc194859635)

[2.1.11.23 Passzív interfész 43](#_Toc194859636)

[2.1.11.24 HSRP hitelesítés 43](#_Toc194859637)

[2.1.11.25 Direkt kulcsok 44](#_Toc194859638)

[2.1.11.26 Hardveres tűzfal: ASA 44](#_Toc194859639)

[2.1.12 WIFI 46](#_Toc194859640)

[2.1.13 Bannerek 47](#_Toc194859641)

[2.2 WAN 47](#_Toc194859642)

[2.2.1 Internetszolgáltató 47](#_Toc194859643)

[2.2.2 BGP, statikus forgalomirányítás 48](#_Toc194859644)

[2.2.3 PPPoE 49](#_Toc194859645)

[2.2.4 PAT, SNAT 50](#_Toc194859646)

[2.2.5 DMVPN 51](#_Toc194859647)

[2.2.6 EIGRP 51](#_Toc194859648)

[3. Szerverek 52](#_Toc194859649)

[3.1 Windows szerverek 52](#_Toc194859650)

[3.1.1 ADDS, RODC, subdomain 52](#_Toc194859651)

[3.1.2 Group Policy 53](#_Toc194859652)

[3.1.3 DNS 55](#_Toc194859653)

[3.1.4 WDS 56](#_Toc194859654)

[3.1.5 Nyomtató szerver 57](#_Toc194859655)

[3.1.6 RADIUS 58](#_Toc194859656)

[3.2 Linux szerverek 59](#_Toc194859657)

[3.2.1 KKK-LIN 59](#_Toc194859658)

[3.2.1.1 Ansible 60](#_Toc194859659)

[3.2.1.2 LibreNMS 60](#_Toc194859660)

[3.2.1.3 Webszerver 62](#_Toc194859661)

[3.2.1.4 Email 65](#_Toc194859662)

[3.2.1.5 Rsyslog szerver 71](#_Toc194859663)

[3.2.1.6 Tűzfal 73](#_Toc194859664)

[3.2.1.7 Webmail 75](#_Toc194859665)

[3.2.2 KKK-VPN 77](#_Toc194859666)

[3.2.2.1 Wireguard 79](#_Toc194859667)

[3.2.2.2 Rsyslog kliens 81](#_Toc194859668)

[3.3 Felhő 81](#_Toc194859669)

[3.3.1 Amazon Web Services 81](#_Toc194859670)

[3.3.2 A felhő 82](#_Toc194859671)

[3.3.3 Webkiszolgáló 82](#_Toc194859672)

[3.3.4 Fájlkiszolgáló szolgáltatás 84](#_Toc194859673)

[3.3.5 Adatbázis szolgáltatás 84](#_Toc194859674)

[3.4 Automatizált mentés 86](#_Toc194859675)

[3.4.1 KKK-BACKUP 86](#_Toc194859676)

[3.4.1.1 RAID 5 87](#_Toc194859677)

[3.4.1.2 Samba, Rsync 88](#_Toc194859678)

[3.4.2 Automatizált mentés: Linux szerverek 88](#_Toc194859679)

[3.4.3 Automatizált mentés: Windows szerverek 90](#_Toc194859680)

[4.Automatizáció 91](#_Toc194859681)

[4.1 Ansible kódok 92](#_Toc194859682)

[4.1.1 ADDS 92](#_Toc194859683)

[4.1.2 DNS 94](#_Toc194859684)

[4.1.3 DHCP 98](#_Toc194859685)

[4.1.4 ADDFS 101](#_Toc194859686)

[4.1.5 FSRM 105](#_Toc194859687)

[4.2 PowerShell 108](#_Toc194859688)

[5. Árkalkuláció 108](#_Toc194859689)

[6. Tesztelés 111](#_Toc194859690)

[6.1 Hálózat 111](#_Toc194859691)

[6.1.1 Traceroute 111](#_Toc194859692)

[6.1.2 VTP 111](#_Toc194859693)

[6.1.3 LACP 112](#_Toc194859694)

[6.1.4 HSRP 113](#_Toc194859695)

[6.1.5 BGP 113](#_Toc194859696)

[6.1.6 SSH 114](#_Toc194859697)

[6.1.7 PPPoE 114](#_Toc194859698)

[6.1.8 DMVPN, OSPF, EIGRP 115](#_Toc194859699)

[6.2 Szerverek 117](#_Toc194859700)

[6.2.1 Windows szerverek 117](#_Toc194859701)

[6.2.1.1 DFS 117](#_Toc194859702)

[6.2.1.2 FSRM 118](#_Toc194859703)

[6.2.1.3 RADIUS, WiFi 118](#_Toc194859704)

[6.2.1.4 WDS, DHCP 119](#_Toc194859705)

[6.2.1.5 Kliens tartományba csatlakoztatása, DNS 120](#_Toc194859706)

[6.2.1.6 AD DS bejelentkezés 120](#_Toc194859707)

[6.2.1.7 Group policy 121](#_Toc194859708)

[6.2.2 Linux szerverek 122](#_Toc194859709)

[6.2.2.1 LibreNMS 122](#_Toc194859710)

[6.2.2.2Webszerver 123](#_Toc194859711)

[6.2.2.3 EMAIL 125](#_Toc194859712)

[6.2.2.4 RSYSLOG 128](#_Toc194859713)

[6.2.2.5 VPN 129](#_Toc194859714)

[6.2.2.6 RSYSLOG kliens 130](#_Toc194859715)

[6.2.3 Felhő 131](#_Toc194859716)

[7. Összegzés 132](#_Toc194859717)

[8. Hivatkozások 132](#_Toc194859718)

[8.1 PowerShell scriptek 132](#_Toc194859719)

[8.1.1 Központ 132](#_Toc194859720)

[8.1.1.1 OU-kat és csoportokat létrehozó script 132](#_Toc194859721)

[8.1.1.2 Felhasználókat létrehozó script 133](#_Toc194859722)

[8.1.2 Debrecen 134](#_Toc194859723)

[8.1.2.1 OU-kat és csoportokat létrehozó script 134](#_Toc194859724)

[8.1.2.2 Felhasználókat létrehozó script 135](#_Toc194859725)

[8.1.3 Szeged 137](#_Toc194859726)

[8.1.3.1 OU-kat és csoportokat létrehozó script 137](#_Toc194859727)

[8.1.3.2 Felhasználókat létrehozó script 138](#_Toc194859728)

[8.2 Ansible kódok 140](#_Toc194859729)

[8.2.1 Addfs.yaml playbook 140](#_Toc194859730)

[8.2.2 Adds.yaml playbook 142](#_Toc194859731)

[8.2.3 Dhcp.yaml playbook 144](#_Toc194859732)

[8.2.4 Dns.yaml playbook 146](#_Toc194859733)

[8.2.5 Fsrm.yaml playbook 148](#_Toc194859734)

[8.2.6 Inventory.yaml playbook 149](#_Toc194859735)

1. Projekt ismertetése

1.1 A projekt célja

A projekt célja, hogy csapatunk elkészítse a Kihagyhatatlan Kincsek Kiadó zenei kiadó számára a cég teljes hálózatát. A projekt keretében meg kell terveznünk és ki kell alakítanunk a központi és a fiókirodák hálózatát és a közöttük lévő kapcsolódásokat. Emellett figyelembe kell venni a biztonsági intézkedéseket, hogy a zenei anyagok védelme és az elérhetősége megfelelően biztosított legyen. A célunk az, hogy egy stabil és megbízható hálózatot hozzunk létre, amely támogatja a cég napi működését, és lehetővé teszi a jövőbeni bővítéseket és innovációkat.

1.2 A Kihagyhatatlan Kincsek Kiadó

A Kihagyhatatlan Kincsek Kiadó egy zenei kiadó, amely egy új cég. A kiadó magyar zenei élet egyik meghatározó szereplője szeretne lenni. A cég célja, hogy támogassa a haza zenei tehetségeket, valamint elősegítse a zenei műfajok sokszínűségét. A kiadó különböző zenei stílusok széles spektrumát képviseli, beleértve a popot, rockot, jazz-t és klasszikus zenét.

Elkötelezett amellett, hogy lehetőséget biztosítson a feltörekvő művészek számára, miközben innovatív marketing- és promóciós megoldásokat kínál. A Kihagyhatatlan Kincsek Kiadó célja, hogy a legjobb zenei anyagokat juttassa el a közönséghez, valamint támogassa a zenei közösséget különféle projektek és események révén.

1.3 A cég felépítése

A cég központja Budapesten található, a fiókirodái pedig Debrecenben és Szegeden.

1.3.1 Központ – Budapest

A Kihagyhatatlan Kincsek Kiadó központja Budapesten található, ahol a cég főbb zenei tevékenységei zajlanak. A kiadói részleg keretében a szerződéskezelők felelősek a szerződések kezeléséért és a jogi dokumentumok nyomon követéséért. Emellett a digitális tartalomkezelő gondoskodik a zenei tartalmak online elérhetőségéről és menedzseléséről, míg zenei elemzők dolgoznak azon, hogy a legfrissebb zenei trendek figyelembevételével támogassák a kiadási döntéseket. A kiadói asszisztens pedig segíti a kiadási folyamatokat és elvégzi az adminisztratív feladatokat.

A marketing és PR osztály feladata, hogy a cég arculatát kialakítsa és a zenei kampányokat lebonyolítsa. A digitális marketing szakemberek online kampányokat terveznek és valósítanak meg különböző platformokon, beleértve a közösségi médiát és a hirdetéseket. A webfejlesztő a cég weboldalának karbantartásáért és frissítéséért felel, míg a grafikai tervezők a kiadványok borítóit és promóciós anyagait készítik. A szociális média menedzser kezeli a cég közösségi média platformjait, interakcióba lép a követőkkel, míg a PR szakértők kapcsolatot tartanak a sajtóval és a közönséggel.

A zenei produkció területén a hangmérnök a hangfelvételek készítésével és szerkesztésével foglalkozik professzionális szoftverek segítségével. A zenei producerek a zenei projektek digitális irányításáért felelnek, míg a zenei szerkesztők a zenei anyagok szerkesztését végzik. A zenei asszisztensek segítenek a stúdióban a felvételek előkészítésében és lebonyolításában.

Az adminisztrációs osztály pénzügyi elemzője pénzügyi kimutatásokat készít és elemzéseket végez, míg a HR menedzser a munkavállalók nyilvántartásait vezeti és a toborzási folyamatokat irányít. A titkárok adminisztratív támogatást nyújtanak a vezetőség számára, és az IT szakember felelős a számítógépes rendszerek és szoftverek működéséért és karbantartásáért.

A vezetői osztály magában foglalja a kiadói, a marketing és PR, a zenei produkciós és az adminisztrációs osztály vezetőit. Ezen szakemberek együttműködése biztosítja a cég tevékenységének hatékony irányítását és zökkenőmentes működését.

1.3.2 Fiók – Debrecen

A debreceni fiók a tehetségkutatásra specializálódott, és célja, hogy felfedezze és támogassa a helyi művészeket, valamint megvalósítsa a zenei közösség által megálmodott projekteket. A tehetségkutató szakember a helyi zenészek nyilvántartásáért felel, és szervezi a helyi zenészek előadásait. A zenei mentor támogatja a feltörekvő zenészeket, és tanácsokat ad a fejlődésükhöz.

A kiadói szolgáltatások területén a zenei marketing asszisztens kampányokat támogat a helyi zenekarok számára, míg a piackutató piaci elemzéseket és trendek kutatását végzi.

A debreceni fiókvezető koordinálja a fiók tevékenységeit, biztosítja a csapat hatékony együttműködését, és felügyeli a tehetségkutatás, zenei mentorálás, valamint a kiadói szolgáltatások projektjeit. Emellett kapcsolatot tart a helyi zenei közösséggel és stratégiai döntéseket hoz a fiók célkitűzéseinek elérése érdekében.

1.3.3 Fiók – Szeged

A szegedi fiók az eseményszervezésre és a helyi rendezvények lebonyolítására összpontosít. A csapat célja, hogy minél több zenei eseményt hozzon létre, amelyek bemutatják a helyi tehetségeket és erősítik a zenei közösséget. Az eseményszervező tervezi és koordinálja az eseményeket. A logisztikai szakember a rendezvényekhez szükséges eszközök és erőforrások tervezéséért és nyilvántartásáért felel, míg a színpadi technikus gondoskodik a technikai háttérről az események során, beleértve a világítást és a hangosítást.

A közönségkapcsolati menedzser digitális platformokon tartja a kapcsolatot a közönséggel, például e-mailek és közösségi média segítségével. A hírlevél szerkesztő készíti a cég hírlevelét, amely informálja a közönséget az újdonságokról és eseményekről. A visszajelzés-elemző gyűjti és elemzi a közönségi visszajelzéseket az események után, hogy javítsa a jövőbeli rendezvényeket.

A szegedi fiókvezető felel a fiók tevékenységeinek irányításáért, az események zökkenőmentes lebonyolításáért és a csapat munkájának összehangolásáért. Emellett biztosítja, hogy a rendezvények stratégiai célokat szolgáljanak, és erősítsék a helyi zenei közösséget.

1.4 A cég igénye

A Kihagyhatatlan Kincsek Kiadó egy dinamikusan fejlődő zenei kiadó. A cég célja, hogy a zenei iparban kiemelkedő szerepet töltsön be, és ennek érdekében fontos, hogy a digitális jelenlétüket is megfelelően támogassák.

A cég számára elengedhetetlen egy korszerű hálózat kialakítása, amely összeköti a központot és a fiókirodákat. Az alábbi speciális igények merültek fel a cég részéről, amelyeket a csapatunktól mindenképpen elvárnak a hálózat kialakítása során:

* **Zavartalan hálózati kapcsolatok**: A központ és a fiókirodák közötti stabil és gyors internetkapcsolat kialakítása alapvető fontosságú. Ez lehetővé teszi a munkatársak számára, hogy valós időben osszanak meg zenei fájlokat és projektinformációkat, így a zenei produkciók gördülékenyebbé válnak.
* **Hálózati monitoring:** A cég elvárja, hogy az IT csapat a kezdeti időszakban folyamatosan figyelje a hálózat teljesítményét és biztonságát. A hálózati forgalom elemzése segít az esetleges problémák korai észlelésében, valamint a megelőző intézkedések megtételében.
* **VPN hozzáférés:** A fiókirodák munkatársai számára biztosítani kell a távoli elérést a központi hálózathoz. A VPN rendszer kiépítése segít abban, hogy a dolgozók biztonságosan csatlakozhassanak a cég hálózatához.
* **Hálózati eszközök telepítése és beállítása:** A cég elkötelezett amellett, hogy a legmodernebb hálózati eszközöket alkalmazza. A routerek, switch-ek és egyéb eszközök rendszeres frissítése és karbantartása elengedhetetlen a hálózat zökkenőmentes működéséhez.
* Magas erőforrás-igényű gépek: A munkájukból adódóan a következő részlegek számára nagy teljesítményű gépeket kell elhelyezni: Grafikai Tervező, Webfejlesztő, Hangmérnök, Zenei Producer, Zenei Szerkesztő, IT szakember

A **Kihagyhatatlan Kincsek Kiadó** egy erős és biztonságos hálózatot szeretne, amely segíti a zenei produkciók hatékony lebonyolítását és a cég működésének zökkenőmentességét. A csapatunk feladata, hogy ezeket az igényeket teljesítse, és olyan hálózati megoldásokat kínáljon, amelyek hozzájárulnak a cég sikeréhez.

1.5 Csapatunk

A hálózat kiépítésével csapatunkat, Csete Ádámot, Nagy Barnabást és Téringer Gergőt bízták meg. A csapatunk által elvégzendő feladatok listája:

* Közös feladatok:
  + Logikai-, fizikai topológia elkészítése
  + Hardveres eszközök kiválasztása, árkalkuláció készítése
  + A munka dokumentálása
  + Funkcionális tesztelés
* Nagy Barnabás:
  + Felhőszolgáltatások alkalmazása
  + LAN hálózat biztonsága
  + WAN hálózat tervezése és konfigurálása
  + Automatizált mentés a szervereken
* Csete Ádám:
  + Hálózati eszközök konfigurálása
* Téringer Gergő:
  + Szerverek szolgáltatásainak telepítése és konfigurálása
  + Automatizált folyamatok megvalósítása

A feladatok felosztása, közös megbeszélést követően kerülnek kiosztásra, és mindig egy másik tag fogja tesztelni, annak a működését. A fent említett, feladatkörökön kívül, a csapattársak, segítenek egymásnak, tehát mindenki részt vesz a hálózat minden részének konfigurálásában.

A feladat elkészítése során a következő operációs rendszereket fogjuk használni:

* Debian 12.7
* Windows 10
* Windows Server 2022

1.6 Csapatmunka

A csoportmunka során a csapatunk Messengeren, Discordon keresztül kommunikál, a feladatok kiosztását, pedig a Trello menedzsment szoftveren keresztül végezzük, mivel kis csapat létszámra alkalmas, és ingyenes működése mellett rengeteg bővítése lehetőséget tartalmaz a szoftver. A fájlok megosztására a Google Drive SaaS felhőszolgáltatást használjuk, illetve GitHub-ra mentjük a konfigurációs fájlokat és Ansible scripteket. Az *1. képen* látható a GitHub, a *2. képen* pedig a Trello. A képek 2025.02.22.-én készültek.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

. kép - GitHub: 2025.02.22

A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

. kép - Trello: 2025.02.22

A *3. képen* a közös kommunikációra használt Messenger csoport látható.

A képen szöveg, Emberi arc, képernyőkép, ember látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

. kép - Messenger

1. Hálózat

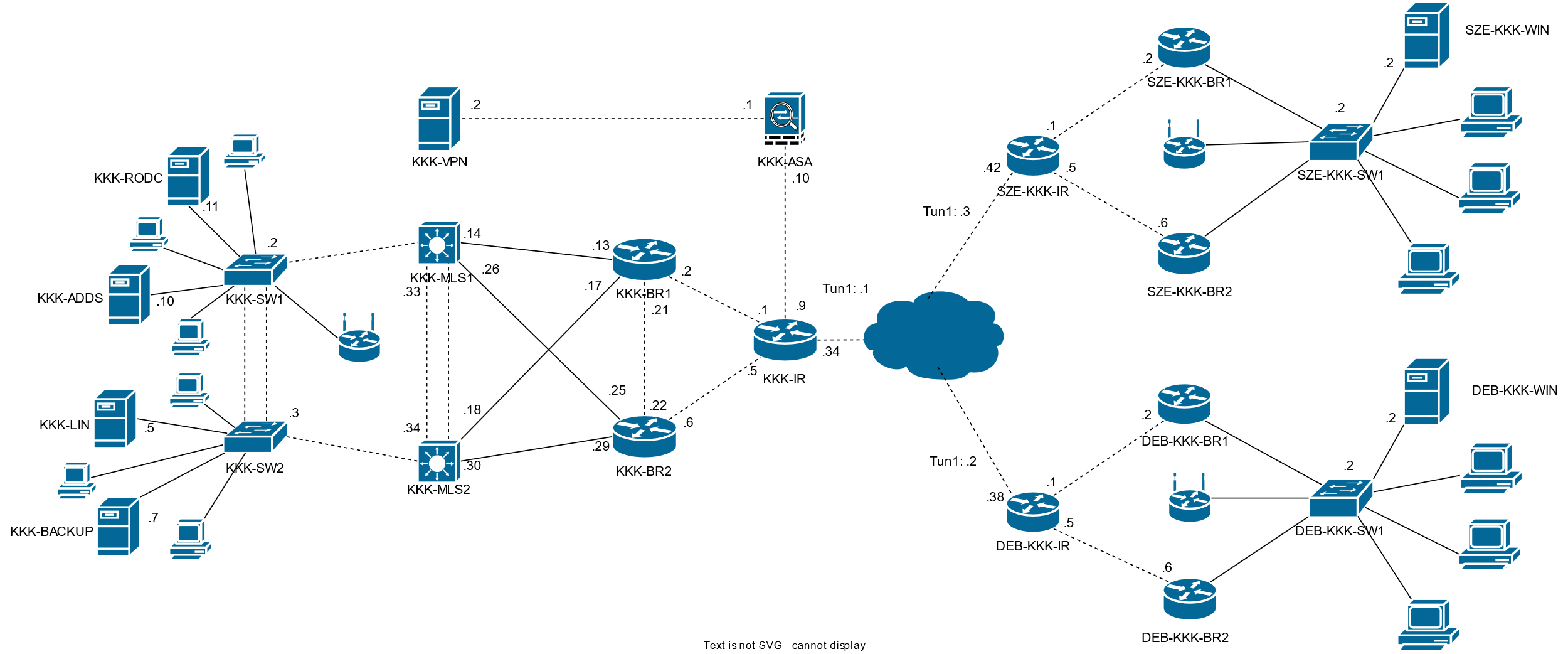
A Kihagyhatatlan Kincsek Kiadó hálózatát a következő rész mutatja be.

2.1 LAN

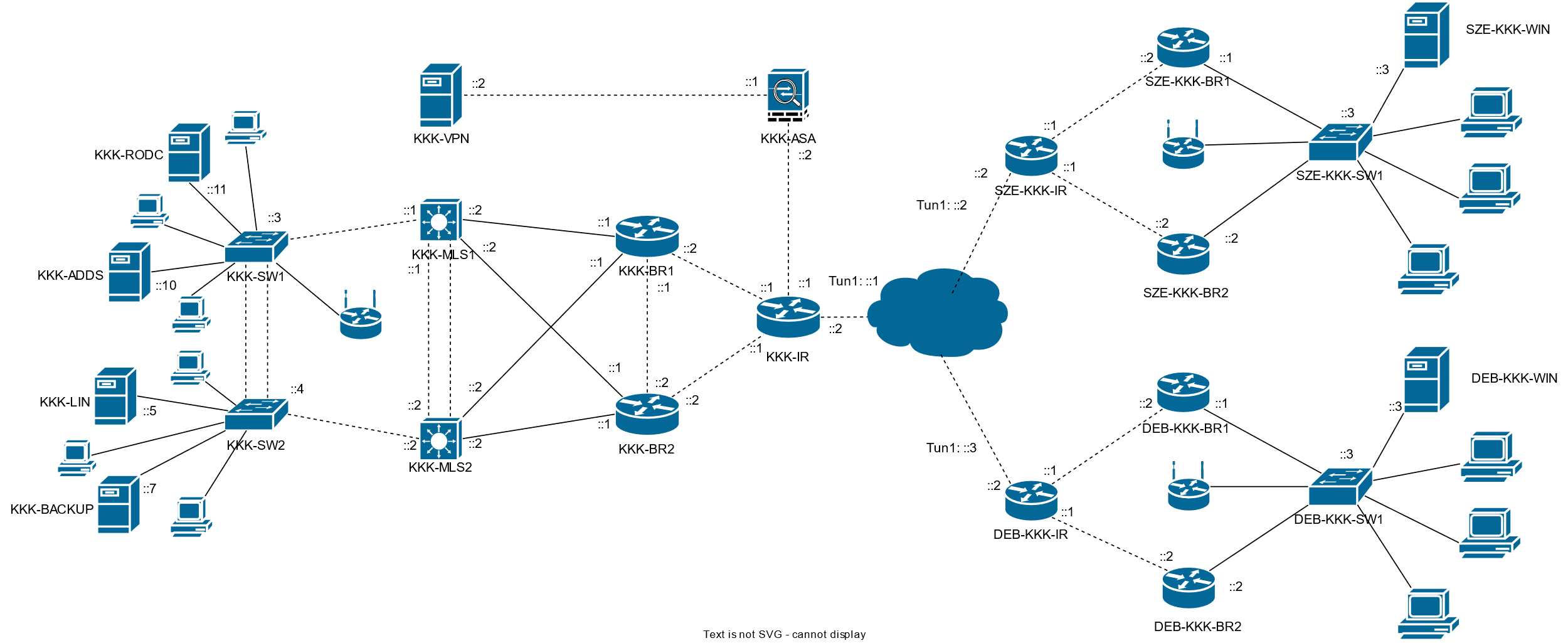
A következő alcímek alatt találhatók információk mind a három telephely helyi hálózatáról.

2.1.1 Logikai topológia

A *4. képen* az IPv4-es, az *5. képen* pedig az IPV6-os címek hoszt részeit tartalmazó logikai topológia látható, amiken mind a három telephely rajta van.



. kép - Logikai topológia: IPv4 címek



. kép - Logikai topológia: IPv6 címe

2.1.2 Fizikai topológia

A cégtől kapott tervrajzok és igényeik alapján lett a fizikai topológia kialakítva.

2.1.2.1 Épületek paraméterei

A központi épület négy szintből áll, míg a két fiókirodában csak földszint található. A központi épület közepén helyezkedik el a lépcső, és szintjeinek belső elosztása (a legfelső, harmadik szint kivételével) azonos. Ezért a földszint paraméterei megegyeznek a többi szint paramétereivel. A harmadik szint annyiban tér el, hogy néhány ajtó más helyre került, valamint a bal felső helyiségben egy válaszfal található, ami a 3,36 méter hosszú fallal párhuzamos és a mellette lévő két helyiség falával egy vonalban van. A dupla ajtók az épületek bejáratait jelölik. A budapesti épület paraméterei a *6. képen*, a szegedié a *7. képen*, a debrecenié pedig a *8. képen* láthatók.

A képen diagram, Tervrajz, Műszaki rajz, sematikus rajz látható

Automatikusan generált leírás

6. kép - A budapesti épület paraméterei

A képen diagram, Tervrajz, sor, Műszaki rajz látható

Automatikusan generált leírás

7. kép - A szegedi épület paraméterei

A képen diagram, Tervrajz, sor, képernyőkép látható

Automatikusan generált leírás

8. kép - A debreceni épület paraméterei

2.1.2.2 Topológia

A hálózat fizikai kialakítása során a cég kérésére nem helyeztünk el kábelt a mellékhelyiségek falában, hogy megóvjuk az ott lévő burkolatot. Ennek következtében bizonyos esetekben szinte az egész épületet meg kellett kerülni. Mindegyik épület minden szintjének belmagassága 3 méter. A központban a legtávolabbi PC is csak körülbelül 50 méter hosszú kábellel csatlakozik a kapcsolóhoz, ami bőven megfelel az Ethernet-szabvány által előírt 100 méteres maximális kábelhossznak. Ezáltal nem kell aggódni amiatt, hogy a földszinten lévő végpontok hálózati elérése jelentősen romlana a nagy távolság miatt. A kábelek minden épületben a falban futnak. A központi épületben minden szinten a jobb felső helyiség sarkában kerültek felvezetésre a legfelső emeleten található kapcsolókhoz. A fizikai topológiákon szürke vonal jelzi, hogy hol fut a kábel a falban. Minden helyiségbe dupla kábel lett vezetve és ezáltal dupla fali aljzat lett elhelyezve, hogy később, ha szükséges, rendelkezésre álljon tartalék. Így a fizikai topológiákon a zöld vonalak két UTP kábelt jelölnek. A fizikai topológiák a felsorolásban megadott képeken tekinthetők meg.

* budapesti épület földszintje a *9. képen*,
* budapesti épület első emelete a *10. képen*,
* budapesti épület második emelete a *11. képen*,
* budapesti épület harmadik emelete a *12. képen*,
* debreceni épület földszintje a *13. képen*,
* szegedi épület földszintje a *14. képen*

látható.

A képen szöveg, diagram, képernyőkép, Tervrajz látható

Automatikusan generált leírás

9. kép - Fizikai topológia: budapesti épület földszintje

A képen szöveg, diagram, képernyőkép, Tervrajz látható

Automatikusan generált leírás

10. kép - Fizikai topológia: budapesti épület első emelete

A képen diagram, szöveg, képernyőkép, Tervrajz látható

Automatikusan generált leírás

11. kép - Fizikai topológia: budapesti épület második emelete

A képen szöveg, képernyőkép, diagram, Tervrajz látható

Automatikusan generált leírás

12. kép - Fizikai topológia: budapesti épület harmadik emelete

A képen szöveg, diagram, képernyőkép, Tervrajz látható

Automatikusan generált leírás

13. kép - Fizikai topológia: debreceni épület

A képen diagram, képernyőkép, sor, Tervrajz látható

Automatikusan generált leírás

14. kép - Fizikai topológia: szegedi épület

2.1.3 VLAN-ok

A vállalatnál dolgozók foglalkozási köre alapján is kerültek kialakításra a VLAN-ok. Erre elsősorban azért volt szükség, hogy a dolgozók hálózati forgalma jól elkülönüljön, és ezáltal a későbbi vállalati irányelvek könnyebben érvényesíthetők legyenek. Emellett ez a megoldás biztosította a hálózat hatékony szegmentálását, valamint a szórási tartományok méretének csökkentését. Mivel a különböző telephelyeken eltérő szakmák képviseltetik magukat, a VLAN-ok is ennek megfelelően, telephelyenként kerültek kialakításra. A VLAN-ok azonosítását az adott telephely privát IP-címtartományának második oktettje segíti, amely megegyezik a VLAN-ok első két számjegyével. Az *1. táblázatban* látható a debreceni, a *2. táblázatban* a szegedi, a *3. táblázatban* pedig a központi telephely VLAN-jai.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Debrecen VLAN** | | | **VLAN neve** | **VLAN száma** | | FIOKVEZETO | 3741 | | TEHETSEG | 3779 | | KIADO | 3737 | | |  |  | | --- | --- | | **Szeged VLAN** | | | **VLAN neve** | **VLAN száma** | | FIOKVEZETO | 2319 | | KOZONSEG | 2323 | | ESEMENY | 2341 | |
| . táblázat - VLAN: Debrecen | 2. táblázat - VLAN: Szeged |

|  |  |
| --- | --- |
| **Központ VLAN** | |
| **VLAN neve** | **VLAN száma** |
| ADMINISZTRACIO | 1147 |
| MARKETING | 1173 |
| VEZETO | 1171 |
| KIADO | 1197 |
| ZENE | 1119 |

3. táblázat - VLAN: Központ

Emellett vannak olyan VLAN-ok, amik mind a három telephelyen megjelennek. Ez a MANAGEMENT, NATIVE VLAN. A fizikai biztonságot nyújtó Patent Cégcsoport számára lett létrehozva a BIZTONSAG VLAN. Az IT VLAN a rendszergazdáé, az SRV VLAN a szervereké, a WIFI VLAN pedig a AP-ké. A BLACKHOLE VLAN pedig a nem használt kapcsoló portoknak lett létrehozva. Ezek számai a *4. táblázatban* láthatóak.

|  |  |
| --- | --- |
| **Közös VLAN** | |
| **VLAN neve** | **VLAN száma** |
| MANAGEMENT | 37 |
| BLACKHOLE | 29 |
| NATIVE | 11 |
| WIFI | 41 |
| SRV | 13 |
| BIZTONSAG | 20 |
| IT | 69 |

4. táblázat - VLAN: Közös

2.1.4 Kapcsolók

A központban mindegyik második rétegbeli kapcsoló a G1/1/1-es interfészével kapcsolódik az adott MLS kapcsolóhoz, vagyis a KKK-SW1 a KKK-MLS1-hez a KKK-SW2 a KKK-MLS2-höz. A harmadik rétegbeli kapcsolók pedig a G1/1/3-as interfészeivel kapcsolódnak. Mindkét második rétegbeli kapcsolón a kettő közötti port-channel-ben a G1/1/2 és a G1/1/3 interfésze van benne. A KKK-SW1 G1/0/1 interfészéhez a KKK-RODC, a G1/0/2 porthoz pedig a KKK-ADDS szerver csatlakozik. A KKK-SW2 G1/0/1 interfészéhez a KKK-LIN, a G1/0/2-hőz a KKK-BACKUP kapcsolódik. A földszint és az első emelet AP-ja a KKK-SW1 G1/0/3 és G1/0/4 interfészéhez, a többi AP pedig lentről-felfelé a KKK-SW2 G1/0/3 és G1/0/4 portjaihoz kapcsolódik. A KKK-SW1 G1/0/5-as portjától a G1/0/36-os portjáig csatlakozik az első 16 kliens. Fentről lefele, illetve minden szinten a szerver teremnek megfelelő helyiség az első és utána a fizikai topológián balra haladva van a kliensek sorrendje. Ez alapján KKK-SW1 kapcsolón a Kiadói vezető az első és a Hangmérnök az utolsó. A port kiosztás ugyanígy folytatódik a KKK-SW2 kapcsolón. Mindkét kapcsolón lett hagyva 8-8 port az esetleges későbbi bővítéshez.

Mind a debreceni, mind pedig a szegedi telephelyen is a kapcsoló G1/1/1 interfésze az ottani BR1 forgalomirányítóhoz kapcsolódik, a G1/1/2 interfésze pedig a BR2-höz. Mindkét telephelyen a szerverek a G1/0/40 interfészhez kapcsolódnak, az AP-k pedig a G1/0/35 porthoz kapcsolódnak. A fizikai topológián a debreceni telepen balra indulva, a szegedin pedig jobbra indulva lettek kiosztva a portok a G1/0/1-től.

2.1.5 IPv4 és IPv6 címzés

A hálózat dual-stack, vagyis minden eszköz rendelkezik IPv6-os és IPv4-es címmel. IPv4 esetén 10.0.0.0/8 privát IPv4 cím tartomány lett felosztva a telephelyek között. A központi telephelyen 10.11.0.0/16, debrecenin a 10.37.0.0/16, a szegedin pedig a 10.23.0.0/16 tartomány lett tovább osztva. Az *5. táblázatban* a központi, a *6. táblázatban* a szegedi, a *7. táblázatban* pedig a debreceni tartomány tovább osztása látható.

|  |  |  |
| --- | --- | --- |
| **Központ - IPv4** | | |
| **IPv4 hálózatcím** | **Maszk** | **Megjegyzés** |
| 10.11.47.0 | /27 | ADMINISZTRACIO VLAN |
| 10.11.73.0 | /26 | MARKETING VLAN |
| 10.11.71.0 | /28 | VEZETO VLAN |
| 10.11.97.0 | /27 | KIADO VLAN |
| 10.11.19.0 | /26 | ZENE VLAN |
| 10.11.208.0 | /20 | BIZTONSÁG |
| 10.11.137.0 | /26 | MANAGEMENT VLAN |
| 10.11.140.0 | /22 | WIFI VLAN |
| 10.11.113.0 | /28 | SRV VLAN |
| 10.11.169.0 | /28 | IT VLAN |
| 10.11.250.0 | /24 | PONT-PONT KAPCSOLATOK |
| 10.11.199.0 | /29 | ASA ALHÁLÓZAT |
| 10.11.255.0 | /24 | Loopback |

5. táblázat - IPv4 cím felosztás: Központ

|  |  |  |
| --- | --- | --- |
| **Szeged - IPv4** | | |
| **IPv4 hálózatcím** | **Maszk** | **Megjegyzés** |
| 10.23.19.0 | /29 | FIOKVEZETO VLAN |
| 10.23.23.0 | /28 | KOZONSEG VLAN |
| 10.23.41.0 | /28 | ESEMENY VLAN |
| 10.23.137.0 | /27 | MANAGEMENT VLAN |
| 10.23.140.0 | /23 | WIFI VLAN |
| 10.23.208.0 | /20 | BIZTONSÁG |
| 10.23.113.0 | /29 | SRV VLAN |
| 10.23.169.0 | /29 | IT VLAN |
| 10.23.250.0 | /24 | PONT-PONT KAPCSOLATOK |
| 10.23.255.0 | /24 | Loopback |

6. táblázat - IPv4 cím felosztás: Szeged

|  |  |  |
| --- | --- | --- |
| **Debrecen - IPv4** | | |
| **IPv4 hálózatcím** | **Maszk** | **Megjegyzés** |
| 10.37.41.0 | /29 | FIOKVEZETO VLAN |
| 10.37.79.0 | /28 | TEHETSEG VLAN |
| 10.37.37.0 | /28 | KIADO VLAN |
| 10.37.137.0 | /27 | MANAGEMENT VLAN |
| 10.37.140.0 | /23 | WIFI VLAN |
| 10.37.208.0 | /20 | BIZTONSÁG |
| 10.37.113.0 | /29 | SRV VLAN |
| 10.37.169.0 | /29 | IT VLAN |
| 10.37.250.0 | /24 | PONT-PONT KAPCSOLATOK |
| 10.37.255.0 | /24 | Loopback |

7. táblázat - IPv4 cím felosztás: Debrecen

IPv6 esetében a 2001:db8:c1c1::/48-as címtartomány lett szétosztva a telephelyek között. A központ kapta a 2001:db8:c1c1:ab00::/56-os, a szegedi telep a 2001:db8:c1c1:c100::/56-os, a debreceni telep pedig a 2001:db8:c1c1:ba00::/56-os tartomány kapta. Ezek lettek tovább osztva /64-es tartományokra. A pont-pont kapcsoltoknál csak link-local címek lettek használva. A *8. táblázatban* a központi, a *9. táblázatban* a szegedi, a *10. táblázatban* pedig a debreceni tartomány tovább osztása láható.

|  |  |
| --- | --- |
| **Központ - IPv6** | |
| **IPv6 alhálózat** | **Megjegyzés** |
| 2001:db8:c1c1:abb1::/64 | ADMINISZTRACIO VLAN |
| 2001:db8:c1c1:abf8::/64 | MARKETING VLAN |
| 2001:db8:c1c1:abba::/64 | VEZETO VLAN |
| 2001:db8:c1c1:abf6::/64 | KIADO VLAN |
| 2001:db8:c1c1:ab8d::/64 | ZENE VLAN |
| 2001:db8:c1c1:ab20::/64 | BIZTONSÁG |
| 2001:db8:c1c1:ab8a::/64 | MANAGEMENT VLAN |
| 2001:db8:c1c1:abda::/64 | WIFI VLAN |
| 2001:db8:c1c1:abfa::/64 | SRV VLAN |
| 2001:db8:c1c1:abb5::/64 | IT VLAN |
| 2001:db8:c1c1:ab22::/64 | ASA alhálózata |
| 2001:db8:c1c1:abfe::/64 | Loopback |

8. táblázat – IPv6 cím felosztás: Központ

|  |  |
| --- | --- |
| **Szeged - IPv6** | |
| **IPv6 alhálózat** | **Megjegyzés** |
| 2001:db8:c1c1:babf::/64 | FIOKVEZETO VLAN |
| 2001:db8:c1c1:ba85::/64 | KOZONSEG VLAN |
| 2001:db8:c1c1:babc::/64 | ESEMENY VLAN |
| 2001:db8:c1c1:ba38::/64 | MANAGEMENT VLAN |
| 2001:db8:c1c1:bac1::/64 | WIFI VLAN |
| 2001:db8:c1c1:ba20::/64 | BIZTONSÁG |
| 2001:db8:c1c1:baba::/64 | SRV VLAN |
| 2001:db8:c1c1:bab3::/64 | IT VLAN |
| 2001:db8:c1c1:ba48::/64 | Loopback |

9. táblázat – IPv6 cím felosztás: Szeged

|  |  |
| --- | --- |
| **Debrecen - IPv6** | |
| **IPv6 alhálózat** | **Megjegyzés** |
| 2001:db8:c1c1:c155::/64 | FIOKVEZETO VLAN |
| 2001:db8:c1c1:c1a5::/64 | TEHETSEG VLAN |
| 2001:db8:c1c1:c15a::/64 | KIADO VLAN |
| 2001:db8:c1c1:c1e3::/64 | MANAGEMENT VLAN |
| 2001:db8:c1c1:c1c1::/64 | WIFI VLAN |
| 2001:db8:c1c1:c120::/64 | BIZTONSÁG |
| 2001:db8:c1c1:c1ca::/64 | SRV VLAN |
| 2001:db8:c1c1:c1b5::/64 | IT VLAN |
| 2001:db8:c1c1:c1d2::/64 | Loopback |

10. táblázat – IPv6 cím felosztás: Debrecen

A DMVPN alagútban a 10.255.255.0/28-as címtartomány lett felosztva.

A kliensek DHCP-vel kapnak IPv4 és IPv6 címeket. IPv6 esetén állapottartó DHCPv6 szerver lett konfigurálva. A következő parancsok lettek alkalmazva DHCP kliensek felé néző interfészeken a megfelelő működés érdekében. Ezek segítik a szórásos üzenetek célba érését, illetve az állapottartó DHCPv6 szerver működését:

ipv6 dhcp relay destination <DHCP szerver IPv6 címe>

ipv6 nd managed-config-flag

ipv6 nd prefix default no-autoconfig

ip helper-address <DHCP szerver IPv4 címe>

A hálózati eszközök és a szerverek statikusan kaptak IPv4 és IPv6 címet. IPv4 címei a

* központi forgalomirányítóknak a *11. táblázatban*,
* debreceni forgalomirányítóknak a *12. táblázatban*,
* szegedi forgalomirányítóknak a *13. táblázatban*,
* kapcsolóknak a *14. táblázatban*,
* szervereknek a *15. táblázatban* láthatóak.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Központ - Router** | | | | | |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | **Interfész** |
| KKK-IR | G0/0/0 | 10.11.250.1 | /30 | KKK-BR1 | G0/0/2 |
| G0/0/1 | 10.11.250.5 | /30 | KKK-BR2 | G0/0/2 |
| Lo0 | 10.11.255.1 | /32 | - | |
| Dialer1 (g0/1/0) | 203.5.100.34 | /30 | ISP | Virtual-Access (g0/0/0) |
| Tun1 | 10.255.255.1 | /28 | SZE-KKK-IR DEB-KKK-IR | Tun1 |
| G0/0/2 | 10.11.250.9 | /30 | KKK-ASA | G0/8 |
| KKK-BR1 | G0/0/2 | 10.11.250.2 | /30 | KKK-IR | G0/0/0 |
| G0/0/0 | 10.11.250.13 | /30 | KKK-MLS1 | G1/1/1 |
| G0/0/1 | 10.11.250.17 | /30 | KKK-MLS2 | G1/1/1 |
| G0/1/0 | 10.11.250.21 | /30 | KKK-BR2 | G0/1/0 |
| Lo0 | 10.11.255.2 | /32 | - | |
| KKK-BR2 | G0/0/2 | 10.11.250.6 | /30 | KKK-IR | G0/0/1 |
| G0/0/0 | 10.11.250.25 | /30 | KKK-MLS1 | G1/1/2 |
| G0/0/1 | 10.11.250.29 | /30 | KKK-MLS2 | G1/1/2 |
| G0/1/0 | 10.11.250.22 | /30 | KKK-BR1 | G0/1/0 |
| Lo0 | 10.11.255.3 | /32 | - | |
| KKK-ASA | G0/5 | 10.11.199.1 | /29 | KKK-VPN | ens33 |
| G0/8 | 10.11.250.10 | /30 | KKK-IR | G0/0/2 |
| Lo0 | 10.11.255.6 | /32 | - | |

11. táblázat - IPv4 címek: Központ forgalomirányítói

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Debrecen - Router** | | | | | |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | **Interfész** |
| DEB-KKK-IR | G0/0/0 | 10.37.250.1 | /30 | DEB-KKK-BR1 | G0/0/2 |
| G0/0/1 | 10.37.250.5 | /30 | DEB-KKK-BR2 | G0/0/2 |
| Lo0 | 10.37.255.1 | /32 | - | |
| Dialer1 (G0/1/0) | 203.5.100.38 | /30 | ISP | Virtual-Access (G0/0/1) |
| Tun1 | 10.255.255.2 | /28 | SZE-KKK-IR KKK-IR | Tun1 |
| DEB-KKK-BR1 | G0/0/2 | 10.37.250.2 | /30 | DEB-KKK-IR | G0/0/0 |
| G0/0/0.41 | 10.37.41.5 | /29 | DEB-KKK-SW1 | G1/1/1 |
| G0/0/0.79 | 10.37.79.13 | /28 |
| G0/0/0.37 | 10.37.37.13 | /28 |
| G0/0/0.137 | 10.37.137.29 | /27 |
| G0/0/0.140 | 10.37.141.251 | /23 |
| G0/0/0.208 | 10.37.223.251 | /20 |
| G0/0/0.113 | 10.37.113.5 | /29 |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | **Interfész** |
| DEB-KKK-BR1 | G0/0/0.169 | 10.37.169.5 | /29 | DEB-KKK-SW1 | G1/1/1 |
| Lo0 | 10.23.255.2 | /32 | - | |
| DEB-KKK-BR2 | G0/0/2 | 10.23.250.6 | /30 | DEB-KKK-IR | G0/0/1 |
| G0/0/0.41 | 10.37.41.6 | /29 | DEB-KKK-SW1 | G1/1/2 |
| G0/0/0.79 | 10.37.79.14 | /28 |
| G0/0/0.37 | 10.37.37.14 | /28 |
| G0/0/0.137 | 10.37.137.30 | /27 |
| G0/0/0.140 | 10.37.141.252 | /23 |
| G0/0/0.208 | 10.37.223.252 | /20 |
| G0/0/0.113 | 10.37.113.6 | /29 |
| G0/0/0.169 | 10.37.169.6 | /29 |
| Lo0 | 10.37.255.3 | /32 | - | |

12. táblázat - IPv4 címek: Debrecen forgalomirányítói

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Szeged - Router** | | | | | | | |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | | | **Interfész** |
| SZE-KKK-IR | G0/0/0 | 10.23.250.1 | /30 | SZE-KKK-BR1 | | | G0/0/2 |
| G0/0/1 | 10.23.250.5 | /30 | SZE-KKK-BR2 | | | G0/0/2 |
| Lo0 | 10.23.255.1 | /32 | - | | | |
| Dialer1 (G0/1/0) | 203.5.100.42 | /30 | ISP | | | Virtual-Access (G0/0/2) |
| Tun1 | 10.255.255.3 | /28 | DEB-KKK-IR KKK-IR | | | Tun1 |
| SZE-KKK-BR1 | G0/0/2 | 10.23.250.2 | /30 | SZE-KKK-IR | | | G0/0/0 |
| G0/0/0.19 | 10.23.19.5 | /29 | SZE-KKK-SW1 | | | G1/1/1 |
| G0/0/0.23 | 10.23.23.13 | /28 |
| G0/0/0.41 | 10.23.41.13 | /28 |
| G0/0/0.137 | 10.23.137.29 | /27 |
| G0/0/0.140 | 10.23.141.251 | /23 |
| G0/0/0.208 | 10.23.223.251 | /20 |
| G0/0/0.113 | 10.23.113.5 | /29 |
| G0/0/0.169 | 10.23.169.5 | /29 |
| Lo0 | 10.23.255.2 | /32 | - | | | |
| SZE-KKK-BR2 | G0/0/2 | 10.23.250.6 | /30 | SZE-KKK-IR | G0/0/1 | | | |
| G0/0/0.19 | 10.23.19.6 | /29 | SZE-KKK-SW1 | | G1/1/2 | |
| G0/0/0.23 | 10.23.23.14 | /28 |
| G0/0/0.41 | 10.23.41.14 | /28 |
| G0/0/0.137 | 10.23.137.30 | /27 |
| G0/0/0.140 | 10.23.141.252 | /23 |
| G0/0/0.208 | 10.23.223.252 | /20 |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | | **Interfész** | |
| SZE-KKK-BR2 | G0/0/0.113 | 10.23.113.6 | /29 | SZE-KKK-SW1 | | G1/1/2 | |
| G0/0/0.169 | 10.23.169.6 | /29 |
| Lo0 | 10.23.255.3 | /32 | - | | | |

13. táblázat - IPv4 címek: Szeged forgalomirányítói

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Kapcsolók** | | | | | | |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | | **Interfész** |
| KKK-MLS1 | G1/1/1 | 10.11.250.14 | /30 | KKK-BR1 | | G0/0/0 |
| G1/1/2 | 10.11.250.26 | /30 | KKK-BR2 | | G0/0/0 |
| Port1 (G1/0/1-G1/0/2) | 10.11.250.33 | /30 | KKK-MLS2 | | Port1 (G1/0/1-G1/0/2) |
| Lo0 | 10.11.255.4 | /32 | - | | |
| VLAN 1147 | 10.11.47.29 | /27 | - | | |
| VLAN 1173 | 10.11.73.59 | /26 |
| VLAN 1171 | 10.11.71.13 | /28 |
| VLAN 1197 | 10.11.97.29 | /27 |
| VLAN 1119 | 10.11.19.59 | /26 |
| VLAN 20 | 10.11.223.251 | /20 |
| VLAN 37 | 10.11.137.59 | /26 |
| VLAN 41 | 10.11.143.251 | /22 |
| VLAN 13 | 10.11.113.13 | /28 |
| VLAN 69 | 10.11.169.13 | /28 |
| KKK-MLS2 | G1/1/1 | 10.11.250.18 | /30 | KKK-BR1 | | G0/0/1 |
| G1/1/2 | 10.11.250.30 | /30 | KKK-BR2 | | G0/0/1 |
| Port1 (G1/0/1-G1/0/2) | 10.11.250.34 | /30 | KKK-MLS1 | | Port1 (G1/0/1-G1/0/2) |
| Lo0 | 10.11.255.5 | /32 | - | | |
| VLAN 1147 | 10.11.47.30 | /27 | - | | |
| VLAN 1173 | 10.11.73.60 | /26 |
| VLAN 1171 | 10.11.71.14 | /28 |
| VLAN 1197 | 10.11.97.30 | /27 |
| VLAN 1119 | 10.11.19.60 | /26 |
| VLAN 20 | 10.11.223.252 | /20 |
| VLAN 37 | 10.11.137.60 | /26 |
| VLAN 41 | 10.11.143.252 | /22 |
| VLAN 13 | 10.11.113.14 | /28 |
| VLAN 69 | 10.11.169.14 | /28 |
| KKK-SW1 | VLAN 37 | 10.11.137.2 | /26 | - | | |
| KKK-SW2 | VLAN 37 | 10.11.137.3 | /26 | - | | |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | **Interfész** | |
| DEB-KKK-SW1 | VLAN 37 | 10.37.137.2 | /27 | - | | |
| SZE-KKK-SW1 | VLAN 37 | 10.23.137.2 | /27 | - | | |

14. táblázat - IPv4 címek: kapcsolók

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Hosztok** | | | | | |
| **Eszköz** | **Interfész** | **IPv4 cím** | **Maszk** | **Másik eszköz** | **Interfész** |
| KKK-LIN | ens33 | 10.11.113.5 | /28 | KKK-SW2 | G1/0/1 |
| KKK-ADDS | Ethernet1 | 10.11.113.10 | KKK-SW1 | G1/0/2 |
| KKK-RODC | Ethernet1 | 10.11.113.11 | KKK-SW1 | G1/0/1 |
| KKK-VPN | ens33 | 10.11.199.2 | /29 | KKK-ASA | G0/5 |
| KKK-BACKUP | ens33 | 10.11.113.7 | /28 | KKK-SW2 | G1/0/2 |
| DEB-KKK-WIN | Ethernet1 | 10.37.113.2 | /29 | DEB-KKK-SW1 | G1/0/40 |
| SZE-KKK-WIN | Ethernet1 | 10.23.113.2 | /29 | SZE-KKK-SW1 | G1/0/40 |

15. táblázat - IPv4 címek: hosztok

IPv6 címei a

* központi forgalomirányítóknak a *16. táblázatban*,
* debreceni forgalomirányítóknak a *17. táblázatban*,
* szegedi forgalomirányítóknak a *18. táblázatban*,
* kapcsolóknak a *19. táblázatban*,
* szervereknek a *20. táblázatban*

láthatóak.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Központ - Router** | | | | | | |
| **Eszköz** | **Interfész** | **IPv6 cím** | | **Link local** | **Másik eszköz** | **Interfész** |
| KKK-IR | G0/0/0 | fe80:1::1 | | | KKK-BR1 | G0/0/2 |
| G0/0/1 | fe80:2::1 | | | KKK-BR2 | G0/0/2 |
| Lo0 | 2001:db8:c1c1:abfe::1/128 | | fe80:abfe::1 | - | |
| Dialer1 (g0/1/0) | 2001:db8:baba:ab12::2/64 | | fe80:ab12::2 | ISP | Virtual-Access (g0/0/0) |
| Tun1 | fe80:255::1 | | | SZE-KKK-IR DEB-KKK-IR | Tun1 |
| G0/0/2 | fe80:9::1 | | | KKK-ASA | G0/8 |
| KKK-BR1 | G0/0/2 | fe80:1::2 | | | KKK-IR | G0/0/0 |
| G0/0/0 | fe80:3::1 | | | KKK-MLS1 | G1/1/1 |
| G0/0/1 | fe80:4::1 | | | KKK-MLS2 | G1/1/1 |
| G0/1/0 | fe80:5::1 | | | KKK-BR2 | G0/1/0 |
| Lo0 | 2001:db8:c1c1:abfe::2/128 | | fe80:abfe::2 | - | |
| **Eszköz** | **Interfész** | **IPv6 cím** | **Link local** | | **Másik eszköz** | **Interfész** |
| KKK-BR2 | G0/0/2 | fe80:2::2 | | | KKK-IR | G0/0/1 |
| G0/0/0 | fe80:6::1 | | | KKK-MLS1 | G1/1/2 |
| G0/0/1 | fe80:7::1 | | | KKK-MLS2 | G1/1/2 |
| G0/1/0 | fe80:5::2 | | | KKK-BR1 | G0/1/0 |
| Lo0 | 2001:db8:c1c1:abfe::3/128 | | fe80:abfe::3 | - | |
| KKK-ASA | G0/5 | 2001:db8:c1c1:ab22::1/64 | | fe80:ab22::1 | KKK-VPN | ens33 |
| G0/8 | fe80:9::2 | | | KKK-IR | G0/0/2 |
| Lo0 | 2001:db8:c1c1:abfe::4/128 | | fe80:abfe::4 | - | |

16. táblázat – IPv6 címek: Központ forgalomirányítói

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Debrecen - Router** | | | | | |
| **Eszköz** | **Interfész** | **IPv6 cím** | **Link local** | **Másik eszköz** | **Interfész** |
| DEB-KKK-IR | G0/0/0 | fe80:1::1 | | DEB-KKK-BR1 | G0/0/2 |
| G0/0/1 | fe80:2::1 | | DEB-KKK-BR2 | G0/0/2 |
| Lo0 | 2001:db8:c1c1:c1d2::1/128 | fe80:c1d2::1 | - | |
| Dialer1 (G0/1/0) | 2001:db8:baba:c112::2/64 | fe80:c112::2 | ISP | Virtual-Access (G0/0/1) |
| Tun1 | fe80:255::2 | | SZE-KKK-IR KKK-IR | Tun1 |
| DEB-KKK-BR1 | G0/0/2 | fe80:1::2 | | DEB-KKK-IR | G0/0/0 |
| G0/0/0.41 | 2001:db8:c1c1:c155::1/64 | fe80:c155::1 | DEB-KKK-SW1 | G1/1/1 |
| G0/0/0.79 | 2001:db8:c1c1:c1a5::1/64 | fe80:c1a5::1 |
| G0/0/0.37 | 2001:db8:c1c1:c15a::1/64 | fe80:c15a::1 |
| G0/0/0.137 | 2001:db8:c1c1:c1e3::1/64 | fe80:c1e3::1 |
| G0/0/0.140 | 2001:db8:c1c1:c1c1::1/64 | fe80:c1c1::1 |
| G0/0/0.208 | 2001:db8:c1c1:c120::1/64 | fe80:c1a20::1 |
| G0/0/0.113 | 2001:db8:c1c1:c1ca::1/64 | fe80:c1ca::1 |
| G0/0/0.169 | 2001:db8:c1c1:c1b5::1/64 | fe80:c1b5::1 |
| Lo0 | 2001:db8:c1c1:c1d2::2/128 | fe80:c1d2::2 | - | |
| DEB-KKK-BR2 | G0/0/2 | fe80:2::2 | | DEB-KKK-IR | G0/0/1 |
| G0/0/0.41 | 2001:db8:c1c1:c155::2/64 | fe80:c155::2 | DEB-KKK-SW1 | G1/1/2 |
| G0/0/0.79 | 2001:db8:c1c1:c1a5::2/64 | fe80:c1a5::2 |
| G0/0/0.37 | 2001:db8:c1c1:c15a::2/64 | fe80:c15a::2 |
| G0/0/0.137 | 2001:db8:c1c1:c1e3::2/64 | fe80:c1e3::2 |
| G0/0/0.140 | 2001:db8:c1c1:c1c1::2/64 | fe80:c1c1::2 |
| G0/0/0.208 | 2001:db8:c1c1:c120::2/64 | fe80:c1a20::2 |
| G0/0/0.113 | 2001:db8:c1c1:c1ca::2/64 | fe80:c1ca::2 |
| G0/0/0.169 | 2001:db8:c1c1:c1b5::2/64 | fe80:c1b5::2 |
| Lo0 | 2001:db8:c1c1:c1d2::3/128 | fe80:c1d2::3 | - | |

17. táblázat – IPv6 címek: Debrecen forgalomirányítói

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Szeged - Router** | | | | | |
| **Eszköz** | **Interfész** | **IPv6 cím** | **Link local** | **Másik eszköz** | **Interfész** |
| SZE-KKK-IR | G0/0/0 | fe80:1::1 | | SZE-KKK-BR1 | G0/0/2 |
| G0/0/1 | fe80:2::1 | | SZE-KKK-BR2 | G0/0/2 |
| Lo0 | 2001:db8:c1c1:ba48::1/128 | fe80:ba48::1 | - | |
| Dialer1 (G0/1/0) | 2001:db8:baba:ba12::2/64 | fe80:ba12::2 | ISP | Virtual-Access (G0/0/2) |
| Tun1 | fe80:255::3 | | DEB-KKK-IR KKK-IR | Tun1 |
| SZE-KKK-BR1 | G0/0/2 | fe80:1::2 | | SZE-KKK-IR | G0/0/0 |
| G0/0/0.19 | 2001:db8:c1c1:babf::1/64 | fe80:babf::1 | SZE-KKK-SW1 | G1/1/1 |
| G0/0/0.23 | 2001:db8:c1c1:ba85::1/64 | fe80:ba85::1 |
| G0/0/0.41 | 2001:db8:c1c1:babc::1/64 | fe80:babc::1 |
| G0/0/0.137 | 2001:db8:c1c1:ba38::1/64 | fe80:ba38::1 |
| G0/0/0.140 | 2001:db8:c1c1:bac1::1/64 | fe80:bac1::1 |
| G0/0/0.208 | 2001:db8:c1c1:ba20::1/64 | fe80:ba20::1 |
| G0/0/0.113 | 2001:db8:c1c1:baba::1/64 | fe80:baba::1 |
| G0/0/0.169 | 2001:db8:c1c1:bab3::1/64 | fe80:bab3::1 |
| Lo0 | 2001:db8:c1c1:ba48::2/128 | fe80:ba48::2 | - | |
| SZE-KKK-BR2 | G0/0/2 | fe80:2::2 | | SZE-KKK-IR | G0/0/1 |
| G0/0/0.19 | 2001:db8:c1c1:babf::2/64 | fe80:babf::2 | SZE-KKK-SW1 | G1/1/2 |
| G0/0/0.23 | 2001:db8:c1c1:ba85::2/64 | fe80:ba85::2 |
| G0/0/0.41 | 2001:db8:c1c1:babc::2/64 | fe80:babc::2 |
| G0/0/0.137 | 2001:db8:c1c1:ba38::2/64 | fe80:ba38::2 |
| G0/0/0.140 | 2001:db8:c1c1:bac1::2/64 | fe80:bac1::2 |
| G0/0/0.208 | 2001:db8:c1c1:ba20::2/64 | fe80:ba20::2 |
| G0/0/0.113 | 2001:db8:c1c1:baba::2/64 | fe80:baba::2 |
| G0/0/0.169 | 2001:db8:c1c1:bab3::2/64 | fe80:bab3::2 |
| Lo0 | 2001:db8:c1c1:ba48::3/128 | fe80:ba48::3 | - | |

18. táblázat – IPv6 címek: Szeged forgalomirányítói

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Központ - Switch** | | | | | |
| **Eszköz** | **Interfész** | **IPv6 cím** | **Link local** | **Másik eszköz** | **Interfész** |
| KKK-MLS1 | G1/1/1 | fe80:3::2 | | KKK-BR1 | G0/0/0 |
| G1/1/2 | fe80:6::2 | | KKK-BR2 | G0/0/0 |
| Port1 (G1/0/1-G1/0/2) | fe80:10::1 | | KKK-MLS2 | Port1 (G1/0/1-G1/0/2) |
| Lo0 | 2001:db8:c1c1:abfe::5/128 | fe80:abfe::5 | - | |
| VLAN 1147 | 2001:db8:c1c1:abb1::1/64 | fe80:abb1::1 | - | |
| VLAN 1173 | 2001:db8:c1c1:abf8::1/64 | fe80:abf8::1 |
| VLAN 1171 | 2001:db8:c1c1:abba::1/64 | fe80:abba::1 |
| VLAN 1197 | 2001:db8:c1c1:abf6::1/64 | fe80:abf6::1 |
| VLAN 1119 | 2001:db8:c1c1:ab8d::1/64 | fe80:ab8d::1 |
| **Eszköz** | **Interfész** | **IPv6 cím** | **Link local** | **Másik eszköz** | **Interfész** |
| KKK-MLS1 | VLAN 20 | 2001:db8:c1c1:ab20::1/64 | fe80:ab20::1 | - | |
| VLAN 37 | 2001:db8:c1c1:ab8a::1/64 | fe80:ab8a::1 |
| VLAN 41 | 2001:db8:c1c1:abda::1/64 | fe80:abda::1 |
| VLAN 13 | 2001:db8:c1c1:abfa::1/64 | fe80:abfa::1 |
| VLAN 69 | 2001:db8:c1c1:abb5::1/64 | fe80:aba5::1 |
| KKK-MLS2 | G1/1/1 | fe80:4::2 | | KKK-BR1 | G0/0/1 |
| G1/1/2 | fe80:7::2 | | KKK-BR2 | G0/0/1 |
| Port1 (G1/0/1-G1/0/2) | fe80:10::2 | | KKK-MLS1 | Port1 (G1/0/1-G1/0/2) |
| Lo0 | 2001:db8:c1c1:abfe::6/128 | fe80:abfe::6 | - | |
| VLAN 1147 | 2001:db8:c1c1:abb1::2/64 | fe80:abb1::2 | - | |
| VLAN 1173 | 2001:db8:c1c1:abf8::2/64 | fe80:abf8::2 |
| VLAN 1171 | 2001:db8:c1c1:abba::2/64 | fe80:abba::2 |
| VLAN 1197 | 2001:db8:c1c1:abf6::2/64 | fe80:abf6::2 |
| VLAN 1119 | 2001:db8:c1c1:ab8d::2/64 | fe80:ab8d::2 |
| VLAN 20 | 2001:db8:c1c1:ab20::2/64 | fe80:ab20::2 |
| VLAN 37 | 2001:db8:c1c1:ab8a::2/64 | fe80:ab8a::2 |
| VLAN 41 | 2001:db8:c1c1:abda::2/64 | fe80:abda::2 |
| VLAN 13 | 2001:db8:c1c1:abfa::2/64 | fe80:abfa::2 |
| VLAN 69 | 2001:db8:c1c1:abb5::2/64 | fe80:aba5::2 |
| KKK-SW1 | VLAN 37 | 2001:db8:c1c1:ab8a::3/64 | fe80:ab80::3 | - | |
| KKK-SW2 | VLAN 37 | 2001:db8:c1c1:ab8a::4/64 | fe80:ab80::4 | - | |
| DEB-KKK-SW1 | VLAN 37 | 2001:db8:c1c1:c1e3::3/64 | fe80:c1e3::3 | - | |
| SZE-KKK-SW1 | VLAN 37 | 2001:db8:c1c1:ba38::3/64 | fe80:ba38::3 | - | |

19. táblázat – IPv6 címek: kapcsolók

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Hosztok** | | | | | |
| **Eszköz** | **Interfész** | **IPv6 cím** | **Link local** | **Másik eszköz** | **Interfész** |
| KKK-LIN | ens33 | 2001:db8:c1c1:abfa::5/64 | fe80:abfa::5 | KKK-SW2 | G1/0/1 |
| KKK-ADDS | Ethernet1 | 2001:db8:c1c1:abfa::10/64 | fe80:abfa::10 | KKK-SW1 | G1/0/2 |
| KKK-RODC | Ethernet1 | 2001:db8:c1c1:abfa::11/64 | fe80:abfa::11 | KKK-SW1 | G1/0/1 |
| KKK-VPN | ens33 | 2001:db8:c1c1:ab22::2/64 | fe80:ab22::2 | KKK-ASA | G0/5 |
| KKK-BACKUP | ens33 | 2001:db8:c1c1:abfa::7/64 | fe80:abfa::7 | KKK-SW2 | G1/0/2 |
| DEB-KKK-WIN | Ethernet1 | 2001:db8:c1c1:c1ca::3/64 | fe80:c1ca::3 | DEB-KKK-SW1 | G1/0/40 |
| SZE-KKK-WIN | Ethernet1 | 2001:db8:c1c1:baba::3/64 | fe80:baba::3 | SZE-KKK-SW1 | G1/0/40 |

20. táblázat – IPv6 címek: hosztok

2.1.6 VTP

A központban a switchek között a VTP (VLAN Trunking Protocol) használatával történik a VLAN konfigurációk központi kezelése. A KKK-MLS1 a központi VTP szerver, amely biztosítja, hogy a VLAN információk szinkronban legyenek az összes kapcsolón a központi hálózaton. A VTP használata jelentősen csökkenti a hibalehetőségeket, mivel a VLAN-ok létrehozása és módosítása csak a központi eszközön történik.

A KKK-MLS1 VTP szerverként van konfigurálva, míg a többi eszköz VTP kliensként működik, így azok nem tudják módosítani a VLAN konfigurációkat. A VTP konfigurálása egyszerűsíti a VLAN-ok kezelését, mivel minden módosítás automatikusan továbbítódik az összes érintett eszközhöz. Az eszközök közötti kommunikáció VTP adatcsomagok segítségével történik, amelyeket minden kapcsoló fogad és továbbít.

A VTP domain és verzió megfelelő beállítása elengedhetetlen a hálózatban történő zökkenőmentes VLAN kezeléshez. A VTP tartományunk a kkk.com és a 2-es verziót használtuk. A jelszó a következő: **voWPmRDiTM**

2.1.7 STP

A STP (Spanning Tree Protocol) célja, hogy megakadályozza a hurok problémák (looping) kialakulását a hálózatban úgy, hogy egyetlen aktív adatátviteli utat választ a redundáns kapcsolatok közül. Ez a protokoll biztosítja, hogy a hálózaton ne legyenek ciklikus utak, amelyek folyamatos adatforgalmat generálnának, ami a hálózat meghibásodásához vezethetne. A hálózat jelenlegi állapotában nincs hurok, de fel kell készülnünk arra, hogy a jövőbeli bővítés során kialakulhat benne.

Az STP a Root Bridge (gyökérkapcsoló) az összes többi eszköz számára alapként szolgál, mivel Root Bridge-hez kapcsolódó portok lesznek azok, amelyeken keresztül a forgalom irányítható. Gyökérkapcsolóként VLAN-onként eltérően vagy a KKK-MLS1, vagy a KKK-MLS2 lett konfigurálva. A prioritással a fiókirodákban nem foglalkoztunk, mivel ott csak egy-egy kapcsoló található. A *21. táblázatban* látható, hogy melyik harmadik rétegbeli kapcsoló, melyik VLAN-ban gyökérponti híd, vagyis melyik VLAN STP feszítőfájában kapta meg a primary prioritást. Minden más VLAN-ban az adott MLS kapcsoló a secondary prioritást kapta.

|  |  |
| --- | --- |
| **STP gyökérponti hidak** | |
| **MLS kapcsoló** | **VLAN neve** |
| KKK-MLS1 | VEZETO  MANAGEMENT  SRV  IT |
| KKK-MLS2 | ADMINISZTRACIO  MARKETING  KIADO  ZENE  WIFI  BIZTONSAG |

21. táblázat – STP: gyökérponti hidak

A gyors konvergencia érdekében és azért, hogy külön gyökérponti hidakat tudjunk létrehozni a VLAN-okban, rapid-pvst módot alkalmaztuk. A konvergencia gyorsítása érdekében az összes access típusú porton engedélyeztük a PortFast-et.

2.1.8 Második rétegbeli redundancia

A második rétegbeli redundancia célja, hogy biztosítsa a hálózaton belüli folyamatos adatforgalmat akkor is, ha egyes kapcsolatok meghibásodnak.

2.1.8.1 LACP

Az LACP (Link Aggregation Control Protocol) lehetővé teszi több hálózati kapcsolat kombinálását egyetlen logikai kapcsolattá, amely megnöveli az elérhető sávszélességet és növeli a hálózati redundanciát. A terheléselosztás az alapértelmezett beállítás, vagyis a forrás MAC-cím alapján történik. Az LACP dinamikusan kezeli az aggregált kapcsolatokat, és automatikusan figyeli azok állapotát. Ha bármelyik link meghibásodik, azonnal eltávolítja azt az aggregált csoportból, és a forgalom továbbra is zavartalanul haladhat a fennmaradó működő linkeken.

A kapcsolatokat az eszközök közötti LACP üzenetek segítségével alakítják ki, és a protokoll folyamatosan ellenőrzi azok állapotát, biztosítva a legnagyobb sávszélesség elérését és a legnagyobb rendelkezésre állást.

A hálózatban a két központi switch (KKK-SW1, és KKK-SW2) között használtuk. A KKK-SW1 aktív, a KKK-SW2 pedig passzív állapotban van. Az egyes channel group lett használva, így egyes számmal jött létre a port-channel a kapcsolók között.

2.1.9 Harmadik rétegbeli redundancia

A harmadik rétegbeli redundancia célja, hogy biztosítsa az IP alapú forgalom folyamatos elérhetőségét és a hálózati útvonalak redundanciáját. A harmadik rétegbeli redundancia különböző technikákkal, például redundáns útvonalakkal és protokollokkal valósítható meg, amelyek automatikusan kezelik a meghibásodásokat, és biztosítják a legjobb elérhetőséget.

2.1.9.1 Redundáns kapcsolatok

A redundáns kapcsolatok használata biztosítja, hogy ha az egyik útvonal meghibásodik, akkor a hálózati eszközök automatikusan átváltanak egy másik, elérhető útvonalra. A redundáns kapcsolatok révén a forgalom folyamatosan halad, és nem fordul elő hálózati kiesés.

A rendszer dinamikus útválasztási protokollokat használ, hogy figyelemmel kísérje a hálózati topológia változásait, és ha bármelyik útvonal elérhetetlenné válik, azonnal új útvonalat választ. Központban találhatóak redundáns kapcsolatok az MLS kapcsolók között, az MLS kapcsolók és a BR forgalomirányítók között, valamint a BR routerek között.

2.1.9.2 HSRP

A HSRP (Hot Standby Router Protocol) lehetővé teszi a hálózat számára, hogy egyetlen virtuális IP címet használjon, miközben két vagy több útválasztó működik együtt. A HSRP biztosítja, hogy ha az aktív útválasztó meghibásodik, a forgalmat automatikusan átirányítja egy standby útválasztóra, ezzel biztosítva a folyamatos elérhetőséget.

Mind a három telephelyen lett alkalmazva a HSRP a redundancia elérése miatt. A terhelés elosztás miatt alhálózatonként más-más forgalomirányító az aktív. Az aktív routerek prioritásai 109-re lettek állítva a fiókirodákban, a központban pedig 119-re. A *22. táblázatban* látható, hogy melyik VLAN-ban melyik router az aktív.

|  |  |
| --- | --- |
| **HSRP aktív routerek** | |
| **Átjáró neve** | **VLAN neve** |
| KKK-MLS1 | VEZETO  MANAGEMENT  SRV  IT |
| KKK-MLS2 | ADMINISZTRACIO  MARKETING |
| **Átjáró neve** | **VLAN neve** |
| KKK-MLS2 | ZENE  WIFI  BIZTONSAG |
| SZE-KKK-BR1 | FIOKVEZETO  MANAGEMENT  SRV  IT |
| SZE-KKK-BR2 | TEHETSEG  KIADO  WIFI  BIZTONSAG |
| DEB-KKK-BR1 | FIOKVEZETO  MANAGEMENT  SRV  IT |
| DEB-KKK-BR2 | KOZONSEG  ESEMENY  WIFI  BIZTONSAG |

22. táblázat – HSRP: aktív routerek

A virtuális IPv4 cím az adott alhálózat első használható címe, a virtuális IPv6 cím pedig az ott használt link-local cím ffff hoszt résszel. A kettes verzió lett választva annak érdekében, hogy az IPv4-es group számok megegyezhessenek a VLAN számokkal és hogy IPv6-os csoportokat is létrehozhassunk. Az IPv6-os group számok eggyel többek, mint az IPv4-es párjuk. Az MLS kapcsolóknál a BR routerekhez, a fiokirodákban pedig a BR routereken az IR routerhez kapcsolódó interfészek trackelve vannak. Egy-egy interfésze lekapcsolódása 10-zel csökkenti a prioritást. Ahhoz, hogy ebben az esetben a standby átvegye az aktív szerepet a preemption is be lett állítva.

2.1.9.3 Harmadik rétegbeli EtherChannel

A központban a KKK-MLS1 és KKK-MLS2 között a redundancia erősítése miatt harmadik rétegbeli EtherChannel lett kialakítva. A KKK-MLS1 aktív, a KKK-MLS2 pedig passzív állapotban van. Mindkét MLS kapcsolón az 1-es csoport lett alkalmazva. Mindkét kapcsolón az egyes port-channel kapta az IPv4 és IPv6 címet.

2.1.10 OSPFv2, OSPFv3

Az OSPF (Open Shortest Path First) egy dinamikus útválasztási protokoll, amelyet leginkább nagyobb, IP alapú hálózatokban alkalmaznak. Az OSPF link state alapú, és biztosítja a hálózati eszközök közötti hatékony adatforgalmat és gyors hibaelhárítást. Az OSPF segítségével gyorsan lehet alkalmazkodni a hálózatban történő változásokhoz, biztosítva ezzel a megbízhatóságot és az elérhetőséget. Az összes forgalomirányítón a OSPFv2 esetén a 100-as, OSPFv3 esetén pedig a 200-as folyamat azonosító lett alkalmazva. Az IR routerek mindkét verzióban hirdetik az alapértelemezett átjárót. Valamint ezeken a routereken az OSPF bele van forgatva a DMVPN-enen működő EIGRP-be. A fiókirodákban csak area 0 van, addig a központban az area 5-ben a két MLS kapcsoló, az area 10-ben az ASA, area 0-ban pedig a KKK-BR-1, a KKK-BR-2 és a KKK-IR forgalomirányító van.

2.1.11 Biztonság

A cég kérésére a hálózat biztonságával is foglalkoztunk.

2.1.11.1 Fizikai biztonság

A központban a helyiség, ahol a szerverek és a hálózati eszközök találhatóak, az épület legfelső szintjén van és csak az IT szakember irodáján keresztül lehet bemenni, hogy minél nehezebben elérhető és felfedezhető legyen egy támadó számára. Az épületbe csak a biztonsági őr által beengedett, ott dolgozó személyek mehetnek be munkaidőben. Beléptető rendszer nem üzemel, mivel az alacsony létszámú személyzet miatt feleslegesnek lett ítélve. Kamerák találhatóak az összes szinten. A szerverekhez és a hálózati eszközökhöz, valamint az IT szakember irodájába csak az ott dolgozó IT szakember és a vezetők mehetnek be arc- és ujjlenyomat azonosítás, és egy hetente változó, 20 karakter hosszú jelszó megadása után.

A többi fiókirodában csak a fiókvezetők irodáján keresztül lehet bemenni a szerverekhez és a hálózati eszközökhöz. Hasonló biztonsági intézkedések vannak, mint a központban. A különbség, hogy nincs biometrikus azonosítás. Csak egyszer használható, 20 karakter hosszú jelszavak vannak, amelyet a központban lévő IT szakember mondhat meg a fiókvezetőnek egyszerű hibák megoldására. Nagyobb hiba esetén az IT szakember személyes megjelenése szükséges. Ezeket a kamerákat és a szervertermekbe való belépést ellenőrző eszközöket egy másik cég (Patent Cégcsoport**)** telepítette és üzemelteti.

2.1.11.2 Hitelesítés

A hálózati eszközök kis száma miatt, valamint azért, mert csak egyetlen ember, az IT szakember férhet hozzá ezekhez, ezért ezeken a helyi adatbázis alapú hitelesítés lett választva egy központi AAA szerver helyett. Mindegyik eszközön jelszó került konfigurálásra a privilegizált EXEC módba való belépéshez. A felhasználónév minden eszköz konzol és VTY vonalán azonos. A jelszavak és a felhasználónevek minden eszközön megegyeznek, és a jelszavak mindegyiken titkosítva vannak tárolva. Mindegyik eszközön létre lett hozva egy kosa nevű felhasználó arra az esetre, ha távolról valamiért nem lehetne elérni az adott eszközt. Mivel ezzel a felhasználóval is be lehet jelentkezni a VTY vonalon, ezért a biztonság érdekében ez a felhasználó a 10-es privilege szintet kapta. A kosa felhasználóval privilegizált exec módban a configure terminal, globális konfigurációs módban az interface, valamint interfész konfigurációs módban a shutdown, a no shutdown, és az ip address parancsot lehet kiadni. Illetve még a kapcsolóknál a portokat lehet access típusúra állítani, valamint hozzárendelni a 37-es, MANAGEMENT VLAN-hoz. Erre azért volt szükség, hogy ha a trunk port romlik el, akkor is el lehessen érni a management VLAN-t. Tehát ezzel a felhasználóval helyre lehet állítani egy meghibásodott interfészt, hogy után el lehessen érni az eszközt VTY vonalon. A weboldalon keresztüli konfigurációhoz pedig egy kovacs.gabor nevű felhasználó lett létrehozva. A jelszó és a felhasználónév a *23. táblázatban* látható.

|  |  |  |
| --- | --- | --- |
| **Jelszavak** | | |
| **Típus** | **Felhasználónév** | **Jelszó** |
| enable | **-** | 9@PqZ$7sRfT!1a |
| konzol | kosa  kovacs.gabor | bxJ<5EMJegS;U+O]4={T  ECh5Gy5sbHw |
| VTY |
| admin | **-** |

23. táblázat - Jelszavak

2.1.11.3 Belépés próbálkozások száma

A brute force támadások elleni védelem érdekében korlátozva lett egy megadott időn belüli belépés próbálkozások száma. A konfigurálás alapján 1 percen belüli 3 helytelen bejelentkezés után 10 percig nem lehet bejelentkezni.

2.1.11.4 Inaktív időkorlát

Ha a hitelesített felhasználók inaktívak, akkor egy idő után automatikusan ki lesznek jelentkeztetve. Erre azért van szükség, hogy illetéktelen felhasználók ne tudják használni ezeket a nyitva maradt bejelentkezéseket. A beállítások szerint egy bejelentkezés nyolc percig lehet inaktív.

2.1.11.5 Webes felület

Az ip http server és az ip http secure-server szolgáltatások kikapcsolásával megszűnt a forgalomirányítók és kapcsolók web-alapú kezelésének lehetősége. Mivel a parancssori kezelés kerül alkalmazásra a hálózati eszközök konfigurálásához, ezek a szolgáltatások feleslegessé váltak. A web-alapú kezelés növelné a hálózati eszközökhöz való jogosulatlan hozzáférés kockázatát, és a támadási felületet. Minden forgalomirányítón és kapcsolón kikapcsolásra került a HTTP és a HTTPS szerver.

2.1.11.6 SSH

A biztonság növelése érdekében a hálózati eszközöket Telnet helyett csak a forgalmat titkosító SSH-n keresztül lehet VTY vonalon elérni. A nagyobb biztonság érdekében az SSH 2-es verziója került beállításra, és 4096 bit hosszú RSA kulcsokat használ. A felhasználó hitelesítése az IT szakember számítógépén generált, 2048 bit hosszú aszimmetrikus kulcspár használatával történik. Ez a megoldás biztonságosabb, mint egy sima jelszó, mivel ez egy hosszabb, random karakterekből álló string, illetve csak az IT szakember gépén van eltárolva, nem úgy, mint a sima jelszó, ami az összes hálózati eszközön. A publikus kulcs minden forgalomirányitón manuálisan lett eltárolva, a privát kulcs és a felhasználónév pedig az IT szakember gépén található PuTTY nevű szoftverben el lett mentve egy új, Network\_Devices nevű session-be. Így az IT szakembernek csak a konfigurálni kívánt eszköz IP címét kell beírnia. A publikus és a privát kulcs pedig a *C:\SSH\keys* mappában van elmentve az IT szakember számítógépén public\_key.pub és private\_key.ppk néven. A kulcsok a PuTTY key generator nevű alkalmazással lettek generálva. Az *15-16. képen* látható a PuTTY-ban elmentett privát kulcs elérési útvonala és felhasználónév.

A képen szöveg, képernyőkép, képernyő, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

15. kép – PuTTY: privát kulcs elérési útvonalának elmentése

A képen szöveg, képernyőkép, képernyő, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

16. kép – PuTTY: felhasználónév elmentése

Illetve a *17. képen* példaként az SSH konfigurációja a SZE-KKK-BR1-n.

A képen szöveg, képernyőkép, Betűtípus, menü látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

17. kép – SZE-KKK-BR1-n SSH konfiguráció

2.1.11.7 Management VLAN

A kapcsolókon nem az alapértelmezett, 1-es VLAN lett konfigurálva management VLAN-ként, hanem a 37-es. Ezáltal megszűnt annak a veszélye, hogy a kapcsoló konfigurálása során valamelyik 1-es VLAN-ban maradt vagy került interfészhez csatlakozva valaki könnyedén, forgalomirányitó nélkül elérje a kapcsolók üzemeltetési síkját. Emellett meg lett nehezítve a hálózat felderítése a támadó számára, illetve csökkentve lett a VLAN ugrásos támadás sikeressége.

2.1.11.8 VLAN

Az alapértelmezett, 1-es VLAN nem lett alkalmazva a hálózat szegmentálása során. Ezzel meg lett nehezítve a hálózat felderítése a támadó számára, illetve csökkentve lett a VLAN ugrásos támadás sikeressége. Emellett megszűnt annak a veszélye, hogy a kapcsoló konfigurálása során valamelyik 1-es VLAN-ban maradt vagy került interfészhez csatlakozva valaki könnyedén, forgalomirányitó nélkül elérjen egy használatban lévő VLAN-ban található eszközöket.

2.1.11.9 Nem használt portok

Minden nem használt port le lett kapcsolva, access módba lett állítva, és a nem használt (black hole) 29-es VLAN-ba került. Így csak a használt portokon történhet forgalom. Ezzel csökkentve lett a különféle támadások, például a VLAN ugrás, a MAC-cím elárasztás, az ARP hamisítás, valamint a DHCP-hamisítás, vagy a DHCP-kiéheztetés esélye. Azonban ezek a veszélyek továbbra is fennállnak az aktív portokon, ezért ezeken további konfigurációkra van szükség.

2.1.11.10 Portbiztonság

A MAC-cím elárasztás és a DHCP-kiéheztetés ellen portbiztonság lett beállítva a kapcsolók access típusú interfészein, amikhez a végpontok csatlakoznak. A portbiztonság maximum 1 MAC-címet tanul meg sticky módszerrel, ami biztosítja, hogy egy megadott eszközöktől származó forgalomra korlátozódjon a hozzáférés a portokon. A legerősebb biztonságot nyújtó, alapértelmezett megsértési mód, a shutdown beállításával a kapcsolók leállítják a portot, ha eltérést talál az üzenet forrás MAC-címe és a megtanult cím között. A hálózat üzembe helyezése után minden switch futó konfigurációja el lett mentve az indító konfigurációba. Ezáltal biztosítva lett, hogy a kapcsolók csak az előre telepített végpontok MAC-címeit fogadják el újraindításuk után is.

2.1.11.11 Trunk portok

A kapcsolók trunk portjain ki lett kapcsolva a Dynamic Trunking Protocol (DTP) funkció, és be lett állítva, hogy csak az előre létrehozott VLAN-ok forgalmait engedjék át. Ezen kívül meg lett változtatva a native VLAN az alapértelmezett, 1-es VLAN-ról a nem használt 11-es VLAN-ra. Ezzel jelentősen csökkentve lett a VLAN ugrásos támadás kockázata.

2.1.11.12 DHCP Snooping

A DHCP Snooping kapcsolókon való beállításával lett megoldva a védekezés a DHCP-támadások ellen. A DHCP-hamisítás elleni védekezés érdekében a DHCP Snooping funkcióban a kapcsolók trönk portjait, valamint a DHCP-szerverhez csatlakozó portot megbízható (trusted) portként lettek beállítva. Ezzel biztosítva lett, hogy csak az előre telepített DHCP-szerver küldhessen DHCP válaszüzeneteket (DHCPOFFER, DHCPACK, DHCPNAK). A DHCP-kiéheztetés veszélye a nem megbízható (untrusted) portokon a DHCP-üzenetek korlátozásával lett csökkentve. A fogadható üzenetek száma négyre lett korlátozva. A DHCP Snooping minden használt VLAN-ban konfigurálva lett.

2.1.11.13 Dinamikus ARP-ellenőrzés

Az ARP-mérgezés és az ARP-hamisítás ellen a dinamikus ARP-ellenőrzés (DAI) lett konfigurálva a kapcsolókon. A DHCP Snoopinghoz hasonlóan lett beállítva, vagyis csak a szerverekhez csatlakozó és a trönk portok lettek megbízhatók. A nem megbízható portokon a forrás- és cél MAC- és IP-címeket a DHCP Snooping táblája alapján ellenőrzi, és eltérés esetén eldobja az ARP-üzenetet. A DAI minden használt VLAN-ban konfigurálva lett.

2.1.11.14 BPDU Guard

A feszítőfa elleni támadás különféle veszélyeket rejt magában, mint például közbeékelődést vagy szórási vihar keletkezését, ami az egész kapcsolt hálózat leállásához vezethet. Ez a BPDU Guard konfigurálásával lett kiküszöbölve, amely minden access módban és PortFast-tal konfigurált porton be lett állítva. Így, ha ezen portok valamelyikén egy BPDU érkezik, az a port lekapcsolt állapotba kerül.

2.1.11.15 Root guard

A hálózat stabilitása és a feszítőfa protokoll megfelelő működésének érdekében a root guard funkció lett alkalmazva a multilayer switcheken. Bár a hálózatban jelenleg nincsen második rétegben hurok, viszont az esetleges jövőbeli bővítések során keletkezhet. Biztonsági szempontból a root guard használata azért fontos, mert megakadályozza, hogy egy rosszindulatú vagy hibás konfigurációjú eszköz gyökérponti hídként vegye át az irányítást a Spanning Tree protokoll felett. A másodi rétegben levő kapcsolóknál ez a beállítás nem lett alkalmazva, mivel az access típusú portjait a BPDU guard védi. Az egymást összekötő trunk portjain lehetne alkalmazni, de ha hurok keletkezik a hálózatban a későbbi bővítések miatt, akkor a kapcsolók ezen portjain is érkezhet BPDU a gyökérponti hídtól. Ekkor ezen a porton megszűnne a bejövő keretek továbbítása, illetve folyamatosan syslog üzenetet kapnánk. Mindkét harmadik rétegbeli kapcsoló összes használt és nem használt portján be lett állítva a root guard, hogy a jövőben is biztosítva legyen, hogy minden VLAN-ban az adott harmadik rétegbeli kapcsoló maradjon a gyökérponti híd.

2.1.11.16 CDP letiltása

A Cisco Felderítő Protokoll hasznos hálózati eszközök konfigurálásánál és hibaelhárításánál, viszont CDP-üzenetek elfogásával a támadó fontos információkat tudhat meg a hálózatról. A CDP előnyös a hálózati hibák elhárításához, ezért egy köztes megoldásként CDP-üzenetek küldése csak a hálózati eszközök között engedélyezett, és a végpontok fele pedig tiltott. Mivel a CDP alapvetően engedélyezett minden interfészen, ezért a végpontokhoz csatlakozókon a no cdp enable parancs lett kiadva.

2.1.11.17 ACL-ek

A hálózati eszközöket távolról, VTY vonalon csak az IT szakember alhálózatából lehet elérni. Ezért minden hálózati eszközön létre lett hozva egy VTYv4 nevű normál, nevesített és egy VTYv6 nevű kiterjesztett, nevesített ACL, amely a VTY vonalakon csak az IT szakember alhálózatából származó, bemenő forgalmat engedélyezi, és minden mást tilt. Mindegyik telepen mindegyik IPv4-es IT alhálózatból származó forgalmat engedélyezi, így a 10.11.169.0/28, a 10.23.169.0/29 és a 10.37.169.0/29 alhálózatból lehet elérni a hálózati eszközöket távolról. Azért van szükség az összes privát IPv4-es alhálózat engedélyezésére, mivel az interneten keresztül is el tudja érni az IT szakember a más telephelyen lévő hálózati eszközöket a tunneling-nek köszönhetően. Az IPv6 esetében is mindegyik IT alhálózatból (2001:db8:c1c1:abb5::/64, 2001:db8:c1c1:bab3::/64, 2001:db8:c1c1:c1b5::/64) mindegyik épületben lévő mindegyik hálózati eszközt el lehet érni. Az IPv6-os ACE-k célja any. Emellett a KKK-LIN szerver is elérheti a hálózati eszközöket a webszerver szolgálatás megfelelő működéséhez.

A management VLAN-okba az IT szakember alhálózatából és a KKK-LIN szervertől származó forgalom engedélyezett. Ez a feltétel a MANAGEMENTv4 nevű normál, nevesített és MANAGEMENTv6 nevű kiterjesztett, nevesített ACL-ekben lett megfogalmazva, ami a

* KKK-MLS1 VLAN 11
* KKK-MLS2 VLAN 11

interfészén, és a

* DEB-KKK-BR1 G0/0/0.11
* DEB-KKK-BR2 G0/0/0.11
* SZE-KKK-BR1 G0/0/0.11
* SZE-KKK-BR2 G0/0/0.11

alinterfészén a kimenő forgalmat szűri. A forrás alhálózatok ugyanúgy lettek megadva, mint a távoli elérést korlátozó ACL-eknél. Az IPv6-os ACE-k célja any.

Az IT szakember alhálózatát semelyik másik, a cégben dolgozók számára létrehozott alhálózatokból semmilyen protokollon keresztül nem lehet elérni. Csak a szerverek és a többi IT alhálózatból, illetve a management VLAN-ból származó forgalom engedélyezett az IT szakember alhálózata felé. Ezt a ITv4 nevű kiterjesztett, nevesített ACL-ek biztosítják, ami a

* KKK-MLS1 VLAN 69
* KKK-MLS2 VLAN 69

interfészén, és a

* DEB-KKK-BR1 G0/0/0.69
* DEB-KKK-BR2 G0/0/0.69
* SZE-KKK-BR1 G0/0/0.69
* SZE-KKK-BR2 G0/0/0.69

alinterfészén a kimenő forgalmat szűri. Az ACE-k célja any. Az ACL-ek csak IPv4-es címeket szűrnek, mivel IPv6-os címmel bárhonnan el lehet érni az alhálózatot és mindent nem lehet tiltani. Azonban az IT szakembernek el kell érnie a többi VLAN-ban lévő számitógépet SSH, RDP vagy ICMP protokollon keresztül. Ezek válasz üzeneteit viszont tiltja az ACL, de ezen protokollok átengedése biztonsági kockázat. Ezért a probléma a következő ACE használatával lett megoldva, ami az ACL elején található és ami engedélyezi a TCP válaszforgalmat:

permit tcp any any established

A vezeték nélküli routerek alhálózatából származó forgalom nem érheti el a többi alhálózatot semmilyen protokollon keresztül, de minden más forgalom engedélyezett. Mivel egyik alhálózatot sem érhetik el, ezért az ACL rövidítése érdekében az egész 10.0.0.0/8 és 2001:db8:c1c1::/48 hálózat elérése lett tiltva. Ezt a WIFIv4 és WIFIv6 nevű kiterjesztett, nevesített ACL-ek biztosítják, ami a

* KKK-MLS1 VLAN 41
* KKK-MLS2 VLAN 41

interfészén, és a

* DEB-KKK-BR1 G0/0/0.41
* DEB-KKK-BR2 G0/0/0.41
* SZE-KKK-BR1 G0/0/0.41
* SZE-KKK-BR2 G0/0/0.41

alinterfészén a bemenő forgalmat szűri. Az összes ACE forrása any. Azonban a Wi-Fi hozzáférés hitelesítése a központi RADIUS szerver biztosítja, illetve a vezetéknélküli forgalomirányítóhoz csatlakozó eszközök a központi DHCP szervertől kapnak IP címet. Ezért a tiltások előtt engedélyezve lett, hogy az adott szervereket el lehet érni DHCP (67-es UDP port) és RADIUS (1812-es és 1813-as UDP port) protokollal.

Mivel a KKK-MLS1 harmadik rétegbeli kapcsolón mindegyik előbb említett ACL létre lett hozva, ezért példaként a *18. képen* látszódnak a létrehozott ACL-ek.

A képen szöveg, képernyőkép látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

18. kép – ACL-ek az KKK-MLS1-en

2.1.11.18 Unicast Reverse Path Forwarding (uRPF)

Minden telephelyen a kapcsolókhoz csatlakozó forgalomirányítók, valamint a harmadik rétegbeli kapcsolók esetében uRPF került konfigurálásra. Ez a beállítás a bemenő forgalmat szűri a forgalomirányítók alinterfészein, illetve a harmadik rétegbeli kapcsolók virtuális interfészein. A csomagok forrás IP-címének érvényességét azért kell ellenőrizni, hogy megakadályozzuk a hamis IP-címekkel végrehajtott túlterheléses támadásokat. Az IP-címhamisítás problémája egyrészt az az, hogy az ilyen forgalom valódi forrása nem követhető vissza, ami megnehezíti a támadások kivédését. Ezenfelül az IP-címhamisítás lehetővé teheti ACL szabályok megkerülését is. Például az IT-szakember VLAN-ját egyik belső alhálózatból sem lehet elérni, viszont egy támadó mégis hozzáférhet, ha egy hamis forrás IP-címet használ. Ezek a biztonsági kockázatok indokolták a strict módú uRPF konfigurálását. Az uRPF beállítások azért a szélső forgalomirányítókra, illetve a harmadik rétegbeli kapcsolókra kerültek, hogy a hamisított csomagok feleslegesen ne jussanak tovább a hálózat belső részeire. Az IPv4-es és az IPv6-os csomagok is szűrve vannak.

2.1.11.19 WLAN

A cégben található vezeték nélküli routereken a cég dolgozói számára létrehozott Wi-Fi SSID szórása le van tiltva, így azt az ott dolgozóknak manuálisan kell beállítaniuk, ha kapcsolódni szeretnének a hálózathoz a vezeték nélküli eszközeikkel. Az egyik legmodernebb biztonsági módszer, a WPA2-Enterprise lett engedélyezve. Az AD DS és a RADIUS szerver integrációja lett megvalósítva a Wi-Fi hozzáférés központi hitelesítésére. A felhasználók csatlakozhatnak a hálózathoz a tartomány-hitelesítő adataik használatával. A vezeték nélküli routerek külön VLAN-ban vannak, így a vezeték nélküli eszközöktől származó forgalom jól elkülönül a hálózat többi forgalmától.

A vendégek számára vendég Wi-Fi lett létrehozva, ahol szintén a WPA2-as biztonsági módszer van alkalmazva. A vendégek a tárgyalóban és a konferenciateremben lehelyezett papíron olvashatják a szükséges jelszót a vendég Wi-Fi-hez való csatlakozáshoz.

2.1.11.20 OSPFv2 és OSPFv3 hitelesítés

Az OSPF dinamikus útválasztásának védelmére mindkét verzióban hitelesítés lett beállítva, hogy a támadó az OSPF csomagok megváltoztatásával, vagy hamisak küldésével ne tudja megváltoztatni a forgalomirányítást és ezáltal ne téríthesse el a hálózat forgalmát. Az OSPFv2-nél az egyszerű, szöveges, titkosítatlan jelszó helyett az MD5 kriptográfia hash algoritmus van használva a biztonságosabb hitelesítésre és a csomag sértetlenségének ellenőrzésére. Mindegyik telephelyen minden terület egyedi kulcs és kulcs azonosító párost kapott. Ezek a párosok az *24. táblázatban* láthatóak.

|  |  |  |  |
| --- | --- | --- | --- |
| **OSPFv2 hitelesítés** | | | |
| **Telephely** | **Terület** | **Kulcs azonosító** | **Kulcs** |
| Budapest | 0 | 84 | G7pL2qR4sW5kX9zB |
| 5 | 138 | hT8sR0vJ3pW4nY6x |
| 10 | 196 | Q9tM1aE5fV6kL2zY |
| Debrecen | 0 | 121 | A9hbL3kmX457qZ1p |
| Szeged | 0 | 92 | Z8ggJ2evK053dQ9m |

24. táblázat - OSPFv2 hitelesítés

A hálózatban az OSPFv3 hitelesítés IPsec alapú, mely minden OSPFv3 területen egyedi SPI és megosztott kulcs használatával biztosítja a kapcsolatok biztonságát. A nagyobb biztonság érdekében mindegyik telephely mindegyik területéhez egyedi SPI és kulcs páros lett konfigurálva. Így, ha egy terület hitelesítését feltörik, akkor a támadás csak az adott területre tud koncentrálódni. Az SHA-1 hash algoritmus lett alkalmazva. Az SPI és kulcs párosok az *25. táblázatban* találhatóak.

|  |  |  |  |
| --- | --- | --- | --- |
| **OSPFv3 hitelesítés** | | | |
| **Telephely** | **Terület** | **SPI** | **Kulcs** |
| Budapest | 0 | 2008 | 3F2A1B8C9D1E4F6A7B8C9D0E1F2A3B4C5D6E7F8A |
| 5 | 12075 | 7A4B1C8D9E2F6A1B8D9E4C5F1A2B3C4D5E6F7A8B |
| 10 | 21478 | 1E2A3B4C5D6E7F8A9B0C1D2E3F4A5B6C7D8E9F0A |
| Debrecen | 0 | 905 | 0A1B2C3D4E5F6A7B8C9D0E1F2A3B4C5D6E7F8A9B |
| Szeged | 0 | 5641 | 9F8E7D6C5B4A3C2D1E0F9A8B7C6D5E4F3A2B1C0D |

25. táblázat - OSPFv3 hitelesítés

A *19. képen* látható, hogy a SZE-KKK-IR-on az OSPFv2 és az OSPFv3 hitelesítés már konfigurálva lett, ezért a SZE-KKK-BR1 a SZE-KKK-IR hello üzeneteit nem tudja fogadni, mivel a hitelesítés még ott nem lett beállítva, ezért lejár a dead időzítő és a szomszédság megszűnik. A SZE-KKK-BR1 konfigurálása után viszont újra helyre áll a szomszédsági viszony.

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

19. kép – OSPFv2 és OSPFv3 hitelesítés konfigurálása és tesztelése a SZE-KKK-BR1 routeren

2.1.11.21 EIGRP hitelesítés

Az EIGRP által küldött forgalomirányítási információk védelmére hitelesítés lett beállítva. Az egyszerű, szöveges, titkosítatlan jelszó helyett MD5 kriptográfia hash algoritmus van használva a biztonság erősítésére és a csomagok sértetlenségének ellenőrzésére. Mivel az EIGRP-nél kötelező a key-chain használata, ezért létre lett hozva egy KEYEIGRP néven. A kulcs a huszonötös azonosítót kapta és 2024 szeptember elsejétől végtelen ideig érvényes. Az IPv6-os és az IPv4-es szomszédok is ugyanazzal a kulccsal hitelesítik egymást. A kulcs, ami konfigurálásra került a következő: **F7#h8m$Q2!xK4pL**

A *20. képen* látható, hogy a KKK-IR routeren már konfigurálásra került az EIGRP hitelesítés, de a SZE-KKK-IR-en még nem, ezért a szomszédság megszűnik. Miután a SZE-KKK-IR forgalomirányító is konfigurálva lett, újra létre jön az IPv4-es és az IPv6-os EIGRP szomszédság.

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

*20. kép – EIGRP hitelesítés konfigurálása és tesztelése a SZE-KKK-IR routeren*

2.1.11.22 BGP hitelesítés

A BGP hitelesítés a forgalomirányítási információk védelmére szolgál. Az egyszerű, szöveges, titkosítatlan jelszó helyett az MD5 hash algoritmus van használva a biztonság erősítésére és a csomag sértetlenségének ellenőrzésére. IPv4-nél és IPv6-nál ugyanazok a kulcsok lettek alkalmazva, viszont a kulcsok eltérnek telephelyenként. A kulcsok a *26. táblázatban* találhatóak.

|  |  |
| --- | --- |
| **BGP hitelesítés** | |
| **Telephely** | **Kulcs** |
| Budapest | D3s6!aP9v#4@bQ8R |
| Szeged | R#7qT!m4L0s3^N2x |
| Debrecen | 8Y%jLz9!F@1r2Vp# |

26. táblázat - BPGP hitelesítés

A *21. képen* látható példaként a KKK-IR BPG hitelesítésének konfigurációja.

A képen szöveg, Betűtípus, képernyőkép, sor látható

Automatikusan generált leírás

21. kép – KKK-IR-en a BGP hitelesítés konfigurációja

2.1.11.23 Passzív interfész

Az OSPF konfigurálásánál a végpontok felé néző interfészek passzívok lettek. Ez a beállítás növeli a biztonságot az útválasztási információk terjesztésének korlátozásával, csökkentve a potenciális támadási felületet. Nagyon sok passzív interfészt kellett volna beállítani, ezért az egyszerű konfigurálás érdekében a passive-interface default parancs lett alkalmazva és a no passive-interface paranccsal lettek beállítva a nem passzív interfészek.

2.1.11.24 HSRP hitelesítés

A HSRP-vel konfigurált forgalomirányítók közötti üzenetváltások biztonságának érdekében hitelesítés lett beállítva. A nagyobb biztonság elérése és az üzenetek sértetlenségének ellenőrzése érdekében a hitelesítés az MD5 hash algoritmust használja az egyszerű, szöveges, titkosítatlan jelszó helyett. Minden telephelyen más kulcs van használva, de telephelyenként az IPv6-os és IPv4-es csoport ugyanazt a kulcsot használja. Ezek a *27. táblázatban* találhatóak.

|  |  |
| --- | --- |
| **HSRP hitelesítés** | |
| **Telephely** | **Kulcs** |
| Budapest | T7v@2u$M!qW9bZ3^ |
| Szeged | eR5&hX4!jN8p$W1m |
| Debrecen | Y8^fS3\*zZ2@qJ7!k |

27. táblázat - HSRP hitelesítés

2.1.11.25 Direkt kulcsok

A dinamikus forgalomirányító protokollok és a HSRP hitelesítésére direkt kulcsok lettek konfigurálva. Ennek okai a következők:

* Egyszerűbb konfigurálni a hálózati eszközöket direkt kulccsal, így gyorsabb a hálózat telepítése. Emellett nagy mértékben csökken a hibázás esélye a telepítés során.
* Könnyebb nyilvántartani a hitelesítéshez használt kulcsokat. Ez segíti a hálózat karbantartását.
* Nagyon sokféle hitelesítés és azokon belül is sokféle kulcs van alkalmazva, ezáltal sok key-chain kellett volna létrehozni a hálózati eszközökön. Ez rossz hatással lett volna a hálózat teljesítményére.

2.1.11.26 Hardveres tűzfal: ASA

A távoli hozzáférést biztosító szervert, a KKK-VPN-t minél közelebb a hálózat határához helyeztük el, hogy a forgalma minél kevesebb hálózati eszközt érintsen. Nem lehetet közvetlenül a KKK-IR forgalomirányítóhoz kötni, mivel az már elég sok szolgáltatással le van terhelve. Ezért volt szükség még egy eszközre, ami viszont több biztonsági funkciót tud, mint egy router. Így esett a választás a Cisco Adaptive Security Appliance tűzfalra. A tűzfal OSPF-fel hirdeti a hálózatait.

A kliens és a VPN szerver közötti forgalom titkosított, ezért a VPN szerverből kimenő forgalom van ellenőrizve. Ennek megfelelően az ASA KKK-VPN felé néző, G0/5 interfésze az outside nevet kapta, a KKK-IR felé néző, G0/8 pedig az inside nevet. Így a G0/5 biztonsági szintje 0, a G0/8-é pedig 100. Az inside interfészről az outside-ra menő forgalom engedélyezett, de fordítva a forgalom tiltott. Így két nevesített, kiterjesztett ACL lett alkalmazva a G0/8 interfész bemenő forgalmára. Az ACL a SERVERv4 és SERVERv6 nevet kapta, amik a KKK-LINUX-t engedik elérni a 22-es, a 80-as és a 443-as porton. Emellett az összes Linux szervert SSH-val, az összes Windows szervert pedig RDP-vel lehet elérni. Ezeket csak az IT szakember IP címeiről, vagyis a 10.99.99.30 és a 2001:db8:c1c1:9999::30 címről lehet elérni. Az összes kliens gépet el lehet érni RDP-vel, arra az esetre, ha valaki távolról szeretne dolgozni. Ezenfelül bármilyen más forgalom az IPv4-es és az IPv6-os címtartományunk felé tiltott és az internet elérés miatt minden más engedélyezett. Az ASA alhálózatából a forgalom nincs korlátozva. Az ACL-ek a *22. képen* látható.

A képen szöveg, képernyőkép, szám, Párhuzamos látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

22. kép – ASA: ACL-ek

Az ASA Deep Packet Inspection (DPI) funkcióját kihasználva az ACL által az OSI-modell 2., illetve 3. szintjén ellenőrözött és engedélyezett forgalmat magasabb szinteken is ellenőrzi. A forgalmat az alapján nézi, hogy megfelel az adott RFC szabványnak, illetve letiltja a nem várt válasz üzeneteket. Ezek csak a nem biztonságos protokollokat figyelik. A biztonságos protokollok figyelése a SSL/TLS által adott védelem gyengüléséhez, illetve az ASA nagy mértékű leterheléséhez vezetne. A HTTP esetében a túlméretezett csomagok kiszűrésével a puffer elleni támadásokkal szemben védekezik. Emellett káros JavaScript, SQL, HTTP parancsokat URL-eket és férgeket, vírusokat is keres. A DNS és az ICMP protokollok figyelésével a tunneling, a DNS cache mérgezés, a DOS támadások, illeve a nem várt ICMP válasz üzenetek ellen védekezünk. A *23. képen* láthatóak a protokollok, amikkel az alapbeállítást kiegészítettük.

A képen szöveg, Betűtípus, képernyőkép, tipográfia látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

23. kép – A megfigyelt protokollok

A csomagok külön-külön vizsgálata mellett a threat-detection funkcióval a forgalom is vizsgálva lett. A hálózat, a port szkennelés, a DoS, és a brute force támadások ellen nyújt védelmet. Mivel alapértelemezetten csak eldobja a gyanús csomagokat és logot készít róla, ezért a beállítás a threat-detection scanning-threat shun, illetve a threat-detection scanning-threat shun duration 28800 paranccsal lett kiegészítve. Így a szokatlan forgalom növekedést észlelve blokkolja 8 órára a forrás IP címet.

2.1.12 WIFI

Mindegyik telephelyen ki lett alakítva egy ”Guest\_KKK” WiFi a vendégeknek, illetve egy ”KKK” WiFi az ott dolgozóknak. A vendégeké a 2,4 GHz-es, a dolgozóké a 5 GHz WiFi. A központban az első emeleten lévő Wifi Router AP módban van. A kliensek a DHCP szervertől kapnak címet. 2,4 GHz-en lentről felfele az AP-k az 1, a 6, és a 11 -es csatornát használják. A fiókirodákban az 1-es csatorna van használva. Mivel 5 GHz-n a csatornák nem fedik egymást, ezért azok kiosztása automatikus az AP-kon. A Guest\_KKK WiFi a WPA2-Personal biztonsági módszer alkalmazza előre megosztott kulccsal, és AES titkosítással. A megosztott kulcs: **49UzbeBQlW5a**. A KKK WiFi WPA2-Enterprise biztonsági módszer alkalmazza, RADIUS szerveres hitelesítéssel és AES titkosítással. A RADIUS szerver jelszava: **yl34bOIbX0S9**. A 2,4 GHZ-en a 802.11 szabvány b, g, és n szabványait használja, 5 GHZ-n pedig az a, n, és ac szabványokat. Azért támogatják a régebbi szabványokat, hogy egy régebbi eszköz is tudjon csatlakozni a vezetéknélküli hálózathoz. A *24. képen* látható a 2,4 GHz WiFi, a *25. képen* pedig a 5 GHz Wifi beállításai. Az *25. kép* a szegedi telephely AP-ján készült.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

24. kép – 2,4 GHz Wifi beállításai

A képen szöveg, képernyőkép, Betűtípus, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

25. kép – 5 GHz Wifi beállításai

2.1.13 Bannerek

Mindegyik kapcsolón és forgalomirányítón egy banner üzenet található, ami figyelmezteti a jogosulatlanul hozzáfért felhasználókat. A banner az *26. képen* láható.

A képen szöveg, képernyőkép, Betűtípus, sor látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

26. kép – Banner üzenet

2.2 WAN

A hálózat WAN része a szolgáltatóval egyeztetve lett kialakítva.

2.2.1 Internetszolgáltató

Az internetszolgáltató választásánál cél volt, hogy ismert és a telephelyek távolsága miatt az egész országban elérhető legyen. Három szolgáltató felelt meg az elvárásoknak a Magyar Telekom Nyrt, a Yettel Magyarország Zrt, és a One Magyarország (Digi). Miután mind a háromtól árajánlat lett kérve a Telekomra esett a választás, mivel ez a cég adta legjobb ár-értékű ajánlatot. Figyelembe lett véve a maximális sávszélesség, amit a szolgáltató képes nyújtani, illetve az optikai kábel bekötésének, az internetnek, illetve a publikus IPv4 cím bérlésének a költsége. A Telekomtól kapott árajánlat a *27. képen* látható. Ez a szolgáltató egyik értékesítési konzultánsának, Dániel Józsefnek az egyik válasz üzenete. Ez havi, nettó költséget jelent. Az optikai kábel bekötése ingyenes.

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

27. kép – Telekom ajánlata

A szolgáltatótól viszont három feladatot kaptunk: hirdessük a kapott publikus IPv4 címeinket BGP-vel, állítsuk be a BPG hitelesítést, illetve konfiguráljuk PPPoE kliensként a forgalomirányítót, ami az ISP-hez csatlakozik.

2.2.2 BGP, statikus forgalomirányítás

A szolgáltatótól kapott /27-es méretű címtartomány el lett osztva a telephelyek között. Ezek a Border Gateway nevű protokollal vannak hirdetve a szolgáltató felé. Privát AS számok lettek használva, mivel a szolgáltatótól vannak az IPv4-es címek bérelve és a hálózat mérete se követeli meg, hogy az európai RIR-nél (RIPE NCC) egy publikus AS szám legyen kérve. A használt privát ASN-ek a szolgáltatóval egyeztetve lettek. Az ISP forgalomirányítói a Telekom publikus AS számával lettek megadva a telephelyek routerein. Mivel ezzel csak a címtartományunk van hirdetve és a telephelyek routerei nem kapnak hirdetéseket, ezért egy statikus alapértelemezett útvonal lett megadva az ISP felé a PPPoE konfigurálása során. Az ASN-nek a *28. táblázatban* láthatóak. Példaként a *28. képen* látható a DEB-KKK-IR router BPG konfigurációja.

A szolgáltatótól kapott IPv6-os címek is ugyanúgy vannak hirdetve, mint az IPv4-esek. IPv6 esetében azonban létre kellett hozni egy peer-group-ot és az IPv6-os szomszédot bele kellett rakni. A peer-group név a szegedi forgalomirányítón PEERSZ, a debrecenin PEERD, és a központin PEERK. Mindkét address-family-ben az IPv4-es és IPv6-os cím is aktiválva lett, így, ha az ISP routere egyik címen nem érhető el, akkor is lehet hirdetni mindkét típusú IP címtartományt. Mivel IPv6 esetében nem lehet automatikusan létrehozni egy statikus alapértelemezett útvonalat, ezért mindegyik telephely esetében manuálisan lettek megadva.

|  |  |
| --- | --- |
| **AS számok** | |
| **Telephely** | **ASN** |
| ISP | 5483 |
| Budapest | 65001 |
| Szeged | 65003 |
| Debrecen | 65002 |

28. táblázat – Az ISP és a telephelyek AS számai

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

28. kép – DEB-KKK-IR BGP konfigurációja

2.2.3 PPPoE

A Telekom kérésére a routerihez csatlakozó forgalomirányítóinkat PPPoE kliensként kellett konfigurálni, hogy az ISP PPPoE szervere hitelesíteni tudja a routereinket. A konfiguráció során mind a három routeren létre lett hozva egy Dialer interfész 1-es számmal, ahol az MTU a maximális, 1492-es értékre lett állítva, illetve meg lett adva a szolgáltatótól kapott IP cím és a jelszó. Emellett hozzá lett rendelve az 1-es pool és a ppp ipcp route default paranccsal biztosítva lett, hogy a PPPoE kapcsolat létrejötte és a hitelesítés után a forgalomirányító táblába bekerüljön egy IPv4-es alapértelmezett útvonal az ISP felé. Az autentikáció során a szolgáltató kérésére nem a sima, szöveges jelszót használó PAP, hanem a biztonságosabb, titkosított jelszavú CHAP lett alkalmazva. A PPP beágyazás is konfigurálva lett a Dialer 1-es interfészen. A fizikai interfészek nem kaptak IP címet, hanem PPPoE kliensként lettek konfigurálva és az egyes pool lett hozzárendelve. Mind a három routeren azonos interfész, vagyis a GigabitEthernet 0/1/0 lett konfigurálva, és az előbb leírt beállítások is megegyeznek a forgalomirányítókon. Csak a jelszavak, illetve az IP címek térnek el. Az ISP a routereinket a hosztnevük alapján vette fel felhasználóként a PPPoE szerverén. A *29. táblázatban* látható a kiosztott jelszavak és IP címek. A *29. képen* példaként a KKK-IR PPPoE kliensként való konfigurációja látható.

|  |  |  |
| --- | --- | --- |
| PPPoE paraméterei | | |
| **Eszköz** | **IP cím** | **Jelszó** |
| KKK-IR | 203.5.100.34 | DXl2hucBm2 |
| DEB-KKK-IR | 203.5.100.38 | A3SnLFYINU |
| SZE-KKK-IR | 203.5.100.42 | RJmP8JciAb |

29. táblázat – ISP-től kapott PPPoE paraméterek

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

29. kép – KKK-IR PPPoE konfigurációja

2.2.4 PAT, SNAT

Mind a három telephelynél NAT lett alkalmazva. A szolgáltatótól kapott /27-es címtartomány lett szétosztva úgy, hogy a központi (KKK-IR) egy /28-as, a szegedi (SZE-KKK-IR) és a debreceni (DEB-KKK-IR) telephely pedig egy /29 címtartományt kapott. A NAT egyik típusa, a PAT lett konfigurálva pool-okkal. Mindegyik említett forgalomirányítón a NAT külső interfésze a Dialer1. Hogy a VPN szervert kívülről is el lehessen érni, ezért SNAT lett beállítva a KKK-IR routeren. A belső, privát címek engedélyezésre, meghatározásara normál, nevesített ACL lett létre hozva mindegyik forgalomirányítón PAT névvel, illetve a pool-ok is PAT névvel lettek létrehozva. A *30. táblázatban* láthatóak a konfigurált címek a NAT beállítása során.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NAT címek** | | | | |
| **Eszköz** | **Pool** | **Maszk** | **SNAT** | **Belső hálózat** |
| KKK-IR | 203.55.100.2 - 203.55.100.14 | /28 | 203.55.100.1  (10.11.199.2) | 10.11.0.0 /16 |
| DEB-KKK-IR | 203.55.100.17 - 203.55.100.22 | /29 | - | 10.37.0.0 /16 |
| SZE-KKK-IR | 203.55.100.25 - 203.55.100.30 | 10.23.0.0 /16 |

30. táblázat – NAT címek

2.2.5 DMVPN

A biztonságos kapcsolat kialakítás érdekében VPN-t használunk a telephelyek közötti kapcsolathoz. A Cisco saját protokollját a DMVPN-t használjuk a könnyű kapcsolatkialakítás érdekében. Phase 3-as kapcsolatot konfiguráltunk, ami azt jelenti, hogy magát a full-mesh topológiát kiépítik az eszközök egymás közt, így csak egy központi konfigurációt kell alkalmazni, és nem kell minden egyes eszközt (esetleges későbbi telephelyek kialakítása esetén sem), minden egyes régi és új routeren felvenni a kapcsolatot az új eszközökkel. A VPN kapcsolatot IKEv2-s előre megosztott kulcsos megoldással van védve. Sha512-vel és aes256-tal van biztosítva az integritás és a titkosítás. A site-ok közötti, tehát a VPN alagúton belüli IP forgalomirányítást EIGRP-vel végezzük, amin keresztül átadjuk a site-ok /16-os tartományait.

2.2.6 EIGRP

Az IR routerek között DMVPN alagútban EIGRP lett alkalmazva a dinamikus forgalomirányítás érdekében. Az EIGRP a 100-as AS számmal lett konfigurálva és az OSPF bele van forgatva. Ezzel lett biztosítva, hogy az összes alhálózatot ismerje mindegyik forgalomirányító mind a három telephelyen.

1. Szerverek

A hálózatban Linux és Windows szerverek nyújtanak különböző szolgáltatásokat, illetve egy kezdetleges felhő infrastruktúrát is létrehoztunk a cég későbbi fejlődéséhez.

3.1 Windows szerverek

Ez a rész a Windows szervereken lévő AD DS, DNS szerver, WDS, nyomtató szerver, RADIUS szerver szolgáltatásokat mutatja be. A DHCP, az ADDFS és az FSRM szolgáltatást az Automatizáció (”4.1.3”, ”4.1.4”, ”4.1.5”) fejezet mutatja be.

3.1.1 ADDS, RODC, subdomain

Az AD DS erdő root domain-je a kkk.com, amit a központ kapott meg. A deb.kkk.com a debreceni, az sze.kkk.com pedig a szegedi telep tartománya. Mindkettő a kkk.com gyermeke. Tehát ez a három egy fában és egy erdőben van. A root domain és a két altartomány között szülő-gyermek megbízhatósági viszony van, illetve az utóbbi kettő a kkk.com-on keresztül tranzitívan megbízik egymásban. A kkk.com tartományvezérlője a KKK-ADDS, a deb.kkk.com-é a DEB-KKK-WIN, a sze.kkk.com-é pedig a SZE-KKK-WIN szerver. A terhelés elosztás, illetve redundancia miatt egy RODC is található a központi telepen, ami a KKK-RODC nevű szerver. Mindegyik szerver egy globális katalógus, hogy felgyorsítsuk az erdő más domain-jeibe tartozó objektumok hitelesítését és a köztük való keresést. És így felgyorsul például az erdő más tartományába való bejelentkezés. Ezt azért is tehettük meg, mert az AD környezet nem túl nagy, ezáltal nem okoz nagy terhelést a hálózatnak a globális katalógusok replikálása változás esetén. Mindegyik DC-n egy OU lett létrehozva ”Felhasznalok” néven és benne egy ”Csoportok” OU, valamint minden munkakörnek saját, a munkakör nevével megegyező OU található. A felhasználók SamAccountName nevei a nevük kisbetűkkel ponttal elválasztva, vagyis <vezetéknév>.<keresztnév>, illetve a userPrincipalName nevük, pedig a SamAccountName és után @ és az adott tartománynév. Mindenkinek a pozíciója is fel van tüntetve. Mindegyik felhasználó jelszava **Password123**, ezt az első bejelentkezéskor meg kell változtatniuk. A foglalkozási köröknek megfelelő csoportok lettek létrehozva a ”Csoportok” OU-ban, és minden felhasználó a saját munkakörének megfelelő csoport tagja lett. Példaként a *30. képen* a KKK-ADDS szerveren létrehozott OU-k láthatóak.

A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

30. kép – KKK-ADDS: OU-k

3.1.2 Group Policy

A céges irányelvek biztosítása, illetve a dolgozók munkájának segítése érdekében group policy-kat használtunk mind a három DC szerveren. Mivel ezek mindegyik dolgozóra egyaránt vonatkoznak, ezért a Default Domain Policy lett szerkesztve. A GPO mindenkinek felcsatolja a [\\kkk.com\files](file:///\\kkk.com\files) elérést M betűjelű meghajtóként. Mindenkinek telepíti a ThunderBird szoftvert levelezéshez, illetve a Firefox-ot böngészéshez. Valamint mindenkinek hozzáadja az adott szervert nyomtató szerverként. A ThunderBird alternatívaként szolgál a levelezéshez, mivel van webmail a KKK-LIN szerveren. Emiatt a GPO létrehoz mindenkinek az asztalára egy parancsikont, ami a ”https://mail.kkk.com” URL-re mutat. Valamint a Firefox hivatalos GitHub oldaláról le lettek töltve az ADMX sablonok, és mindegyik DC szerveren be lettek másolva a *C:\Windows\PolicyDefinitions* mappába. Ezután a GPO-ban be lett állítva, hogy a kezdőlap a ”https://mail.kkk.com” legyen. Emellett más egyéb beállítások is vannak, amik a *31-32. képeken* láthatóak.

A képen szöveg, elektronika, képernyőkép, képernyő látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

31. kép – GPO: vezérlőpult, asztal

A képen szöveg, elektronika, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

32. kép – GPO: start menü, tálca, feladatkezelő

Az IT csoportra nem vonatkozik ez a GPO. Ennek beállítása a *33. képen* látható. Az IT OU kapott egy IT nevű GPO-t, ami csak a meghajtót csatolja fel, illetve telepíti a Thunderbird-öt, valamint a Firefoxot és beállítja a nyomtató szervert. Emellett az IT szakember is megkapja a parancsikont.

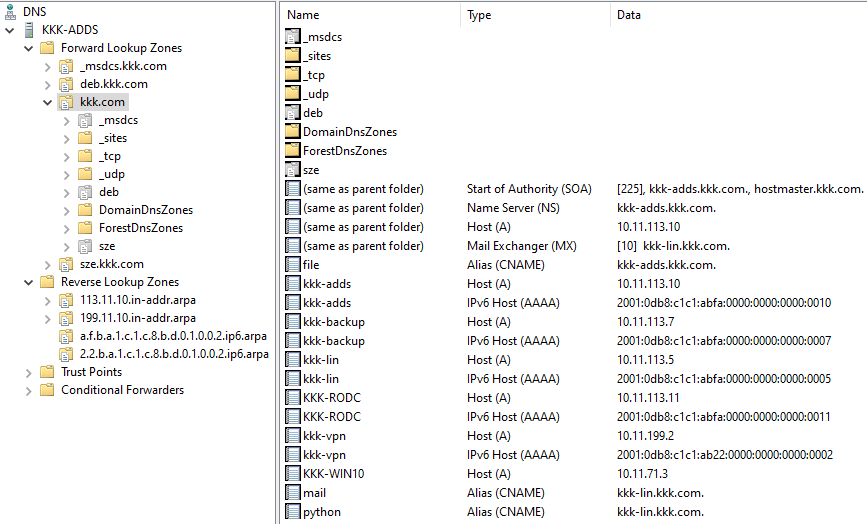
A képen szöveg, képernyőkép, képernyő, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

33. kép – GPO: IT csoport kizárása

3.1.3 DNS

A szerver szolgáltatások gyors és egyszerű eléréséhez, illetve az AD DS működéséhez DNS szolgáltatás lett telepítve a Windows szervereken. A KKK-ADDS szerveren a Linux és a Windows szervereknek fel lett véve egy-egy A és AAAA rekord. Emellett CNAME rekordok lettek felvéve azoknak a szolgáltatásoknak a neveivel, amik a KKK-LIN szerveren futnak. Tehát a CNAME rekordok alias-ok a kkk-lin-re. Egy MX rekord is létre lett hozva a kkk-lin szerverre mutatva az email szolgáltatás megfelelő működéséhez. A másik két DC szerveren pedig csak a DHCP szerver által bejegyzett rekordok találhatóak. Illetve mindegyik szerveren a másik két szerver elsődleges zónájához másodlagos zóna lett létrehozva. Erre a terhelés elosztás miatt volt szükség, illetve ezáltal biztosítva lett a DNS szolgáltatás megfelelő működése valamelyik fiókiroda szerverének rövid idejű kiesése esetén. Valamint a DNS lekérdezéseket is felgyorsítja. A zóna transzfer csak a másik kettő Windows szerver felé engedélyezett. Mindegyik telephelyen reverse zónák is létre lettek hozva az ott lévő szervert tartalmazó alhálózatokkal. A *34. képen* a KKK-ADDS szerver zónái és rekordjai láthatóak.



34. kép – KKK-ADDS: DNS

3.1.4 WDS

A kliensek gyors telepítése érdekében a KKK-ADDS, a DEB-KKK-WIN és a SZE-KKK-WIN szerveren WDS szolgáltatást telepítettük. Konfigurálása során engedélyeztük a megfelelő PXE opciók beállítását az összes DHCP scope-ban annak érdekében, hogy a kliensek megtalálják a WDS szolgáltatást futtató szervert. Minden ismert vagy ismeretlen hosztnak válaszol, ami megkönnyíti a beállítását. Mivel ez biztonsági okokból nem jó megoldás, ezért a kliensek telepítése után a szolgáltatás lekapcsoltuk. Az *35. képen* a boot és az install images láthatóak a KKK-ADDS szerveren.

A képen szöveg, képernyőkép, Betűtípus, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

35. kép – WDS: boot és install image

3.1.5 Nyomtató szerver

A nyomtatók központi kezelésére, a kliensekre való egyszerű telepítése érdekében a Print Management szolgáltatást telepítettük a KKK-ADDS, a SZE-KKK-WIN, és a DEB-KKK-WIN szerverekre. A HP OfficeJet Pro 8132e nyomtatókat használ cég. A fiókirodákban egyelőre egy-egy, míg a központban két darab van. A nyomtatók a központban a marketing, Szegeden az esemény, Debrecenben pedig a tehetség VLAN-ban vannak. A nyomtatóhoz szükséges driver fel lett véve. A nyomtatók IP címei és maguk a nyomtatók fel lettek véve. A nyomtatók a klienseken GPO-val vannak beállítva. Példaként a *36. képen* láthatóak a nyomtató szerver beállításai a SZE-KKK-WIN szerveren.

A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

36. kép – SZE-KKK-WIN: nyomtató szerver

3.1.6 RADIUS

A vezetéknélküli hálózatra csatlakozó alkalmazottak hitelesítésére mind a három telephely DC szerverére telepítettük a Network Policy Server szolgáltatást. Az NPS-t regisztráltuk az AD-ba. Az AP-k IP címe fel lettek véve és nevük „WIFI Ap <szám>”, illetve a „Main WIFI Router”. A számuk a földszintről felfelé növekszik a központban. A két fiókirodában pedig csak egy-egy van, így a nevük „WIFI Ap 1”. Mindegyik csoport számára engedélyezett a hozzáférés. Ezzel biztosítottuk, hogy minden dolgozó a saját AD felhasználójával be tudjon jelentkezni. A Microsoft: Protected EAP (PEAP) EAP metódust használtuk. Ehhez mindenképpen szükség volt egy tanúsítványra, ezért egy-egy önaláírt tanúsítványt hoztunk létre a szervereken azok FQDN-jeivel. Ennek létrehozásához szükséges parancs a következő:

New-SelfSignedCertificate `

-Type SSLServerAuthentication `

-DnsName <FQDN> `

-CertStoreLocation "Cert:\LocalMachine\My" `

-KeySpec KeyExchange `

-KeyUsage DigitalSignature, KeyEncipherment `

-KeyAlgorithm RSA `

-Provider "Microsoft RSA SChannel Cryptographic Provider" `

-HashAlgorithm SHA256 `

-NotAfter (Get-Date).AddYears(5)

Mivel ezek nem elismert tanúsítványok, ezért a tanúsítványt egyszer el kell fogadniuk a klienseknek. A klienseket pedig a szerver az EAP-MSCHAPv2 metódussal azonosítja a PEAP által létrehozott TLS alagútban. A policy-k Secure Wireless Connections névvel lettek létrehozva A *37. képen* látható a felvett RADIUS kliensek a DEB-KKK-WIN szerveren. A *38. képen* pedig a Network Policy és a hitelesítési metódus látható.

A képen szöveg, képernyőkép, szoftver, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

37. kép – DEB-KKK-WIN: RADIUS kliensek

A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

38. kép – DEB-KKK-WIN: RADIUS beállítások

3.2 Linux szerverek

A KKK-LIN és a KKK-VPN szerverek a Linux szerverek.

3.2.1 KKK-LIN

A Linux szervereinken a Debian 12.7-es verzióját használjuk, és grafikus felület nélküli telepítést végzünk, mivel a konfigurációs feladatokhoz nincs szükség GUI-ra és így jobb lesz a szerver erőforráskihasználása. A telepítés során megadott értékeket a *39-40. képeken* láthatják.

|  |
| --- |
| *A képen szöveg, képernyőkép, szoftver, képernyő látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  39. kép – Alap csomagok |

A következő (*40. kép*) képen a szerver IP beállításai tekinthetők meg:

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, fekete látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*40. kép – IP beállítások |

A szerverre pár egyéb csomagot is telepítettünk, ami még a szerver alapvető kezeléséhez szükséges. Ezek a(z):

* mc (GUI-s fájlkezelő)
* dnsutils (DNS lekérdezésekhez szükséges parancsok)
* net-tools (IP cím, interfész konfigurációs beállítások)
* ldap-utils (LDAP lekérdezések a Windows szerverről)

3.2.1.1 Ansible

Mivel a fő Windows szervereink függősége a Ansible playbookok megléte és futtatása, ezért első sorban ezt a környezetet hoztuk létre. A következő csomagokat telepítettük ehhez:

* ansible
* ansible-core

A hálózati automatizáció részben tovább olvashatnak ezekről a megoldásokról, valamint a hivatkozásoknál megtekinthetik a megírt kódokat.

3.2.1.2 LibreNMS

A LibreNMS egy olyan szolgáltatás, ami egyszerre nyújt Syslog, SNMP és egyéb hálózat monitorozási lehetőségeket. A sokoldalúsága miatt döntöttünk a szolgáltatás mellett, amik segítenek a hálózat későbbi esetleges hibáinak, gyengülésének, és ezeknek a gócpontoknak a felderítésében. A szolgáltatás komplex konfigurációjátnak telepítését a GitHub repozitóriumában leírt dokumentációja alapján végeztük el.

A LibreNMS szolgáltatás futtatásra a Docker konténert használjuk a jobb erőforráskihasználás érdekében. Ennek a telepítését a Docker oldalán található dokumentáció alapján végeztük el (<https://docs.docker.com/engine/install/debian>).

Miután feltelepítettük a Docker-t a szervergépünkre, a DockerHub-ból átirányítva a GitHub oldalon lévő dokumentáció alapján telepítettük a szolgáltatást. (<https://github.com/librenms/docker>). A telepítés során a következő lépéseken keresztül haladtunk végig:

* cp examples/compose /var/librenms/ (átmásoljuk a példa fájlokat)
* A saját változóinkra kicseréltük a \*.env -re végződő fájlok hiányos beállításait, amelyeket a következőkben láthatnak:
* docker compose up -d(futtatom a beállításaim alapján a LibreNMS szolgáltatást)

*.env*

TZ=Europe/Budapest

PUID=1000

PGID=1000

MYSQL\_DATABASE=librenms

MYSQL\_USER=librenms

MYSQL\_PASSWORD=VizsgaremekPass12345

*librenms.env*

MEMORY\_LIMIT=512M

MAX\_INPUT\_VARS=1000

UPLOAD\_MAX\_SIZE=16M

OPCACHE\_MEM\_SIZE=128

REAL\_IP\_FROM=10.0.0.0/8

REAL\_IP\_HEADER=X-Forwarded-For

LOG\_IP\_VAR=remote\_addr

CACHE\_DRIVER=redis

SESSION\_DRIVER=redis

REDIS\_HOST=redis

LIBRENMS\_SNMP\_COMMUNITY=KKK-SNMP

LIBRENMS\_WEATHERMAP=false

LIBRENMS\_WEATHERMAP\_SCHEDULE=\*/5 \* \* \* \*

*msmtpd.env*

SMTP\_HOST=mail.kkk.com

SMTP\_PORT=587

SMTP\_TLS=on

SMTP\_STARTTLS=on

SMTP\_TLS\_CHECKCERT=on

SMTP\_AUTH=on

SMTP\_USER=kovacs.gabor

SMTP\_PASSWORD=Password123

SMTP\_FROM=kovacs.gabor@kkk.com

A szervereket a SNMPv2c segítségével adtuk hozzá a felderítéshez, valamint beállítottuk, hogy a logjaikat küldjék tovább erre a szerverre. Az előre megadott „kulcs”, azaz a community name *KKK-SNMP*. A felvett szervereket a LibreNMS webfelületén a *41. képen* láthatják.

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*41. kép – Szerverek listája SNMP-ben |

A hálózati eszközök SNMPv3 segítségével, még biztonságosabban csatlakoznak SHA-512, és AES-256-C titkosítástok segítségével, valamint minden 6-os vagy annál súlyosabb szintű logot, elküldenek a szervernek. Az eszközöket külön szerveztük, minden eszközt a saját locationjába, így adott siteok, eszközeit is képesek vagyunk egy felület alól megnézni, hogy minden rendben van-e velük, és hogy éppen milyen teljesítménnyel működnek, ha szükségünk lenne erre, valamint különböző szolgáltatásokat is megtekinthetünk.

3.2.1.3 Webszerver

A hálózatunkban a webszerver szolgáltatást arra használjuk, hogy Python segítségével, konfigurációkat tudjunk küldeni SSH-n keresztül adott eszközöknek, egy konfiguráció összerakását segítő oldalon. A weboldal alapvetően kisebb kártyákból épül fel, ami mellett megtalálható az éppen beírt kód, a navbarban megadhatjuk, hogy milyen IPv4 címre, milyen felhasználóval és jelszóval küldjük az adatokat.

A weboldal egy Python script segítségével valósítja meg az SSH kapcsolatot, majd küldi el ezeket az adatokat a megadott SSH szervernek, amin végrehajtja, majd egy kimenetet ad a weboldal aljára, amin látható, hogy mi történt a parancssorban, a feladatok elvégzése közben, ha esetleg, valami elgépelés történt volna.

Az oldalt csak az IT csoport tagjai, használhatják, ezt egy LDAP-os bejelentkezéssel biztosítjuk. A webszerver kiszolgálja a klienseket, IPv4-gyel és IPv6-tal is.

A következő szolgáltatásokat telepítettük fel ennek a megoldásnak a részeként:

* python3-pip, python3-venv, python3-flask, python3-paramiko
* apache2, libapache2-mod-wsgi-py3, libnss-ldapd, libpam-ldapd

A python3-mal kezdődő csomagok, a Python script megfelelő futásáért, míg a többi csomag az apache2 webszerver és annak bővített moduljaiért felel, ezek segítenek a Python kód használatában, illetve az LDAP hitelesítésben.

A weboldal hosztolásához, először is megszüntettük az alapoldal elérhetőségét (a2dissite 000-default), ezen kívül létrehoztuk az */etc/apache2/sites-available/ssh\_website.conf* fájlt, aminek a következő a tartalma:

<VirtualHost \*:443>

    ServerName python.kkk.com

    WSGIDaemonProcess ssh\_website python-path=/var/www/ssh\_website:/var/www/ssh\_website/venv/lib/python3.9/site-packages

    WSGIProcessGroup ssh\_website

    WSGIScriptAlias / /var/www/ssh\_website/ssh\_website.wsgi

    SSLEngine on

    SSLCertificateFile /cert/mail.crt

    SSLCertificateKeyFile /cert/mail.key

    SSLCertificateChainFile /cert/ca.crt

    <Location />

        AuthType Basic

        AuthName "Login with just IT users!"

        AuthBasicProvider ldap

        AuthLDAPURL "ldap://kkk-adds.kkk.com/OU=IT,OU=Felhasznalok,DC=kkk,DC=com?sAMAccountName?sub?(objectClass=person)"

        AuthLDAPBindDN "CN=Administrator,CN=Users,DC=kkk,DC=com"

        AuthLDAPBindPassword "Password123"

Require all denied

        Require ldap-group CN=IT,OU=Csoportok,OU=Felhasznalok,DC=kkk,DC=com

        Require valid-user

    </Location>

    <FilesMatch "\.(?:cgi|shtml|phtml|php)$">

        SSLOptions +StdEnvVars

    </FilesMatch>

    <Directory /usr/lib/cgi-bin>

        SSLOptions +StdEnvVars

    </Directory>

    <Directory /var/www/ssh\_website>

        Require all granted

    </Directory>

    Alias /static /var/www/ssh\_website/static

    <Directory /var/www/ssh\_website/static>

        Require all granted

    </Directory>

    ErrorLog ${APACHE\_LOG\_DIR}/ssh\_website\_error.log

    CustomLog ${APACHE\_LOG\_DIR}/ssh\_website\_access.log combined

</VirtualHost>

A weboldalt SSL/TLS titkosítással védjük, hogy csak HTTPS-en keresztül lehessen elérni, az ehhez használt fájlokat, később az email szerver konfigurálásánál olvashatnak részletesebben, ezek közül a fő tanúsítvány az Active Directoryban, GPO segítségével meg lett osztva a domain tagjaival, a *mail.crt, mail.key* fájl pedig a titkosítás folyamatához szükséges tanúsítványok.

A <Location> tageken belül pedig az LDAP-on keresztüli hitelesítést állítottunk be úgy, hogy csak az IT csoport tagjai tudjanak bejelentkezni az oldalra, a felhasználónevük és jelszavuk segítségével, ezen kívül van még egy biztonsági szűrő, ami csak a cég privát IPv4-es és publikus IPv6-os tartományából engedi a belépést a felhasználóknak.

A következő lépésben engedélyeztük a weboldalt a a2ensite ssh\_websiteparanccsal, ami létrehozott egy szimbolikus linket a */etc/apache2/sites-available/ssh\_website.conf* fájlról a */etc/apache2/sites-enabled/* mappába, ezután pedig engedélyeztük a következő modokat, hogy működjenek a konfigurációs fájlban tárolt beállítások. A modok engedélyezésére használt parancs: a2enmod ssl ldap authnz\_ldap auth\_basic. Ezek után a */var/www/ssh\_website/* elérési útvonal alá odamásoltuk a weboldalt, és a mappán a következő biztonsági beállításokat végeztük el: chown -R www-data:www-data /var/www/ssh\_website/valamint chmod -R 750 /var/www/ssh\_website/ amik segítségével az apache2 felhasználója minden fájlt rekurzívan el tud érni a megadott mappában, a csoportja pedig olvasási és áthaladási jogot kap minden objektumra.

Ezt követően újraindítottuk a szolgáltatást és ez a parancs hiba nélkül lefutott, a weboldal, a webszerver elérhetővé vált.

3.2.1.4 Email

A Kihagyhatatlan Kincsek Kiadó hálózatában helyet foglal egy email szerver, ami a belső emailezésre szolgál, hogy ne egy külső szolgáltatón keresztül, hanem belső hálózatból, AD-s bejelentkezéssel lehessen végezni, így az egyedi email szolgáltatás árát megspórolva, egy teljesen jól működő rendszer készült el.

Technikai oldalról megközelítve, a KKK-LIN szerveren van egy dovecot, és postfix szolgáltatás, ami a KKK-ADDS-ről, a Domain Controller szerverünkről végzi az LDAP lekéréseket.

Az email szolgáltatás úgy, mint ahogyan az egész hálózat Dual Stack azaz IPv4, és IPv6-os megoldást is alkalmaz, így elősegítve a későbbi átállást az IPv6-ra.

A konfiguráció során több hibába is belefutottunk, az Active Directoryba való LDAP-os bejelentkezés megvalósítása során, de ezeket mind megoldottuk.

Az email szolgáltatáshoz a Postfix/Dovecot párost használjuk. A Postfix felel az üzenetek elküldéséért, ez az SMTP szerverünk. A Dovecot felel az üzenetek fogadásáért, és emailek lekérdezéséért. A következő csomagokat telepítettük a szolgáltatásokhoz:

* dovecot-core dovecot-imapd dovecot-ldap
* postfix postfix-ldap

Mivel itt is törekedtünk a biztonságos beállításokra ezért, először létrehoztuk a hitelesítő fájlokat. Készítettünk egy CA-t, ehhez egy privát kulcsot, ami a CA privát kulcsa. Ezek segítségével kiállítottunk egy Certificatet a KKK-LIN.kkk.com (mail.kkk.com, python.kkk.com) számára egy tanúsítványt (publikus certificatet, és privát kulcsot). A CA-t beimportáltam a Default Domain Policy-be így a domainen belüli számítógépek számára, ez egy megbízható szerver lesz. Ezeket a kulcsokat, mind a Postfix mind a Dovecot használja (valamint, mint a korábbiakban említettük az Apache2 webszerver is).

A Postfix konfigurációjához, a fő fájlt *(/etc/postfix/main.cf)* kellett szerkeszteni, amit a következőképpen végeztünk el:

smtpd\_banner = $myhostname ESMTP $mail\_name (Debian/GNU)

biff = no

# appending .domain is the MUA's job.

append\_dot\_mydomain = no

readme\_directory = no

# See http://www.postfix.org/COMPATIBILITY\_README.html -- default to 3.6 on

# fresh installs.

compatibility\_level = 3.6

# TLS parameters

smtpd\_tls\_cert\_file=/cert/mail.crt

smtpd\_tls\_key\_file=/cert/mail.key

smtpd\_tls\_CAfile=/cert/ca.crt

smtpd\_tls\_security\_level=encrypt

smtp\_tls\_CApath=/cert

smtp\_tls\_security\_level=may

smtp\_tls\_session\_cache\_database = btree:${data\_directory}/smtp\_scache

smtpd\_relay\_restrictions = permit\_mynetworks permit\_sasl\_authenticated defer\_unauth\_destination

myhostname = mail.kkk.com

alias\_maps = hash:/etc/aliases

alias\_database = hash:/etc/aliases

myorigin = /etc/mailname

mydestination = $myhostname, KKK-LIN.kkk.com, localhost.kkk.com, localhost

relayhost =

mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128 [2001:db8:c1c1:abfa::]/64 10.0.0.0/8

mailbox\_size\_limit = 0

recipient\_delimiter = +

inet\_interfaces = all

inet\_protocols = all

mydomain = kkk.com

virtual\_mailbox\_domains = $mydomain

virtual\_alias\_maps = ldap:/etc/postfix/ldap/virtual\_alias\_maps

virtual\_mailbox\_base = /

virtual\_mailbox\_maps = ldap:/etc/postfix/ldap/virtual\_mailbox\_maps

virtual\_uid\_maps = ldap:/etc/postfix/ldap/virtual\_uid\_maps

virtual\_gid\_maps = ldap:/etc/postfix/ldap/virtual\_gid\_maps

smtpd\_sender\_login\_maps = ldap:/etc/postfix/ldap/smtpd\_sender\_login\_maps

smtpd\_sasl\_type = dovecot

smtpd\_sasl\_path = private/auth

smtpd\_sasl\_auth\_enable = yes

broken\_sasl\_auth\_clients = yes

# Logging is optional but can help A LOT with debugging

maillog\_file = /var/log/postfix.log

A következő beállításokat tárolja ez a fájl:

* TLS beállítások (használja az általunk előállított tanúsítványokat és kulcsokat)
* A szerver nevét, elérhetőségét (IP kommunikáció alapján és név alapján)
* Virtuális mapolásokat (későbbiekben bemutatott fájlok)
* Dovecot használata
* Log fájlok helye

A *main.cf* fájlban található virtuális mapolások, az LDAP lekéréseket szolgálják, a következőkben a 4 fájl-t tekinthetik meg. Mindegyik fájl eleje megegyezik, ezek azt állítják, be, hogy honnan, és milyen felhasználó segítségével kérjék le a beállításokat:

server\_host = kkk-adds.kkk.com:389

version = 3

bind = yes

bind\_dn = cn=Administrator,cn=Users,dc=kkk,dc=com

bind\_pw = Password123

search\_base = ou=Felhasznalok,dc=kkk,dc=com

scope = sub

*virtual\_alias\_maps:*

…

query\_filter = (|(sAMAccountName=%s)(sAMAccountName=%s))

result\_attribute = sAMAccountName

A keresés a sAMAccountName mező alapján keres és azt is adja vissza (felhasználónév).

*virtual\_mailbox\_maps:*

…

query\_filter = (|(sAMAccountName=%s)(userPrincipalName=%s))

result\_attribute = sAMAccountName

result\_format = /mailboxes/%s/

Az előbbiekben leírt keresés a megadott mezők alapján keres, a felhasználónév, illetve a teljes név alapján (felhasználónév@domain), és az emailek lerakási helyét adja vissza *(/mailboxes/felhasználónév)*.

*virtual\_uid\_maps:*

…

query\_filter = (|(sAMAccountName=%s)(userPrincipalName=%s))

result\_attribute = uid

Ez a keresés ugyanúgy, mint az előző kérés a felhasználónév és az email cím alapján szűr. Viszont az uid mezőt adja vissza.

*virtual\_gid\_maps:*

…

query\_filter = (|(sAMAccountName=%s)(userPrincipalName=%s))

result\_attribute = gidNumber

Ez a keresés ugyanúgy, mint az előző kérés a felhasználónév és az email cím alapján szűr. Viszont az gidNumber mezőt adja vissza, tehát a csoportot. Ez a csoport egy univerzális csoport, aminek a felhasználók email könyvtárára, van 7 joga, ezzel létre tudja hozni az emaileket a megadott helyre.

*smtpd\_sender\_login\_maps:*

…

query\_filter = (|(userPrincipalName=%s)(sAMAccountName=%s))

result\_attribute = sAMAccountName

Fordítva intézi a keresést, és a felhasználónevet adja vissza.

A konfiguráció során, ezeknek a mapoknak a beállítása okozta a legnagyobb problémát, mivel az Active Directoryban nem tároltunk le alapvetően homeDirectory, uid is gidNumber mező értéket, amiből az uid-re a Postfixnek, a maradékra, és az uidra ugyanúgy a Dovecotnak van szüksége. Ezek a mezők, viszont nem voltak helyettesíthetők unique nevekkel, mint például a sAMAccountName, ezért a létrehozás során egy növekvő számértéket, adtunk minden felhasználónak, amivel így tudja azonosítani a Dovecot is.

A Postfixnél még két beállítást végeztem el a */etc/postfix/master.cf* fájlban:

# Choose one: enable submission for loopback clients only, or for any client.

#127.0.0.1:submission inet n - y - - smtpd

submission inet n - y - - smtpd

# -o syslog\_name=postfix/submission

-o smtpd\_tls\_security\_level=encrypt

# -o smtpd\_sasl\_auth\_enable=yes

-o smtpd\_tls\_auth\_only=yes

...

# Choose one: enable submissions for loopback clients only, or for any client.

#127.0.0.1:submissions inet n - y - - smtpd

submissions inet n - y - - smtpd

# -o syslog\_name=postfix/submissions

-o smtpd\_tls\_wrappermode=yes

# -o smtpd\_sasl\_auth\_enable=yes

A Dovecot konfigurációja sokkal több fájlból áll össze, ezért csak rövid kódrészleteket mutatunk, amit megváltoztattunk az alapbeállításokhoz képest.

*/etc/dovecot/conf.d/10-mail.conf:*

mail\_location = maildir:/mailboxes/%u

namespace inbox {

prefix =

location = maildir:/mailboxes/%u

inbox = yes

separator = /

}

namespace {

prefix = Sent/

separator = /

list = yes

subscriptions = yes

location = maildir:/mailboxes/%u/Sent

}

namespace {

location = maildir:/mailboxes/%u/Trash

separator = /

prefix = Trash/

list = yes

subscriptions = yes

}

namespace {

location = maildir:/mailboxes/%u/Drafts

separator = /

prefix = Drafts/

list = yes

subscriptions = yes

}

Ezek a beállítások az emailek elhelyezési és kiolvasási helyét határozzák meg. Meg van adva, hogy az emailek a */mailboxes/felhasználónév/* elérési útvonal alatt található és ezalatt van bejövő-, elküldött-, kuka- és piszkozat emailek.

*/etc/dovecot/conf.d/10-ssl.conf:*

##

## SSL settings

##

# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>

ssl = yes

# PEM encoded X.509 SSL/TLS certificate and private key. They're opened before

# dropping root privileges, so keep the key file unreadable by anyone but

# root. Included doc/mkcert.sh can be used to easily generate self-signed

# certificate, just make sure to update the domains in dovecot-openssl.cnf

ssl\_cert = </cert/mail.crt

ssl\_key = </cert/mail.key

Megadjuk a publikus kulcsot és certificatet, valamint megadjuk, hogy bekapcsolt a SSL/TLS titkosítás.

*/etc/dovecot/conf.d/10-auth.conf:*

disable\_plaintext\_auth = no

mail\_privileged\_group = dovecot

auth\_mechanisms = plain login

#!include auth-system.conf.ext

!include auth-ldap.conf.ext

A következő sorokkal a bejelentkezés beállításait változtattuk meg. Kikapcsoltuk a system alapú bejelentkezést, és engedélyeztük az LDAP konfigurációs fájlt.

*/etc/dovecot/dovecot-ldap.conf.ext:*

# LDAP URIs to use. You can use this instead of hosts list. Note that this

# setting isn't supported by all LDAP libraries.

uris = ldap://kkk-adds.kkk.com

dn = CN=Administrator,CN=Users,DC=kkk,DC=com

dnpass = Password123

auth\_bind = yes

base = OU=Felhasznalok,DC=kkk,DC=com

scope = subtree

user\_filter = (&(objectClass=person)(sAMAccountName=%u))

user\_attrs = uid=uid,gidNumber=gid,homeDirectory=home

pass\_filter = (&(objectClass=person)(sAMAccountName=%u))

pass\_attrs = uid=uid,userPassword=password

Ez a fájl tartalmazza, az

* LDAP szervert
* Összekötő felhasználót
* Az alapvető keresési tartományt
* A felhasználó szűrést és attribútumokat
* A jelszó szűrést, és attribútumokat.

A filter és az attribútumok meghatározása az Active Directory miatt nehézkes volt, viszont sikeresen megoldottuk.

*/etc/dovecot/conf.d/10-master.conf:*

service auth {

...

# Postfix smtp-auth

unix\_listener /var/spool/postfix/private/auth {

mode = 0666

user = postfix

group = postfix

}

}

A következő beállítás meghatározza, hogy a Postfix az email küldő szolgáltatásunk.

*/etc/dovecot/conf.d/10-logging.conf:*

log\_path = /var/log/dovecot.log

Ezzel a sorral pedig meghatározzuk, hogy egy külön fájlba gyűjtse a log bejegyzéseket a szolgáltatás.

3.2.1.5 Rsyslog szerver

A logok összegyűjtésére a hálózatban, amit nem tud befogadni a LibreNMS készítettünk egy Rsyslog szervert, ami összegyűjti SSL/TLS hitelesítés segítségével a logokat a hálózatból, és továbbítja a LibreNMS-nek, viszont ez sajnos nem teszi bele a saját logjai közé, így külön fileokban tárolódnak a KKK-LIN szerveren.

A syslog szolgáltatáshoz, pontosabban az Rsyslog-hoz a következő csomagokra van szükség:

* rsyslog
* rsyslog-gnutls
* gnutls-bin

A szolgáltatáshoz azért van szükség erre a három csomagra, hogy az SSL/TLS hitelesítéshez szükséges modulokat is telepítse.

A Rsyslognak alapvetően van egy fő konfigurációs fájlja a */etc/rsyslog.conf*, ami behúzza az összes *.conf*-ra végződő fájlt a */etc/rsyslog.d/* útvonal alól. Emiatt és a könnyebb átláthatóság miatt ebbe a mappába készítettük el a konfigurációs fájlokat. Két fájlt készítettünk, az első a *server.conf* nevű fájl:

# Listen on TCP 6514 with TLS

global(

DefaultNetstreamDriver="gtls"

DefaultNetstreamDriverCAFile="/cert/ca.crt"

DefaultNetstreamDriverCertFile="/cert/mail.crt"

DefaultNetstreamDriverKeyFile="/cert/mail.key"

)

module(

load="imtcp"

StreamDriver.Name="gtls"

StreamDriver.Mode="1"

StreamDriver.AuthMode="anon"

)

input(

type="imtcp"

port="6514"

address="0.0.0.0"

)

input(

type="imtcp"

port="6514"

address="::"

)

$template remote-incoming-logs, "/var/log/remote/%HOSTNAME%.log"

\*.\* ?remote-incoming-logs

A konfigurációs fájl több részből tevődik össze, amit a *31. táblázatban* olvashatnak.

|  |  |
| --- | --- |
| global direktíva | Megadjuk a titoksítás értékeit, illetve a tanúsítványokat |
| modul direktíva | Betölti az *imtcp* modult, ami a logok fogadásáért van, ezen kívül beállítja a gnutls segítségével az SSL/TLS fogadás módját |
| input direktíva (IPv4) | Az összes IPv4-es interfacen a TCP 6514-es porton hallgat |
| input direktíva (IPv6) | Az összes IPv6-os interfészen a TCP 6514-es porton hallgat |
| template | Megadja, hogy ennek minden beérkező logot, a kliens hosztnevének megfelelő fájlba helyezzen, így szeparálva a jobb átláthatóság érdekében |

31. táblázat - /etc/rsyslog.d/server.conf leírása

Ezen kívül van még egy fájl, ami arra szolgálna, hogy a LibreNMS-nek az 514-es UDP porton keresztül továbbítsa a logokat fájlok alapján. Ez a fájl a *send.conf*:

module(load="imfile")

# KKK-LIN

input(

type="imfile"

File="/var/log/remote/KKK-LIN.log"

Tag="KKK-LIN"

Severity="info"

Facility="local6"

)

# KKK-VPN

input(

type="imfile"

File="/var/log/remote/KKK-VPN.log"

Tag="KKK-VPN"

Severity="info"

Facility="local6"

)

\*.\* @127.0.0.1:514

A részeinek leírását a *32. táblázatban* láthatják.

|  |  |
| --- | --- |
| modul direktíva | Betöltjük az imfile modult (egy fájl továbbítása) |
| input direktívák | A fájl változást néző direktívák külön logfájlokra |
| log továbbítás | Az input direktívában meghatározott log fájlok továbbítása a szerver 514-es UDP portjára |

32. táblázat - /etc/rsyslog.d/server.conf leírása

3.2.1.6 Tűzfal

Tűzfalként egy egyszerű szoftvert az UFW-t telepítettük a Linuxos szervereinkre, mivel egyszerű a kezelése. Maga az egész Iptables-t használ, de saját parancsokkal kezelhető.

A következő parancsokkal konfiguráltuk fel a tűzfalat:

ufw add \*port\_szám\*/\*protokoll\*

Ez a parancs hozzáad egy új IPv4-es és IPv6-os szabályt, ami adott porton és protokollon engedélyezi a bejövő forgalmat.

Az ‘ufw status’ paranccsal le lehet kérni az éppen érvényben lévő szabályokat, ezt a *42.képen* láthatják.

|  |
| --- |
| A képen szöveg, képernyőkép, fekete látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.42. kép – KKK-LIN tűzfal szabályok |

3.2.1.7 Webmail

Webmail szolgáltatásként Roundcube Webmail-t konfiguráltunk fel. A beépített csomagtelepítő segítségével a Debian repositroyból telepítettük a *roundcube* csomagot, ami telepítette magával a MariaDB-t is, ami a webmailhez szükséges adatbázist biztosítja.

A szokásos adminisztrátori jelszót alkalmaztuk az adatbázishoz. A roundcube beállításaiban nem kellett sokat állítanunk, mivel a telepítés alatt minden opciót megfelelő értékekkel állított be, csak az Apache2 konfigurációnkat kellett kibővíteni. A szokásos tanúsítványt használtuk ebben az esetben is. A Roundcube az IMAP és SMTP szolgáltatásainkat használja.

Az apache2 Virtual Host beállításai:

<VirtualHost \*:443>

ServerAdmin kovacs.gabor@kkk.com

ServerName KKK-LIN.kkk.com

ServerAlias mail.kkk.com

DocumentRoot /var/lib/roundcube

DirectoryIndex index.php

# Those aliases do not work properly with several hosts on your apache server

# Uncomment them to use it or adapt them to your configuration

# Alias /roundcube /var/lib/roundcube/public\_html

<Directory /var/lib/roundcube/public\_html/>

Options +FollowSymLinks

# This is needed to parse /var/lib/roundcube/.htaccess. See its

# content before setting AllowOverride to None.

AllowOverride All

<IfVersion >= 2.3>

Require all granted

</IfVersion>

<IfVersion < 2.3>

Order allow,deny

Allow from all

</IfVersion>

</Directory>

# Protecting basic directories (not needed when the document root is

# /var/lib/roundcube/public\_html):

<Directory /var/lib/roundcube/config>

Options -FollowSymLinks

AllowOverride None

</Directory>

<Directory /var/lib/roundcube/temp>

Options -FollowSymLinks

AllowOverride None

<IfVersion >= 2.3>

Require all denied

</IfVersion>

<IfVersion < 2.3>

Order allow,deny

Deny from all

</IfVersion>

</Directory>

<Directory /var/lib/roundcube/logs>

Options -FollowSymLinks

AllowOverride None

<IfVersion >= 2.3>

Require all denied

</IfVersion>

<IfVersion < 2.3>

Order allow,deny

Deny from all

</IfVersion>

</Directory>

<Directory /var/lib/roundcube>

Options FollowSymLinks

AllowOverride All

Require all granted

</Directory>

ErrorLog /var/log/apache2/roundcube/error.log

CustomLog /var/log/apache2/roundcube/access.log combined

SSLEngine on

SSLCertificateFile /cert/mail.crt

SSLCertificateKeyFile /cert/mail.key

<FilesMatch /usr/lib/cgi-bin>

SSLOptions +StdEnvVars

</FilesMatch>

<Directory /usr/lib/cgi-bin>

SSLOptions +StdEnvVars

</Directory>

</VirtualHost>

Ezeknek a beállításoknak egy részét, előre letelepített fájlok között lehetett megtalálni, a többi beállítást, pedig mi adtuk hozzá. Ha a szervert mail.kkk.com szerver néven kérik le a böngészőben akkor visszaadja ezt az oldalt. A *43-44. képeken* a bejelentkező és a használati felületet láthatják.

|  |
| --- |
| A képen képernyőkép, szoftver, szöveg, Számítógépes ikon látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.43. kép –Roundcube bejelentkező felület |
| *A képen szöveg, szoftver, Számítógépes ikon, Operációs rendszer látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  44. kép –Roundcube felhasználói felület |

Mint látható a szolgáltatás betölti az emaileket.

3.2.2 KKK-VPN

A Linux szervereinken a Debian 12.7-es verzióját használjuk, és grafikus felület nélküli telepítést végzünk, mivel a konfigurációs feladatokhoz nincs szükség GUI-ra és így jobb lesz a szerver erőforráskihasználása. A telepítés során megadott értékeket a *45-46. képeken* láthatják.

|  |
| --- |
| A képen szöveg, képernyőkép, szoftver, képernyő látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.45. kép – A szervere telepítése alatt feltelepített csomagok |

A következő (*46. kép*) képen a szerver IP beállításai tekinthetők meg:

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, fekete látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.* 46. kép – IP beállítások |

A szerverre pár egyéb csomagot is telepítettünk, ami még a szerver alapvető kezeléséhez szükséges. Ezek a(z):

* mc (GUI-s fájlkezelő)
* dnsutils (DNS lekérdezésekhez szükséges parancsok)
* net-tools (IP cím, interfész konfigurációs beállítások)

3.2.2.1 Wireguard

A Wireguard szolgáltatásra azért volt szükség, hogy távolról is elérhető legyen az email szolgáltatás és a Pythonos weboldal. A Wireguard egy olyan VPN kliens, ami biztonságos, és megfelelő titkosítást nyújt az adatok védelme érdekében. A KKK-VPN szerveren egy ASA mögött található azért, hogy minél jobban meg legyen szűrve az a forgalom, ami nem engedélyezett.

A VPN szolgáltatás telepítéséhez, a *wireguard* csomagot telepítettük le. Ez minden parancsot elérhetővé tesz nekünk, ami a konfigurációhoz szükség van. Ezen kívül az *ldap-utils* szolgáltatást a később olvasható scripthez.

Ahhoz, hogy a becsatlakozó kliensek elérjék a szolgáltatásokat a hálózatból, az *frr* nevű csomagot is telepítettük, ami OSPF forgalomirányító protokollt segítségével beroutolja a hálózatba a forgalmat.

Mivel alapból nem érkezik konfigurációs fájlal a wireguard, ezért sajátot kell létrehozni. Viszont mivel a létrehozás során minden felhasználónak külön kulcsot hozunk létre, ezért egy shell scripetet írtunk a konfigurációs fileok létrehozására, amit a következőkben láthatnak:

#!/bin/bash

# LDAP server details

LDAP\_SERVER="ldap://kkk-adds.kkk.com"

BASE\_DN="ou=Felhasznalok,dc=kkk,dc=com"

BIND\_DN="cn=Administrator,cn=Users,dc=kkk,dc=com"

BIND\_PASSWORD="Password123"

i=3

echo -e "[Interface]\nPrivateKey = uGA+0Q9KV8cTEm+4eew0seQ8F8y0PNdcvAC8X4tpPXc=\nAddress = 10.99.99.2/24, 2001:db8:c1c1:9999::2/64\nListenPort = 51820\n" > /etc/wireguard/wg0.conf

# Perform LDAP search and extract sAMAccountName

sAMAccountNames=$(ldapsearch -x -H "$LDAP\_SERVER" -b "$BASE\_DN" -D "$BIND\_DN" -w "$BIND\_PASSWORD" "objectClass=person" sAMAccountName | grep "^sAMAccountName: " | sed 's/^sAMAccountName: //')

sr\_publickey=$(cat /etc/wireguard/public.key)

# Loop through each sAMAccountName

for accountName in $sAMAccountNames; do

echo "Processing user: $accountName"

mkdir -p "/etc/wireguard/cert/$accountName"

cd "/etc/wireguard/cert/$accountName"

wg genkey | tee "./private-$accountName.key"

chmod go= "./private-$accountName.key"

cat "./private-$accountName.key" | wg pubkey | tee "./public-$accountName.key"

privatekey=$(cat "./private-$accountName.key")

cl\_publickey=$(cat "./public-$accountName.key")

echo -e "[Interface]\nPrivateKey = $privatekey\nAddress = 10.99.99.$i/24, 2001:db8:c1c1:9999::$i/64\nDNS = 10.11.113.10, 2001:db8:c1c1:abfa::10/64\n\n[Peer]\nPublicKey = $sr\_publickey\nEndpoint = 10.11.113.2:51820\nAllowedIPs = 0.0.0.0/0, ::/0" > "./wg-$accountName.conf"

echo -e "\n# $accountName\n[Peer]\nPublicKey = $cl\_publickey\nAllowedIPs = 10.99.99.$i/32, 2001:db8:c1c1:9999::$i/128" >> /etc/wireguard/wg0.conf

i=$((i + 1))

done

Ez a fájl elkészíti a szerver konfigurációs fájlját pont mint az össze felhasználónak az összes felhasználónak, akik az Active Directory Domainen belül vannak. Ez a script csak a fő épületben lévő felhasználóknak készíti el ezeket, van 2 további shell scriptünk, ami a másik két sitenak ugyanezt megcsinálja csak a keresési zóna az adott fő OU.

Ezzel elkészül a konfiguráció, a fájlokat minden felhasználónak a saját meghajtójára másoljuk, valamint emailen keresztül kiküldjük nekik, amitől kezdve ők kezelhetik a saját fájljukat, ha elveszik újra oda lehet adni nekik, illetve lehet újat is generálni, viszont, ekkor minden felhasználónak újra ki kell küldeni az új konfigurációs fájlokat.

Az *frr* konfigurálásához a */etc/frr/daemons* fájlban a következő változtatásokat végeztük el:

…

ospfd=yes

ospf6d=yes

…

Ezzel engedélyeztük a szolgáltatást. A konfiguráció további lépései úgy működtek, mint egy cisco eszközön. A vtysh parancs kiadása után (bash-ben), egy új shell nyílt meg ami, mint említettük, egy Cisco eszköz konfigurációját szimulálja. A konfiguráció mentésével az /*etc/frr/frr.conf* fájl írja a szolgáltatás majd egy újraindítást követően a szerver már működőképes.

3.2.2.2 Rsyslog kliens

A logokat a KKK-LIN szerveren gyűjtjük, ezért rsyslog segítségével, SSL/TLS titkosítással továbbítjuk a szerver összes logját a log szerverünk felé.

A megfelelő működéshez három csomagra telepítésére volt szükségünk, ezek a(z):

* rsyslog
* rsyslog-gnutls
* gnutls-bin

Ezek a csomagok biztosítják magának a syslog meglétét, illetve azt, hogy a szerver biztonságosan továbbíthassa a logokat.

A szolgáltatás kliensoldali konfigurációja során, átmásoltam a szükséges tanúsítványokat a szerverről, amik segítségével titkosítja a szerver a logokat a küldés során. Ezt követően a */etc/rsyslog.d/* mappába létrehoztam egy *send.conf* nevű fájlt, mivel az */etc/rsyslog.conf* fájl behúzza az összes fájlt abból a mappából ami .conf-ra végződik, ezért a konfigurációhoz inkább készítettünk egy külön fájl, aminek segítségével jobban átlátható a konfiguráció.

A következő kódrészlet szolgálja az összes log továbbítását a KKK-LIN szerverre:

$DefaultNetstreamDriverCAFile /cert/syslog/ca.crt

$DefaultNetstreamDriverCertFile /cert/syslog/client.crt

$DefaultNetstreamDriverKeyFile /cert/syslog/client.key

$DefaultNetstreamDriver gtls

$ActionSendStreamDriverMode 1

$ActionSendStreamDriverAuthMode anon

\*.\* @@kkk-lin.kkk.com:6514

3.3 Felhő

A felhőben csak egy kezdetleges infrastruktúra lett létrehozva.

3.3.1 Amazon Web Services

A kettő vezető felhőszolgáltató (Amazon Web Services, Microsoft Azure) közül az AWS lett választva. Ennek okai a következők:

* Az AWS a legtöbb szolgáltatással rendelkezik, több mint 200 különböző szolgáltatással. Ennek köszönhetően, ha később igény merülne fel egy újabb szolgáltatásra, akkor kevesebb eséllyel kell új szolgáltatót használni, és ezáltal könnyebb lesz a felhőszolgáltatások kezelése.
* Az AWS gyakran vezető szerepet játszik az új technológiák és szolgáltatások bevezetésében. Így a legtöbb esetben tudjuk alkalmazni a legújabb lehetőségeket.
* Az AWS volt az első nagyobb felhőszolgáltatás, amely a piacon megjelent, és jelentős szerepet játszott a felhőalapú számítástechnika népszerűsítésében. Több nagy vállalat használja világszerte, mint például a Netflix vagy a Facebook. Bár a Microsoft Azure-t is sok nagy cég használja (pl. Volkswagen), de az AWS-t olyan a technológia fejlődésben élen lévő vállatok is, mint például a NASA vagy a Forma-1.

3.3.2 A felhő

A webkiszolgáló felhőbe telepítésére azért volt szükség, hogy egyszerűen skálázható, megbízható és könnyen karbantartható legyen. A skálázhatóság azért fontos, mert később a cég növekedése esetén többen fogják felkeresni a weboldalt akár a világ minden részéről. A megbízhatóság és a könnyű karbantartás pedig azért szükséges, hogy a webkiszolgáló folyamatosan elérhető legyen. A fájlkiszolgáló szolgáltatás a weboldalon megjelenő képek tárolására kellett. Az adatbázis szolgáltatás a termékek adatainak tárolása miatt kellett. Ezek is a felhőben vannak, hogy a webkiszolgáló gyorsan és mindig elérje őket. Ezek a megoldások együttesen biztosítják a webszerver teljesítményét és rugalmasságát, optimális felhasználói élményt nyújtva.

A szolgáltatások a svéd régióban (Stockholm, eu-north-1) lettek létre hozva. Bár a frankfurti közelebb van, a cég arra számít, hogy az észak-európai országokban nagyobb lesz a kereslet a magyar zenék iránt, mint Európa többi részén.

3.3.3 Webkiszolgáló

Az AWS egy Iaas típusú szolgáltatása, az EC2 lett alkalmazva a webkiszolgáló létrehozásánál. Mivel nem rég jött létre a cég, még nem számítanak sok vásárlóra. Csak egy t3.micro példány lett létrehozva, amely 2 vCPU-val és 1 GB memóriával rendelkezik. Amiatt, hogy a cég a kezdeti időszakban még kevés vásárlóra számít és a költségek csökkentése érdekében ez a példány megfelelőnek lett találva. A t3.micro havonta 750 órát futhat ingyenes 12 hónapig, Az alacsony erőforrás, illetve a költségek csökkentése miatt Windows helyett Debian 12 operációs rendszer van a virtuális gépen. Mivel webkiszolgálóként van bekonfigurálva, ezáltal a virtuális gép egy olyan biztonsági csoportot kapott, ami engedélyezi az SSH mellett a HTTP és HTTPS forgalmat az internet felől. A beállításoknál a központ publikus címtartománya lett megadva, így az SSH forgalom csak a központ felől engedélyezett. Az a virtuális gép SSH-n keresztüli eléréséhez szükséges RSA privát kulcs az IT szakember számítógépén a *C:\felhő* mappában Webszerver-kulcs.pem névű fájlban van eltárolva. A kulcson csak az IT szakember felhasználójának csak olvasás, valamint olvasás és végrehajtás joga van. Ez kötelező beállítás volt. A mappában megnyitott parancssorban a *47. képen* látható parancsot kell kiadni a webkiszolgáló eléréséhez.

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

47. kép - Webkiszolgáló elérése

A virtuális gépen Apache2 webszerver fut és az index.php, valamint a styles.css fájlok a */var/www/html* mappában vannak. A weboldal adatbázishoz való hozzáféréséhez telepítve lett a php, a libapache2-mod-php és a php-mysql csomag. A Certbot is telepítve és használva lett, hogy egyszerűen lehessen szerezni SSL/TLS tanúsítványt, hogy biztosítva legyen a HTTPS kapcsolat a webszerverhez. A Certbot az automatikus tartomány ellenőrzés után a tanúsítványt a Let’s Encrypt-től szerezte. A példány a webszerver nevet kapta. A példány hálózati paraméterei a *48. képen* láthatóak.

A képen szöveg, képernyőkép, Betűtípus, szoftver látható

Automatikusan generált leírás

48. kép - Az EC2 hálózati paraméterei

3.3.4 Fájlkiszolgáló szolgáltatás

A fájlkiszolgáló létrehozásához az AWS egy Paas típusú szolgáltatása, az S3 lett alkalmazva. Létre lett hozva egy webszerverkepek bucket, amiben a weboldalon megjelenő képek vannak tárolva. A létrehozott bucket és néhány eltárolt kép a *49. képen* látható.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Automatikusan generált leírás

49. kép - Webszerverkepek nevű bucket

3.3.5 Adatbázis szolgáltatás

Az adatbázis kiszolgáló létrehozásánál az AWS egyik Paas típusú szolgáltatása, az RDS lett alkalmazva. Egy db.t3.micro példány lett létrehozva, amely 1 vCPU-val és 1 GB memóriával rendelkezik és MySQL motort használ. Ez is, mint a t3.micro megfelelőnek lett találva a kezdeti időszakra. Ugyanúgy 12 hónapon keresztül 750 órát futhat ingyenesen. A példány a ”webszervertermekek” nevet kapta. A db.t3.micronak csak privát IP címe van, így csak az AWS-en létrehozott szolgáltatások érhetik el. A biztonság növelése érdekében jelszó lett alkalmazva. Az adatbázis szolgáltatás eléréséhez szükséges paraméterek a *33. táblázatban* láthatóak. A hálózati beállításai az *50. képen* láthatóak.

|  |  |
| --- | --- |
| **Adatbázis kiszolgáló elérése** | |
| **Paraméter neve** | **Paraméter értéke** |
| Hosztnév | webszervertermekek.c3k6ggiwmhhx.eu-north-1.rds.amazonaws.com |
| Portszám | 3306 |
| Felhasználónév | admin |
| Jelszó | KOlaleha879 |

33. táblázat - Az adatbázis kiszolgáló eléréshez szükséges paraméterek

A képen szöveg, képernyőkép, Betűtípus, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

50. kép - Az RDS hálózati beállításai

A kiszolgálón egy ”webserver” nevű adatbázis és benne egy ”products” nevű tábla lett létrehozva. A táblában ”title”, ”description”, ”price” és ”image\_url” mezők vannak. Több tábla nem lett létrehozva, mivel a weboldal további fejlesztése a cégben dolgozó webfejlesztő feladata. Ezért a weboldalon egyelőre a Kosárba gombra kattintva egy üzenet ugrik fel, ami tájékoztatja a felhasználót arról, hogy csak telefonon keresztül lehet rendelni, valamint a telefonszám is megjelenik.

3.4 Automatizált mentés

A szerverek tárolt adatok és konfigurációk védelme érdekében automatizáltan végzünk biztonsági mentéseket róluk.

3.4.1 KKK-BACKUP

A Linux szervereinken a Debian 12.7-es verzióját használjuk, és grafikus felület nélküli telepítést végzünk, mivel a konfigurációs feladatokhoz nincs szükség GUI-ra és így jobb lesz a szerver erőforráskihasználása. A telepítés során megadott értékeket az *51-53. képeken* láthatják.

|  |
| --- |
| *A képen szöveg, képernyőkép, képernyő, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*51. kép - Hosztnév |
| *A képen szöveg, képernyőkép, képernyő, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*52. kép – Domain név |
| *A képen szöveg, elektronika, képernyőkép, képernyő látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*53. kép – A szervere telepítése alatt feltelepített csomagok |

A következő (*54. kép*) képen a szerver IP beállításai tekinthetők meg:

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, fekete látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*54. kép – IP beállítások |

A szerverre pár egyéb csomagot is telepítettünk, ami még a szerver alapvető kezeléséhez szükséges. Ezek a(z):

* mc (GUI-s fájlkezelő)
* net-tools (IP cím, interfész konfigurációs beállítások)

3.4.1.1 RAID 5

A biztonsági mentések redundáns adattárolása miatt a RAID 5. szintjét alkalmaztuk. Ezzel biztosítjuk, hogy ha 1 merevlemez meghibásodik a háromból, akkor is megmaradjanak a szerverek biztonsági mentései. A következő csomagot telepítettük ehhez:

* mdadm

A következő parancs lefuttatásával lett létrehozva a RAID 5 tömböt:

mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/sdb /dev/sdc /dev/sdd

Ezután ext4 fájlrendszerrel lett formázva a tömb és fel lett csatolva a /mnt/backup könyvtárba. A következő parancsokkal lett elmentve a RAID konfigurációja, illetve biztosítva lett a tömb automatikus felcsatolása bootolás során.

mdadm --detail --scan >> /etc/mdadm/mdadm.conf

update-initramfs -u

echo "UUID=661bbff4-42af-4c01-b6d4-f0b24fb60254 /mnt/backup ext4 defaults 0 0" >> /etc/fstab

A /mnt/backup mappában egy linux, illetve egy windows mappa is létre lett hozva. A Windows szerverek neveivel mappák lettek létrehozva a windows mappában.

3.4.1.2 Samba, Rsync

A backup szerver megfelelő működéséhez a következő csomagok lettek telepítve:

* samba
* rsync

A samba csomag feltelepítésére azért volt szükség, hogy a Windows szerverek elérjék a /mnt/backup/windows/<hosztnév> mappát. A Samba-val a /mnt/backup/windows mappa lett megosztva, amit csak a root felhasználó érhet el. A megosztás neve backup. A megosztás konfigurációja a következő:

[backup]

path = /mnt/backup/windows

browseable = no

guest ok = no

writeable = yes

read only = no

valid users = root

Az rsync csomag telepítésére pedig a Linux szerverek biztonsági mentése miatt volt szükség.

3.4.2 Automatizált mentés: Linux szerverek

A Linux szervereken a következő csomagok lettek telepítve:

* rsync
* sshpass
* cron

A /usr/local/bin mappában található a backup.sh script. A script a /var/log/backup.log fájlba írja a futása során keletkezett logokat. Létrehozza a KKK-BACKUP szerveren a /mnt/backup/linux/<hosztnév> mappát. Ezután a jogok, a szimbolikus linkek, az időbélyegek, illetve a kiterjesztett attribútumok megtartásával egy mentést készít a root fájlrendszerről, kivéve az átmeneti, virtuális fájlokat és felcsatolt fájlrendszereket tartalmazó mappákról. Az előző napi mentést referenciának használja, így csak a megváltozott fájlok lesznek átmásolva, a többi csak hard link lesz. A parancs hiba kimenetét a log fájlba irányítja. Ezután az előző mentéshez tartozó, latest nevű szimbolikus linket kitörli és egy újat hoz létre. Végül pedig a 30 napnál régebbi mentéseket kitörli. A script a következő:

#!/bin/bash

BACKUP\_DIR="/mnt/backup/linux"

REMOTE\_SERVER="root@KKK-BACKUP"

REMOTE\_BACKUP\_DIR="/mnt/backup/linux/$(hostname)"

DATE=$(date +%F)

LATEST="$REMOTE\_BACKUP\_DIR/latest"

LOG\_FILE="/var/log/backup.log"

PASSWORD="root"

EXCLUDES="--exclude=/dev/\* --exclude=/proc/\* --exclude=/sys/\* --exclude=/tmp/\* --exclude=/run/\* --exclude=/mnt/\* --exclude=/media/\* --exclude=/lost+found"

echo "Backup started at $(date)" >> $LOG\_FILE

sshpass -p "$PASSWORD" ssh -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null $REMOTE\_SERVER "mkdir -p $REMOTE\_BACKUP\_DIR"

sshpass -p "$PASSWORD" rsync -aAXv -e "ssh -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null" --link-dest=$LATEST $EXCLUDES / $REMOTE\_SERVER:$REMOTE\_BACKUP\_DIR/$DATE >> $LOG\_FILE 2>&1

sshpass -p "$PASSWORD" ssh $REMOTE\_SERVER "rm -rf $LATEST && ln -s $REMOTE\_BACKUP\_DIR/$DATE $LATEST"

sshpass -p "$PASSWORD" ssh $REMOTE\_SERVER "find $REMOTE\_BACKUP\_DIR -maxdepth 1 -type d -mtime +30 -exec rm -rf {} \;"

echo "Backup completed at $(date)" >> $LOG\_FILE

Mindegyik Linux szerveren egy próba SSH kapcsolat is el lett végezve, hogy teszteljük az SSH kapcsolatot, illetve, hogy a KKK-BACKUP szerver tanúsítványának publikus kulcsát elmentsük. A script a cronnal lett ütemezve. A root crontab-ja lett a következőképpen beállítva:

0 3 \* \* \* /usr/local/bin/backup.sh

Mindegyik Linux szerveren így lett beállítva az automatizált mentés. Hajnali háromtól 10 perces eltolódással végzik el a Linux szerverek a biztonsági mentést. A mentés tehát mindennap egy olyan inkrementális mentést végez, ahol mindegyik napi mentés egy teljes mentéssel egyenértékű.

3.4.3 Automatizált mentés: Windows szerverek

A Windows szervereknél a Windows Server Backup funkció lett felhasználva. Teljes biztonsági mentést végeznek a [\\KKK-BACKUP\backup](file:///\\KKK-BACKUP\backup) Samba megosztáson belül a saját mappájukba. A Windows szerverek éjféltől 20 perces eltolódással végzik a mentéseket. Az *55. képen* a KKK-ADDS szerveren a biztonsági mentés beállításai láthatóak. Az *56. képen* pedig a Samba megosztás elérése látható a KKK-ADDS szerverről.

A képen szöveg, képernyőkép, szoftver, képernyő látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

55. kép – KKK-ADDS: backup

A képen szöveg, képernyőkép, Betűtípus, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

56. kép – KKK-ADDS: Samba megosztás elérése

4.Automatizáció

A hálózatban több automatizációs folyamat található meg. Mivel a Windows szerverek alapvetően elég ingatagok, ezért ezek alapvető konfigurációját Ansible-el valósítjuk meg, amihez a szerver a KKK-LIN eszközöné fut. Ansible segítségével a következő szolgáltatásokat telepítjük és konfiguráljuk a Windows szervereinken (*34. táblázat*). Ezen kívül a Active Directory környezet felépítésére és a felhasználók létrehozására, PowerShell scripteket írtunk, amiket egy batch file (kötegfájl) lefuttatásával lehet elindítani. A felhasználók egy *.csv* file-ban lesznek megtalálhatók, és így ez a folyamat is automatizáltan fog működni.

|  |  |
| --- | --- |
| **Windows szerver Ansible** | |
| **Eszközök** | **Szolgáltatások** |
| KKK-ADDS  KKK-RODC | ADDS |
| DNS |
| AD DFS |
| FSRM |
| KKK-ADDS | DHCP |

34. táblázat – Ansible-el telepített szolgáltatások

Továbbá a KKK-LIN szerveren egy webszerver fut, ami egy távoli konfigurációs felület. Az Active Directory-s bejelentkezés segítségével lehet használni. A rendszergazda megadhatja ezen a felületen, hogy mit szeretne konfigurálni, és az oldal segítségével összekattinthatja a konfigurációt, majd IP cím, SSH felhasználónév, SSH jelszó, és enable jelszó megadását követően rá tudja küldeni SSH segítségével az összerakott kódot az adott eszközre.

Az Ansible kódok megírásánál arra törekedtünk, hogy tökéletesen elkészítse az adott konfigurációt az eszközhöz, és ha másodszorra is lefuttatjuk, akkor ne legyen már változás a feladatoknál. Az Ansible kódok, és a PowerShell scriptek a dokumentáció végén a hivatkozások részben tekinthetők meg.

4.1 Ansible kódok

4.1.1 ADDS

Az Active Directory telepítését és felkonfigurálását végzi el a playbook. A playbookban alapvető Windows *(új néven: microsoft.ad)* modul segítségével, létrehozza a *„kkk.com”* tartományt a KKK-ADDS szerveren, majd, ha szükséges akkor újraindítja, és ezt követően domain kontrolleré promótálódik a szerver. A következő play a DNS kliens beállítása a KKK-RODC szerveren, ami ahhoz szükséges, hogy csatlakoztathassuk a kkk.com tartományba, ezért beálltja a KKK-ADDS IPv4 és IPv6 címét DNS kiszolgálónak. Ezelőtt van egy tesztelés, ami arra szolgál, hogy ha be van már állítva ez a két cím az eszközön akkor ugorja át az előbb leírt lépést. Ezután már csak két play van a playbookban, az első becsatlakoztatja a domainbe a KKK-RODC eszközt, majd, ha szükséges újraindítja, a második feltelepíti az Active Directory Domain Services és DNS szolgáltatást, majd ezután Read-Only Domain Controlleré lépteti elő, és ha szükséges újraindítja. Az *57-59.* képen a sikeres első és a *changed* állapot nélküli második futtatás, valamint az Active Directory Users and Computers-ből kivágott képernyőkép arról, hogy tényleg működött a kód látható a képeken.

|  |
| --- |
| A képen képernyőkép, szöveg, Színesség látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.57. kép – adds.yaml első futtatás |

|  |
| --- |
| A képen képernyőkép, szöveg látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.58. kép – adds.yaml második futtatás |

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, szám látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*59. kép – Domain controllerek |

4.1.2 DNS

A DNS (*Domain Name Services*) szolgáltatást a *dns.yaml* playbookban konfiguráljuk. Első körben a **KKK-ADDS** szerveren beállítja DNS szervernek önmagát és a **KKK-RODC** szervert (IPv4 és IPv6 címeket egyaránt) ezzel megteremtve azt, hogy rekordokat replikálja az írásvédett tartományvezérlőre. Az egész playbook a **KKK-ADDS** szervert konfigurálja. Mivel ehhez nincs alapvető modul, csak community, PowerShell parancsok segítségével kell megoldani a konfigurációt és a tesztelést ahhoz, hogy a második futtatásnál ne legyen *changed* állapotú task. Ennek következtében, az első playben a szerver beállítása előtt még leteszteljük, hogy már be van-e állítva neki a 4 IP cím. A következő play először végig nézi, hogy létezik-e:

* a 113.11.10.in-addr.arpa fordított dns zóna
* a 10.11.113.5-s rekord (A rekord tesztelés)
* a 2001:db8:c1c1:abfa::5 rekord (AAAA rekord tesztelés)
* a mail.kkk.com rekord (CNAME rekord tesztelés)
* a DNS továbbító beállítása

Ezután az előbbi taskok kimenetét használva, ha a megadott szövegrészletek nincsenek bent a kimenetben, akkor lefuttatja a taskokat, amik az inventory fájlban leírt részletek segítségével végig loopol az ott leírtakon. A taskok a következő beállításokat végzik.

* Tartománynévrendszer elsődleges zónáinak létrehozása
* A rekordok létrehozása
* AAAA rekordok létrehozása
* CNAME rekordok létrehozása
* DNS forwarder felvétele

A *60-61. képeken* az első futtatás és a *changed* állapot nélküli második futtatás látható. Ezután a *62-63. képeken* a DNS konzolban látható, hogy a rekordok és zónák tényleg létrejöttek. Ezek mind a következő oldalakon találhatóak

|  |
| --- |
| A képen szöveg, képernyőkép, Betűtípus, tervezés látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.60. kép – dns.yaml első futtatás |
| A képen szöveg, képernyőkép látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.61. kép – dns.yaml második futtatás |
| *A képen szöveg, képernyőkép, szám, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*62. kép – KKK-ADDS DNS konzol |

|  |
| --- |
| *A képen szöveg, képernyőkép, szám, menü látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*63. kép – KKK-RODC DNS konzol |

4.1.3 DHCP

A DHCP (Dynamic Host Configuration Protocol) szolgáltatást a ”dhcp.yaml” playbook konfigurálja. A folyamat átláthatósága és megkönnyítése érdekében az inventory.yaml-ben az egyes IPv4-es és IPv6-os scopeok beállításait felvettük, így loopok segítségével könnyebben és átláthatóbban létrehozhatóak az adott scope-ok. A script a következő taskokon, lépéseken keresztül konfigurálja fel a KKK-ADDS Windows szervert DHCP szerverré:

* Letelepíti a dinamikus állomáskonfiguráló protokoll szolgáltatást és a hozzátartozó vezérlési eszközöket
* Ha szükséges újraindítja az eszközt
* Megnézi, hogy a DHCP modul létezik-e
* A DHCP szervert engedélyei a tartományba
* A Server Managert értesíti arról, hogy sikeres volt a telepítés
* Beállítja, hogy regisztrálja a csatlakozó eszközöket DNS szerverbe (DDNS)
* Létrehozza, és konfigurálja a DHCPv4 scope-okat
* Létrehozza, és konfigurálja a DHCPv6 poolokat

Mivel a legtöbb konfiguráció (a beépített szolgáltatás telepítésen kívül) nem elérhető az alap ansible moduleban, ezért PowerShell scriptek segítségével lettek ezek megvalósítva, így a konfiguráció során, a *changed\_when: false* értéket használtuk fel sok helyen. Az ansible playbook lefutását a *64-65. képeken* láthatják, majd a *66-67. képeken*, pedig a szerveren a DHCP management console-ból, hogy valóban létrejöttek ezek a scope-ok a megadott opciókkal és kizárásokkal.

|  |
| --- |
| *A képen képernyőkép, szöveg, Multimédiás szoftver, Grafikai szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*64. kép – dhcp.yaml első futás |

|  |
| --- |
| *A képen képernyőkép, szöveg látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*65. kép – dhcp.yaml második futás |

|  |
| --- |
| A képen szöveg, képernyőkép, szoftver, szám látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  66. kép – KKK-ADDS IPv4 scope |

|  |
| --- |
| A képen szöveg, képernyőkép látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  67. kép – KKK-ADDS IPv6 scope |

4.1.4 ADDFS

Az Active Directory Distributed File System (*ADDFS*) szolgáltatás telepítését és konfigurálását az *addfs.yaml* playbook végzi. A szolgáltatás telepítését, mindkét szerveren elvégezzük, mivel ez a lényege, hogy több helyen is jelen legyen a file szerver. A playbook a következő playeken és taskokon keresztül végzi el a konfigurációt:

* Telepíti a szolgáltatást (és a replikációhoz szükséges szolgáltatást is) vezérlőeszközökkel együtt
  + Ha szükséges újraindítja az eszközöket
* Létrehozza a mappákat *„C:\”* elérési útvonal alá
* Megosztja ezeket a mappákat Samba segítségével
* Lecsekkolja, hogy létezik-e a DFS namespace gyökér, ha még nem akkor folytatja
* Elkészíti a namespace rootot
* Hozzáadja a namespace mappákat a namespace root-alá
* Másodlagos hozzáférést is készít hozzá
* Elkészíti a replikációs csoportot
* Hozzáadja a szervereket
* Beállítja a replikálandó mappákat
* Létrehozza a kapcsolatot a két eszköz között
* Frissíti a kapcsolat attribútumait a replikációhoz (loop segítségével)

A *68-70. képeken* az első, illetve második lefutást lehet megtekinteni, amiben van egy probléma, a második futásnál, a KKK-ADDS szerveren a Files nevű mappa megosztása, valamiért minden alkalommal (az első futtatást követően) *changed* állapotba kerül, és ezt nem is tudtuk kijavítani. A *71-73. képeken* a konzolból mutatjuk be, hogy sikeresen lefutott a playbook, és meg is csinálta.

|  |
| --- |
| A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.68. kép – addfs.yaml első futás (1. play) |

|  |
| --- |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  69. kép – addfs.yaml első futás (2. play) |

|  |
| --- |
| *A képen képernyőkép, szöveg látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*70. kép – addfs.yaml második futás |

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, sor látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.* 71. kép – Replikációs csatlakozások |

|  |
| --- |
| *A képen szöveg, képernyőkép, szám, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.* 72. kép – Replikációs tagság |

|  |
| --- |
| A képen szöveg, szám, Betűtípus, képernyőkép látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  73. kép – Replikációs mappák |

4.1.5 FSRM

Az File Server Resource Manager (*FSRM*) szolgáltatás telepítését az *fsrm.yaml* file végzi. Ezt is mindkét szerverre alkalmazzuk, mivel az összes Active Directory Distributed File System által kezelt és replikált mappának a beállítását és korlátozását ezzel végezzük el. Minden mappára az ahhoz tartozó csoportokra jellemző fájl kiterjesztéseket engedélyezzük és minden mást korlátozunk. Ezen felül, van egy *hard* korlát beállítva mindegyik csoportra, ami később növelhető a fileok mérete alapján. A következő lépéseken halad keresztül:

* Telepíti az FSRM szolgáltatást management tools-okkal együtt
* Létrehozza a kvótákat (maximum méret korlátozása)
* Létrehozza a fájl kiterjesztés szűrésére alkalmas csoportokat
* Alkalmazza a fájl kiterjesztés szűrését a mappákra

A *74-75. képeken* a playbook sikeres lefutását lehet végigtekinteni, valamint a *76-78. képeken* a létrejött beállításokat.

|  |
| --- |
| *A képen szöveg, képernyőkép, képernyő, szoftver látható  Automatikusan generált leírás*  74. kép – fsrm.yaml első futás |

|  |
| --- |
| *A képen szöveg, képernyőkép, képernyő, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.* 75. kép – fsrm.yaml második futás |

|  |
| --- |
| *A képen szöveg, képernyőkép, sor, szám látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*76. kép – File groupok |

|  |
| --- |
| *A képen szöveg, képernyőkép, szoftver, Weblap látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*77. kép – File screenek |

|  |
| --- |
| *A képen szöveg, szám, Betűtípus, sor látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*78. kép – File kvóták |

4.2 PowerShell

Az organizational unitek (*OU-k*), groupok (*csoportok*), és felhasználók létrehozására, és csoportba csatlakoztatására, *.ps1* scripteket írtunk, amik segítségével, csak fel kell másolni az adott eszközökre a két-két scriptet, a *.csv* file, ami tartalmazza a felhasználó adatokat, amik a létrehozásukhoz szükségesek, valamint egy *.bat* fájlt, aminek a lefuttatásával meggyorsítjuk a lefuttatást is.

5. Árkalkuláció

A hálózat kialakításához szükséges eszközök ára, illetve telepenkénti darabszáma a *35. táblázatban* látható. A táblázatban fel van tüntetve az eszközök darabára, összára, illetve a teljes összár. Az árak 2024.11.14.-i árat jelképezik, ekkor lett az árajánlat kiállítva. A legtöbb eszközből több is lett vásárolva a telepítés során vagy után keletkező hibák gyors javítására.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Árkalkuláció** | | | | | |
| **Eszközök** | **Központ (db)** | **Debrecen (db)** | **Szeged (db)** | **Darabár** | **Összár** |
| 24 portos MLS | 3 db | 0 db | 0 db | 505 882 Ft | 1 517 646 Ft |
| 48 portos L2 switch | 3 db | 2 db | 2 db | 388 263 Ft | 2 717 841 Ft |
| SFP modul (10Gbps) | 26 db | 18 db | 18 db | 34 625 Ft | 2 146 750 Ft |
| Router | 4 db | 4 db | 4 db | 112 500 Ft | 1 350 000 Ft |
| Wifi router | 1 db | 1 db | 1 db | 174 999 Ft | 524 997 Ft |
| Wifi AP | 3 db | 0 db | 0 db | 87 990 Ft | 263 970 Ft |
| ASA | 1 db | 0 db | 0 db | 342 000 Ft | 342 000 Ft |
| Windows szerver (Main ADDS) | 1 db | 0 db | 0 db | 1 401 328 Ft | 1 401 328 Ft |
| Windows szerver | 1 db | 1 db | 1 db | 1 114 344 Ft | 3 343 032 Ft |
| Linux szerver | 2 db | 0 db | 0 db | 1 373 979 Ft | 2 747 958 Ft |
| NAS | 1 db | 0 db | 0 db | 901 560 Ft | 901 560 Ft |
| Háttértár | 16 db | 1 db | 1 db | 43 170 Ft | 777 060 Ft |
| Nagy rack szekrény | 1 db | 0 db | 0 db | 324 700 Ft | 324 700 Ft |
| Kis rack szekrény | 0 db | 1 db | 1 db | 91 500 Ft | 183 000 Ft |
| UTP kábel(100m) | 26 db | 4 db | 5 db | 12 000 Ft | 420 000 Ft |
| RJ45-s csatlakozó (100db) | 5 db | 3 db | 3 db | 395 Ft | 4 345 Ft |
| **Eszközök** | **Központ (db)** | **Debrecen (db)** | **Szeged (db)** | **Darabár** | **Összár** |
| RJ45-aljzat | 80 db | 20 db | 26 db | 805 Ft | 101 430 Ft |
| Számítógépek | 36 db | 8 db | 12 db | 193 070 Ft | 10 811 920 Ft |
| Monitor | 36 db | 8 db | 12 db | 48 599 Ft | 2 721 544 Ft |
| Billentyűzet | 40 db | 12 db | 16 db | 3 399 Ft | 231 132 Ft |
| Egér | 40 db | 12 db | 16 db | 1 889 Ft | 128 452 Ft |
|  |  |  |  | **Összár:** | **32 960 665 Ft** |

35. táblázat - Árkalkuláció

A *36. táblázatban* látható az eszközök, és a hozzájuk tartozó oldalak URL-jei, amiken az eszközök megvásárolhatóak. Az oldalak néhány paramétert is tartalmaznak.

|  |  |
| --- | --- |
| **Eszközök - Oldalak** | |
| **Eszközök** | **Elérés** |
| 24 portos MLS | <https://www.senetic.hu/product/CBS250-24FP-4X-EU> |
| 48 portos L2 switch | <https://www.senetic.hu/product/CBS220-48P-4X-EU> |
| SFP modul (10Gbps) | <https://www.senetic.hu/product/UACC-CM-RJ45-MG> |
| Router | <https://buyrouterswitch.com/cisco-isr4431-k9-p-5751-price.html> |
| Wifi router | <https://www.mediamarkt.hu/hu/product/_asus-rt-be92u-h%C3%A1roms%C3%A1vos-be9700-wifi-7-aimesh-router-1x10g-wan-lan-fekete-90ig0950-mo9a0v-1465454.html?utm_source=arukereso.hu&utm_medium=psm-product%20feed&utm_content=1465454&aku=d5fb9471a0f52d597f1475a5f1d4f33b> |
| Wifi AP | <https://www.konzolvilag.hu/pc/lan-wifi-asus-dual-band-wifi-6-router-ax5700-mbps-rt-ax86u-pro> |
| ASA | <https://www.router-switch.com/asa5508-k9-p-23193.html> |
| Windows szerver (Main ADDS) | <https://www.senetic.hu/product/EMEA_PER550SPL3?listingSource=servers> |
| Windows szerver | <https://www.senetic.hu/product/EMEA_PER450SPL3?listingSource=servers> |
| Linux szerver | <https://www.senetic.hu/product/EMEA_PER550SPL6?listingSource=servers> |
| NAS | <https://www.senetic.hu/product/TS-1232PXU-RP-4G?listingSource=storage> |
| Háttértár | <https://aqua.hu/alkatreszek/4tb-seagate-35-ironwolf-nas-merevlemez-st4000vn006-t1197148> |
| Nagy rack szekrény | https://aqua.hu/periferiak/conteg-42u-19-i7-allo-rackszekreny-600x1000mm-fekete--ri7-42-60100-h-t207995 |
| **Eszközök** | **Elérés** |
| Kis rack szekrény | https://aqua.hu/periferiak/intellinet-12u-19-fali-rack-szekreny-600x600-712088-t618394 |
| UTP kábel(100m) | https://kabelvilag.hu/Gembird-UTP-fali-dobozos-kabel-100m-CAT6 |
| RJ45-s csatlakozó (100db) | https://www.aliexpress.com/item/1005003304055489.html?spm=a2g0o.productlist.main.11.4c1d5bNd5bNdTT&algo\_pvid=896e1681-d6c6-44c1-a7ae-bdc7a12eaabd&algo\_exp\_id=896e1681-d6c6-44c1-a7ae-bdc7a12eaabd-5&pdp\_npi=4%40dis%21HUF%211656.24%21393.21%21%21%214.17%210.99%21%40210385a817315783758132168ea839%2112000025107419993%21sea%21HU%210%21ABX&curPageLogUid=kM3u3fl81I6e&utparam-url=scene%3Asearch%7Cquery\_from%3A |
| RJ45-aljzat | https://www.aliexpress.com/item/1005003223306465.html?spm=a2g0o.productlist.main.1.748f3i5N3i5NIk&algo\_pvid=55d60275-f5de-446b-9532-a5f5af62ce6b&algo\_exp\_id=55d60275-f5de-446b-9532-a5f5af62ce6b-0&pdp\_npi=4%40dis%21HUF%214464.31%21393.21%21%21%2111.24%210.99%21%402103850917315785195226603efbd2%2112000026896671927%21sea%21HU%210%21ABX&curPageLogUid=GruRMaUx5tk3&utparam-url=scene%3Asearch%7Cquery\_from%3A |
| Számítógépek | <https://www.pcx.hu/?action=prodConfig2/getConfig&processor=854682&processor_piece=1&processorCooler=320078&processorCooler_piece=1&motherboard=774508&motherboard_piece=1&memory=162236&memory_piece=1&ssd=978850&ssd_piece=2&powerSupply=791003&powerSupply_piece=1&computerCase=126826&computerCase_piece=1&systemCooler=676667&systemCooler_piece=3&assembly=1&assembly_piece=1&biosUpdate=-12&biosUpdate_piece=1> |
| Monitor | <https://www.mediamarkt.hu/hu/product/_asus-vy249hf-w-24-s%C3%ADk-fullhd-100-hz-16-9-freesync-ips-led-monitor-1446251.html> |
| Billentyűzet | <https://www.mediamarkt.hu/hu/product/_hama-casano-vezet%C3%A9kes-magyar-billenty%C5%B1zet-53813-1104669.html> |
| Egér | <https://www.mediamarkt.hu/hu/product/_isy-icm-1000-fekete-vezet%C3%A9kes-eg%C3%A9r-1276188.html> |

36. táblázat - Eszközök oldalai

A csapat munkadíja 27 000 000 Ft. Tehát a projekt összköltsége: **59 960 665 Ft**

6. Tesztelés

A tesztelés során a hálózat főbb részeit, illetve a főbb szerver szolgáltatásokat teszteltük.

6.1 Hálózat

A hálózat második, illetve harmadik rétege, valamint LAN és WAN része is tesztelve lett.

6.1.1 Traceroute

A *79. képen* látható egy tracert parancs kimente. A parancs egy, a központban a Zene VLAN-ban található kliensen (10.11.19.5) lett kiadva és célja egy, a debreceni fiókirodában a Kiado VLAN-ban található kliens (10.37.37.5). A parancs sikeressége a hálózat teljes működését bizonyítja.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

79. kép – Teszt: traceroute

6.1.2 VTP

A *80-81. képeken* a show vtp status parancs és annak kimenete mutatja, hogy a KKK-MLS1 VTP szerverként lett konfigurálva, és példaként látszik a KKK-MLS2 kapcsolón, hogy minden más kapcsoló hogyan lett kliensként konfigurálva.

A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

80. kép – Teszt: KKK-MLS1 VTP szerverként

A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

81. kép – Teszt: KKK-MLS2 VTP kliensként

6.1.3 LACP

A *82. képen* a KKK-SW1 kapcsolón a show lacp neighbor parancs kimenete. A parancsban látszik, hogy a KKK-SW1-nek van egy szomszédja passzív módban és két interfésze van a channel group-ban.

A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

82. kép – Teszt: KKK-MLS2 VTP kliensként

6.1.4 HSRP

A *83-84. képeken* látszik a KKK-MLS2 harmadik rétegbeli kapcsolón a show standby neighbors parancs kimenete, amin látszik, hogy a KKK-MLS2 melyik VLAN-okban tölt be aktív, és melyikekben passzív szerepet.

A képen szöveg, menü, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen. A képen szöveg, menü, Betűtípus, képernyőkép látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

83. kép – Teszt: KKK-MLS1 HSRP 1. 84. kép – Teszt: KKK-MLS1 HSRP 2.

6.1.5 BGP

A *85. képen* látható a show ip bgp summary parancs kimenete a DEB-KKK-IR forgalomirányítón. A képen látszik, hogy sikeresen létrejött a BGP szomszédság a routerünk és az ISP forgalomirányítója között.

A képen szöveg, Betűtípus, képernyőkép, szám látható

Automatikusan generált leírás

85. kép – Teszt: DEB-KKK-IR-n BGP ellenőrzése

6.1.6 SSH

A *86. képen* látható a sikeres SSH csatlakozás admin felhasználóval és publikus kulcsos azonosítással az SZE-KKK-BR1 forgalomirányítóra.

A képen szöveg, képernyőkép, Betűtípus, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

86. kép – Teszt: sikeres csatlakozás a SZE-KKK-BR1-hez SSH-n

6.1.7 PPPoE

A *87. képen* látható a show pppoe session parancs kimenete a KKK-IR forgalomirányítón. A parancs kimenetéből az látszik, hogy sikeresen hitelesítette magát a forgalomirányítónk az ISP PPPoE szerverénél.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

87. kép – Teszt: KKK-IR-n PPPoE ellenőrzése

6.1.8 DMVPN, OSPF, EIGRP

A következő képeken láthatóak, hogy sikeresen létrejöttek a DMVPN peer-ek, illetve ezután megérkeztek az OSPF által megtanult, és az EIGRP-be beleforgatott távoli telephelyek alhálózatait tartalmazó routing bejegyzések. A *88-93. képen* az IR forgalomirányítók routing tábláinak egy része, illetve a rajtuk kiadott show dmvpn parancs kimenetei láthatóak.

|  |
| --- |
| A képen szöveg, képernyőkép, Betűtípus, fekete-fehér látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  88. kép – KKK-IR ip routing tábla VPN beállítás után |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  89. kép – KKK-IR DMVPN állapota |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  90. kép – SZE-KKK-IR ip routing tábla VPN beállítás után |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  91. kép – SZE-KKK-IR DMVPN állapota |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  92. kép – DEB-KKK-IR ip routing tábla VPN beállítás után |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  93. kép – DEB-KKK-IR DMVPN állapota |

6.2 Szerverek

A szerverek főbb szolgáltatásait, illetve a felhőben lévő webszervert teszteltük.

6.2.1 Windows szerverek

Az AD DS, a GPO-k, a DFS, az FSRM, a RADIUS, a WDS, a DHCP és a DNS szerver szolgáltatásai lettek tesztelve.

6.2.1.1 DFS

A *94-95. képeken* a DFS szolgáltatás teszteléseként az látható, hogy a két szerveren ugyanúgy létrejött ugyanaz a file, lokális, valamint hálózaton keresztüli elérésnél is.

|  |
| --- |
| *A képen szöveg, képernyőkép, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen. 94*. kép – Replikálódás KKK-ADDS |

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*95. kép – Replikálódás KKK-RODC |

6.2.1.2 FSRM

A *96. képen* az látható, hogy ténylegesen működik egy file screening egy *„a.exe”* nevű fájl másolásával az IT mappába (nem lehetséges).

*A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*96. kép – Fájl másolásos tesztelés

6.2.1.3 RADIUS, WiFi

A *97. képen* látható a szerver önaláírt tanúsítványának elfogadása. A *98. képen* látható a RADIUS szerver által hitelesített felhasználóval való belépés egy vezeték nélküli hálózatot használó telefonról.

A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

97. kép – RADIUS: Tanúsítvány elfogadása

A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

98. kép – RADIUS: Bejelentkezés a kovacs.gabor felhasználóval

6.2.1.4 WDS, DHCP

A *99-100. képen* látható a DHCP kérés és a WDS szolgáltatás elérésének a sikeressége a kiadói hálózatból (10.11.97.2/27).

A képen szöveg, Betűtípus, képernyőkép látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

99. kép – Kiadói alhálózat: IP cím a DHCP szervertől

A képen szöveg, képernyőkép, Betűtípus, fekete látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

100. kép – WDS: boot.wim töltése

6.2.1.5 Kliens tartományba csatlakoztatása, DNS

A *101. képen* látható egy kliens számítógép tartományba való beléptetése, ami az AD környezet, illetve a DNS szolgáltatás sikerességét bizonyítja.

A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

101. kép – Egy kliens domain-ba való beléptetése

6.2.1.6 AD DS bejelentkezés

A *102. képen* látható sikeres bejelentkezés a Varga László felhasználó a marketing alhálózatból.

A képen ég, víz, természet, felhő látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

102. kép – ADDS: Bejelentkezés Varga László felhasználóval

6.2.1.7 Group policy

A *103. képen* látható a Firefox telepítése bejelentkezés előtt.

A képen ég, víz, természet, tengerpart látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

103. kép – GPO: Automatizált szoftvertelepítés

A *104. képen* a beállított háttér és a Feladatkezelő elérhetetlensége látható.

A képen szöveg, képernyőkép, Grafikus tervezés, Grafika látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

104. kép – GPO: Háttérkép, Feladatkezelő

6.2.2 Linux szerverek

Az RSYSLOG, a VPN, az Email, a LibreNMS, illetve a Webszerver lett tesztelve.

6.2.2.1 LibreNMS

A felvett hálózati eszközöket a LibreNMS webfelületén a *105. képen* láthatják. Példaként a *106. képen* látható a KKK-VPN hálózati beállításai.

|  |
| --- |
| *A képen szöveg, képernyőkép, szám, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  105. kép – Hálózati eszközök listája SNMP-ben  *A képen szöveg, képernyőkép, Betűtípus, fekete látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*106. kép – KKK-VPN hálózati beállításai |

6.2.2.2Webszerver

A webszerver egy hitelesítést kér, a *107-111. képeken* a kliens és szerveroldali belépést, illetve sikertelen belépést láthatják. Itt látható, hogy Kovács Gábor az IT csoport tagja sikeresen bejelentkezett és lekérte a weboldal tartalmát, viszont Hajnal Tímeának a vezető csoport tagja nem tudta lekérni a weboldalt, megtagadta a lekérést, és 401-es hibakódra, azaz Unauthorized-re futott a kérés.

|  |
| --- |
| *A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  107. kép – Bejelentkezés Kovács Gábor felhasználóval kliens oldalról |
| *A képen képernyőkép, szöveg, művészet látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  108. kép – Bejelentkezés Kovács Gábor felhasználóval szerver oldalról |
| *A képen szöveg, szoftver, Multimédiás szoftver, képernyőkép látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  109. kép – Bejelentkezés Hajnal Tímea felhasználóval kliens oldalról |
| *A képen szöveg, képernyőkép, Betűtípus, sor látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*110. kép – Hiba a bejelentkezés hiányában |
| 111. kép – Bejelentkezés Hajnal Tímea felhasználóval szerver oldalról |

A *112-113. képeken* az látható, ahogy megadott IPv4 címre megadott felhasználóval, a weboldal elküldi a kódot, amit a jobb oldali textarea-ba írt a felhasználó, valamint a kimenetet ugyanehhez a folyamathoz.

|  |
| --- |
| A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  112. kép – Adatok megadása és küldés |
| A képen szöveg, képernyőkép látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  113. kép – POST kérés kimenete |

A *114. képen* a webszolgáltatás folyamatának a futásának állapota látható.

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*  114. kép – Apache2 státusza |

6.2.2.3 EMAIL

A tesztelés során két felhasználót *hajnal.timea* és *kovacs.gabor* felhasználói fiók segítségével teszteltünk. A kliensek egy oda vissza Reply üzenetet küldenek egymásnak, amikről a képeket *115-117. képeken* láthatják. Ezt követően a *118-120. képekenn* az ezekhez tartozó főleg Postfix, de a lekérésről lévő dovecot bejegyzést is láthatják.

|  |
| --- |
| A képen szöveg, képernyőkép, Betűtípus, sor látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.115. kép – E-mail fogadása |
| A képen szöveg, képernyőkép, szoftver, szám látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  116. kép – Válasz e-mail |
| A képen képernyőkép, szöveg, szoftver látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  117. kép – Válasz e-mail fogadása |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  118. kép – Postfix log első email küldése |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  119. kép – Postfix log válaszemail küldése |
| A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  120. kép – Dovecot log (bejelentkezés levelek lekérése) |

6.2.2.4 RSYSLOG

A tesztelés során újraindítottunk az apache2 szervert a KKK-LIN szerveren, valamint kilistáztuk a */var/log/remote/* mappa tartalmát, ami megmutatja, hogy több log fájl található már itt, aminek van tartalma (későbbiekben lesz még róla szó). Ezeket a *121-122. képeken* láthatják.

|  |
| --- |
| *A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*121. kép - /var/log/remote/ mappa tartalma |
| *A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*122. kép - /var/log/remote/KKK-LIN.log apache2 újraindítás |

6.2.2.5 VPN

A szolgáltatást Kovács Gábor IT-s szakember bejelentkezésével teszteltük. Kliensoldali csatlakozás és logok alapján, valamint az IP beállítások és pingelés alapján mutatjuk be, hogy a VPN kapcsolat sikeres. Ezeket a *123-126. képeken* láthatják.

|  |
| --- |
| A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.  A képen szöveg, képernyőkép, Betűtípus, szám látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.123. kép – Aktív kapcsolat |
| 124. kép – Kliens logjai a kapcsolatról |
| A képen szöveg, képernyőkép, Betűtípus, menü látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.125. kép – IP beállítások a kliensen |
| *A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.* 126. kép – Wireguard interfész a szerveren |

6.2.2.6 RSYSLOG kliens

A szolgáltatás teszteléséhez újraindítottuk az *frr* szolgáltatást, aminek a logjait a KKK-LIN szerverről mutatjuk be. Ezeket az *127-128. képeken* láthatják.

|  |
| --- |
| 127. kép – frr szolgáltatás újraindítása |
| *A képen szöveg, képernyőkép, Betűtípus látható  Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.*128. kép - /var/log/remote/KKK-VPN.log fájl utolsó sorainak megjelenítése |

6.2.3 Felhő

A webkiszolgáló publikus IP címéhez a No-IP (<https://www.noip.com/>) weboldalon egy domain név, a kkk.sytes.net lett hozzárendelve, hogy a vásárlók egyszerűen elérjék a cég weboldalát. Az oldal a következő URL-lel érhető el, ahogy a *129. képen* látható.

A képen szöveg, képernyőkép látható

Automatikusan generált leírás

129. kép - <https://kkk.sytes.net/>

7. Összegzés

A csapat elégedett a projekt teljesítményével, mivel teljes mértékben teljesítette a Kihagyhatatlan Kincsek Kiadó elvárásait. A projekt késznek lett tekintve, illetve át lett adva a minket felbérelt cégnek, akik a munkákkal meg voltak elégedve. A projekt megfelelő ütemben zajlott, illetve a végére sikerült az összes tervünket megvalósítani, így nem csak a cég, de mi is elégedett vagyunk.

8. Hivatkozások

Itt találhatóak a felhasznált PowerShell scriptek és Ansible kódok.

8.1 PowerShell scriptek

8.1.1 Központ

8.1.1.1 OU-kat és csoportokat létrehozó script

# Domain variables

$domain = "kkk.com"

# Organizational Units (OUs) to create

$ouk = @(@("Felhasznalok", "DC=kkk,DC=com"), @("IT", "OU=Felhasznalok,DC=kkk,DC=com"), @("Vezeto", "OU=Felhasznalok,DC=kkk,DC=com"), @("Kiado", "OU=Felhasznalok,DC=kkk,DC=com"), @("Adminisztracio", "OU=Felhasznalok,DC=kkk,DC=com"), @("Zene", "OU=Felhasznalok,DC=kkk,DC=com"), @("Marketing", "OU=Felhasznalok,DC=kkk,DC=com"), @("Csoportok", "OU=Felhasznalok,DC=kkk,DC=com"))

# Groups to create

$csoportok = @(@("Vezeto","ou=Csoportok,ou=Felhasznalok,dc=kkk,dc=com"), @("Kiado","ou=Csoportok,ou=Felhasznalok,dc=kkk,dc=com"), @("Adminisztracio","ou=Csoportok,ou=Felhasznalok,dc=kkk,dc=com"), @("Zene","ou=Csoportok,ou=Felhasznalok,dc=kkk,dc=com"), @("Marketing","ou=Csoportok,ou=Felhasznalok,dc=kkk,dc=com"), @("IT","ou=Csoportok,ou=Felhasznalok,dc=kkk,dc=com"))

# Create the Organizational Units

foreach ($i in $ouk) {

New-ADOrganizationalUnit -Name $i[0] -Path $i[1] -ProtectedFromAccidentalDeletion $False

}

# Create the AD Groups

foreach ($i in $csoportok) {

New-ADGroup -Name $i[0] -Path $i[1] -GroupScope Global

}

# Final confirmation message

Write-Host("Organizational Units and groups created successfully.")

8.1.1.2 Felhasználókat létrehozó script

# Set the password for all new users

$passw = "Password123" | ConvertTo-SecureString -AsPlainText -Force

# Path to the CSV file containing user information

$csvFile = ".\Kozpont\_users.csv"

# Function to remove Hungarian accents and convert to ASCII

function Remove-HungarianAccents {

param (

[string]$InputString

)

# Replace accented characters with their non-accented equivalents

$InputString -replace 'á', 'a' -replace 'é', 'e' -replace 'í', 'i' -replace 'ó', 'o' -replace 'ö', 'o' `

-replace 'ő', 'o' -replace 'ú', 'u' -replace 'ü', 'u' -replace 'ű', 'u' `

-replace 'Á', 'a' -replace 'É', 'e' -replace 'Í', 'i' -replace 'Ó', 'o' -replace 'Ö', 'o' `

-replace 'Ő', 'o' -replace 'Ú', 'u' -replace 'Ü', 'u' -replace 'Ű', 'u'

}

# Import the CSV file

if (Test-Path $csvFile) {

$users = Import-Csv -Path $csvFile

} else {

Write-Error "CSV file not found at $csvFile"

exit

}

$i = 1

$uid = 5000

$gid = 1999

# Loop through each user in the CSV

foreach ($user in $users) {

try {

# Extract data from CSV

$fullName = $user.Name

$ouPath = $user.Path

$title = $user.Title

$groupName = $user.Group

# Split full name into first and last name

$nameParts = $fullName -split ' ', 2

$firstName = $nameParts[1]

$lastName = $nameParts[0]

# Generate username from first and last name, removing Hungarian characters and converting to lowercase

$cleanFirstName = Remove-HungarianAccents -InputString $firstName

$cleanLastName = Remove-HungarianAccents -InputString $lastName

$username = "$($cleanLastName).$($cleanFirstName)".ToLower()

# Generate UserPrincipalName

$userPrincipalName = "$username@kkk.com"

# Check if the user already exists

$existingUser = Get-ADUser -Filter {SamAccountName -eq $username} -ErrorAction SilentlyContinue

if ($existingUser) {

Write-Host "The $username user already exist."

continue

}

# Create the new AD user

New-ADUser -Name $fullName `

-GivenName $firstName `

-Surname $lastName `

-SamAccountName $username `

-UserPrincipalName $userPrincipalName `

-Path $ouPath `

-Title $title `

-AccountPassword $passw `

-Enabled $true `

-ChangePasswordAtLogon $true

Set-ADUser -Identity $Username -Add @{

uid = $uid;

gidNumber = $gid

homeDirectory = "/mailboxes/$username"

}

$uid = $uid + 1

Write-Host "The $username user was created successfully in this OU: $ouPath"

# Search for the group in the determined OU

$groupDN = Get-ADGroup -Filter {Name -eq $groupName} -SearchBase "OU=csoportok,OU=Felhasznalok,DC=kkk,DC=com"

if ($groupDN) {

# Add the user to the group

Add-ADGroupMember -Identity $groupDN -Members $username

Write-Host "$i. The $username user was added to the $groupName group."

$i = $i + 1

} else {

Write-Error "Group $groupName not found in $groupOU."

}

} catch {

Write-Error "Error creating user $fullName : $($\_.Exception.Message)"

}

}

8.1.2 Debrecen

8.1.2.1 OU-kat és csoportokat létrehozó script

# Domain variables

$domain = "deb.kkk.com"

# Organizational Units (OUs) to create

$ouk = @(@("Felhasznalok", "DC=deb,DC=kkk,DC=com"), @("IT", "OU=Felhasznalok,DC=deb,DC=kkk,DC=com"), @("Alkalmazott", "OU=Felhasznalok,DC=deb,DC=kkk,DC=com"), @("Fiokvezeto", "OU=Felhasznalok,DC=deb,DC=kkk,DC=com"), @("Tehetseg", "OU=Felhasznalok,DC=deb,DC=kkk,DC=com"), @("Kiado", "OU=Felhasznalok,DC=deb,DC=kkk,DC=com"), @("Csoportok", "OU=Felhasznalok,DC=deb,DC=kkk,DC=com"))

# Groups to create

$csoportok = @(@("Fiokvezeto","ou=Csoportok,ou=Felhasznalok,dc=deb,dc=kkk,dc=com"), @("Tehetseg","ou=Csoportok,ou=Felhasznalok,dc=deb,dc=kkk,dc=com"), @("Kiado","ou=Csoportok,ou=Felhasznalok,dc=deb,dc=kkk,dc=com"), @("IT","ou=Csoportok,ou=Felhasznalok,dc=deb,dc=kkk,dc=com"))

# Create the Organizational Units

foreach ($i in $ouk) {

New-ADOrganizationalUnit -Name $i[0] -Path $i[1] -ProtectedFromAccidentalDeletion $False

}

# Create the AD Groups

foreach ($i in $csoportok) {

New-ADGroup -Name $i[0] -Path $i[1] -GroupScope Global

}

# Final confirmation message

Write-Host "Organizational Units and Groups created successfully."

8.1.2.2 Felhasználókat létrehozó script

# Set the password for all new users

$passw = "Password123" | ConvertTo-SecureString -AsPlainText -Force

# Path to the CSV file containing user information

$csvFile = "Debrecen\_users.csv"

# Function to remove Hungarian accents and convert to ASCII

function Remove-HungarianAccents {

param (

[string]$InputString

)

# Replace accented characters with their non-accented equivalents

$InputString -replace 'á', 'a' -replace 'é', 'e' -replace 'í', 'i' -replace 'ó', 'o' -replace 'ö', 'o' `

-replace 'ő', 'o' -replace 'ú', 'u' -replace 'ü', 'u' -replace 'ű', 'u' `

-replace 'Á', 'a' -replace 'É', 'e' -replace 'Í', 'i' -replace 'Ó', 'o' -replace 'Ö', 'o' `

-replace 'Ő', 'o' -replace 'Ú', 'u' -replace 'Ü', 'u' -replace 'Ű', 'u'

}

# Import the CSV file

if (Test-Path $csvFile) {

$users = Import-Csv -Path $csvFile

} else {

Write-Error "CSV file not found at $csvFile"

exit

}

$i = 1

$uid = 6000

$gid = 1999

# Loop through each user in the CSV

foreach ($user in $users) {

try {

# Extract data from CSV

$fullName = $user.Name

$ouPath = $user.Path

$title = $user.Title

$groupName = $user.Group

# Split full name into first and last name

$nameParts = $fullName -split ' ', 2

$firstName = $nameParts[1]

$lastName = $nameParts[0]

# Generate username from first and last name, removing Hungarian characters and converting to lowercase

$cleanFirstName = Remove-HungarianAccents -InputString $firstName

$cleanLastName = Remove-HungarianAccents -InputString $lastName

$username = "$($cleanLastName).$($cleanFirstName)".ToLower()

# Generate UserPrincipalName

$userPrincipalName = "$username@deb.kkk.com"

# Check if the user already exists

$existingUser = Get-ADUser -Filter {SamAccountName -eq $username} -ErrorAction SilentlyContinue

if ($existingUser) {

Write-Host "The $username user already exist."

continue

}

# Create the new AD user

New-ADUser -Name $fullName `

-GivenName $firstName `

-Surname $lastName `

-SamAccountName $username `

-UserPrincipalName $userPrincipalName `

-Path $ouPath `

-Title $title `

-AccountPassword $passw `

-Enabled $true `

-ChangePasswordAtLogon $true

Set-ADUser -Identity $Username -Add @{

uid = $uid;

gidNumber = $gid

homeDirectory = "/mailboxes/$username"

}

$uid = $uid + 1

Write-Host "The $username user was created successfully in this OU: $ouPath"

# Search for the group in the determined OU

$groupDN = Get-ADGroup -Filter {Name -eq $groupName} -SearchBase "OU=Csoportok,OU=Felhasznalok,DC=deb,DC=kkk,DC=com"

if ($groupDN) {

# Add the user to the group

Add-ADGroupMember -Identity $groupDN -Members $username

Write-Host "$i. The $username user was added to the $groupName group."

$i = $i + 1

} else {

Write-Error "Group $groupName not found in $groupOU."

}

} catch {

Write-Error "Error creating user $fullName : $($\_.Exception.Message)"

}

}

8.1.3 Szeged

8.1.3.1 OU-kat és csoportokat létrehozó script

# Domain variables

$domain = "sze.kkk.com"

# OUs to create

$ouk = @(@("Felhasznalok", "DC=sze,DC=kkk,DC=com"), @("IT", "OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("Fiokvezeto", "OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("Kozonseg", "OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("Esemeny", "OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("Csoportok", "OU=Felhasznalok,DC=sze,DC=kkk,DC=com"))

# Groups to create

$csoportok = @(@("Fiokvezeto", "OU=Csoportok,OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("Kozonseg", "OU=Csoportok,OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("Esemeny", "OU=Csoportok,OU=Felhasznalok,DC=sze,DC=kkk,DC=com"), @("IT", "OU=Csoportok,OU=Felhasznalok,DC=sze,DC=kkk,DC=com"))

# Create the OUs

foreach ($i in $ouk) {

New-ADOrganizationalUnit -Name $i[0] -Path $i[1] -ProtectedFromAccidentalDeletion $false

}

# Create the groups

foreach ($i in $csoportok) {

New-ADGroup -Name $i[0] -Path $i[1] -GroupScope Global

}

# Final message

Write-Host "OUs and groups created successfully."

8.1.3.2 Felhasználókat létrehozó script

# Set the password for all new users

$passw = "Password123" | ConvertTo-SecureString -AsPlainText -Force

# Path to the CSV file containing user information

$csvFile = ".\Szeged\_users.csv"

# Function to remove Hungarian accents and convert to ASCII

function Remove-HungarianAccents {

param (

[string]$InputString

)

# Replace accented characters with their non-accented equivalents

$InputString -replace 'á', 'a' -replace 'é', 'e' -replace 'í', 'i' -replace 'ó', 'o' -replace 'ö', 'o' `

-replace 'ő', 'o' -replace 'ú', 'u' -replace 'ü', 'u' -replace 'ű', 'u' `

-replace 'Á', 'a' -replace 'É', 'e' -replace 'Í', 'i' -replace 'Ó', 'o' -replace 'Ö', 'o' `

-replace 'Ő', 'o' -replace 'Ú', 'u' -replace 'Ü', 'u' -replace 'Ű', 'u'

}

# Import the CSV file

if (Test-Path $csvFile) {

$users = Import-Csv -Path $csvFile

} else {

Write-Error "CSV file not found at $csvFile"

exit

}

$i = 1

$uid = 7000

$gid = 1999

# Loop through each user in the CSV

foreach ($user in $users) {

try {

# Extract data from CSV

$fullName = $user.Name

$ouPath = $user.Path

$title = $user.Title

$groupName = $user.Group

# Split full name into first and last name

$nameParts = $fullName -split ' ', 2

$firstName = $nameParts[1]

$lastName = $nameParts[0]

# Generate username from first and last name, removing Hungarian characters and converting to lowercase

$cleanFirstName = Remove-HungarianAccents -InputString $firstName

$cleanLastName = Remove-HungarianAccents -InputString $lastName

$username = "$($cleanLastName).$($cleanFirstName)".ToLower()

# Generate UserPrincipalName

$userPrincipalName = "$username@sze.kkk.com"

# Check if the user already exists

$existingUser = Get-ADUser -Filter {SamAccountName -eq $username} -ErrorAction SilentlyContinue

if ($existingUser) {

Write-Host "The $username user already exist."

continue

}

# Create the new AD user

New-ADUser -Name $fullName `

-GivenName $firstName `

-Surname $lastName `

-SamAccountName $username `

-UserPrincipalName $userPrincipalName `

-Path $ouPath `

-Title $title `

-AccountPassword $passw `

-Enabled $true `

-ChangePasswordAtLogon $true

Set-ADUser -Identity $Username -Add @{

uid = $uid;

gidNumber = $gid

homeDirectory = "/mailboxes/$username"

}

$uid = $uid + 1

Write-Host "The $username user was created successfully in this OU: $ouPath"

# Search for the group in the determined OU

$groupDN = Get-ADGroup -Filter {Name -eq $groupName} -SearchBase "ou=Csoportok,ou=Felhasznalok,dc=sze,dc=kkk,dc=com"

if ($groupDN) {

# Add the user to the group

Add-ADGroupMember -Identity $groupDN -Members $username

Write-Host "$i. The $username user was added to the $groupName group.."

$i = $i + 1

} else {

Write-Error "Group $groupName not found in $groupOU."

}

} catch {

Write-Error "Error creating user $fullName : $($\_.Exception.Message)"

}

}

8.2 Ansible kódok

8.2.1 Addfs.yaml playbook

---

- name: Install AD DFS

hosts: server

gather\_facts: false

become: true

become\_method: runas

tasks:

- name: Install AD DFS

ansible.windows.win\_feature:

name:

- FS-DFS-Namespace

- FS-DFS-Replication

state: present

include\_management\_tools: true

register: addfs\_feature\_result

- name: Reboot the server if needed

ansible.windows.win\_reboot:

when: addfs\_feature\_result.reboot\_required

- name: Create directories

ansible.windows.win\_file:

path: "{{ item.path }}"

state: "{{ item.state }}"

loop: "{{ addfs }}"

loop\_control:

label: "{{ item.path }}"

- name: Create SMB shares

ansible.windows.win\_share:

name: "{{ item.name }}"

path: "{{ item.path }}"

full: "{{ item.full }}"

change: "{{ item.change }}"

read: "{{ item.read }}"

loop: "{{ addfs }}"

loop\_control:

label: "{{ item.path }}"

- name: Configure AD DFS

hosts: KKK-ADDS

gather\_facts: false

become: true

become\_method: runas

tasks:

- name: Check namespace root

ansible.windows.win\_shell: '(Get-DfsnRoot | Where-Object { $\_.Path -like "\\kkk.com\Files" }) -ne $null'

register: dfsn

changed\_when: '"True" not in dfsn.stdout'

notify: Configure DFS

handlers:

- name: Create namespace root

ansible.windows.win\_shell: 'New-DfsnRoot -Path "\\kkk.com\Files" -TargetPath "\\KKK-ADDS.kkk.com\Files" -Type DomainV2'

listen: Configure DFS

- name: Create namespace folders

ansible.windows.win\_shell: |

New-DfsnFolder -Path "\\kkk.com\Files\Adminisztracio" -TargetPath "\\KKK-ADDS.kkk.com\Adminisztracio"

New-DfsnFolder -Path "\\kkk.com\Files\Marketing" -TargetPath "\\KKK-ADDS.kkk.com\Marketing"

New-DfsnFolder -Path "\\kkk.com\Files\Zene" -TargetPath "\\KKK-ADDS.kkk.com\Zene"

New-DfsnFolder -Path "\\kkk.com\Files\It" -TargetPath "\\KKK-ADDS.kkk.com\It"

New-DfsnFolder -Path "\\kkk.com\Files\KKK" -TargetPath "\\KKK-ADDS.kkk.com\KKK"

New-DfsnFolder -Path "\\kkk.com\Files\Vezeto" -TargetPath "\\KKK-ADDS.kkk.com\Vezeto"

New-DfsnFolder -Path "\\kkk.com\Files\Kiado" -TargetPath "\\KKK-ADDS.kkk.com\Kiado"

listen: Configure DFS

- name: Add folder secondary paths

ansible.windows.win\_shell: |

New-DfsnFolderTarget -Path "\\kkk.com\Files\Adminisztracio" -TargetPath "\\KKK-RODC.kkk.com\Adminisztracio" -State Online

New-DfsnFolderTarget -Path "\\kkk.com\Files\Marketing" -TargetPath "\\KKK-RODC.kkk.com\Marketing" -State Online

New-DfsnFolderTarget -Path "\\kkk.com\Files\Zene" -TargetPath "\\KKK-RODC.kkk.com\Zene" -State Online

New-DfsnFolderTarget -Path "\\kkk.com\Files\It" -TargetPath "\\KKK-RODC.kkk.com\It" -State Online

New-DfsnFolderTarget -Path "\\kkk.com\Files\KKK" -TargetPath "\\KKK-RODC.kkk.com\KKK" -State Online

New-DfsnFolderTarget -Path "\\kkk.com\Files\Vezeto" -TargetPath "\\KKK-RODC.kkk.com\Vezeto" -State Online

New-DfsnFolderTarget -Path "\\kkk.com\Files\Kiado" -TargetPath "\\KKK-RODC.kkk.com\Kiado" -State Online

listen: Configure DFS

- name: Create replication group

ansible.windows.win\_shell: New-DfsReplicationGroup -GroupName "KKK-REPLICATION"

listen: Configure DFS

- name: Add servers to replication group

ansible.windows.win\_shell: |

Add-DfsrMember -GroupName "KKK-REPLICATION" -ComputerName "KKK-ADDS.kkk.com"

Add-DfsrMember -GroupName "KKK-REPLICATION" -ComputerName "KKK-RODC.kkk.com"

listen: Configure DFS

- name: Set replicated folder

ansible.windows.win\_shell: |

New-DfsReplicatedFolder -GroupName "KKK-REPLICATION" -FolderName Files,It,Marketing,Kiado,Adminisztracio,Vezeto,Zene,KKK

listen: Configure DFS

- name: Create replication connection

ansible.windows.win\_shell: >

Add-DfsrConnection

-GroupName "KKK-REPLICATION"

-SourceComputerName "KKK-ADDS.kkk.com"

-DestinationComputerName "KKK-RODC.kkk.com"

listen: Configure DFS

- name: Set local path

ansible.windows.win\_shell: >

Set-DfsrMembership

-Groupname "KKK-REPLICATION"

-FolderName "{{ item.name }}"

-ComputerName "KKK-ADDS.kkk.com"

-ContentPath "{{ item.path }}"

-PrimaryMember $True

-StagingPathQuotaInMB 16384

-Force

Set-DfsrMembership

-Groupname "KKK-REPLICATION"

-FolderName "{{ item.name }}"

-ComputerName "KKK-RODC.kkk.com"

-ContentPath "{{ item.path }}"

-Force

loop: "{{ addfs }}"

loop\_control:

label: "{{ item.name }}"

listen: Configure DFS

8.2.2 Adds.yaml playbook

---

- name: KKK-ADDS domain creation

hosts: KKK-ADDS

gather\_facts: false

vars:

ansible\_user: Administrator

ansible\_become\_user: Administrator

tasks:

- name: Create KKK domain

microsoft.ad.domain:

safe\_mode\_password: Password123

dns\_domain\_name: kkk.com

register: domain

- name: Reboot

win\_reboot:

when: domain.reboot\_required

- name: Domain Controller Promotion

microsoft.ad.domain\_controller:

dns\_domain\_name: kkk.com

domain\_admin\_user: KKK\Administrator

domain\_admin\_password: Password123

state: domain\_controller

safe\_mode\_password: Password123

register: dc

- name: Reboot

win\_reboot:

when: dc.reboot\_required

- name: Configure DNS Server on KKK-RODC

hosts: KKK-RODC

gather\_facts: false

vars:

ansible\_user: Administrator

ansible\_become\_user: Administrator

desired\_dns\_servers:

- 10.11.113.10

- "2001:db8:c1c1:abfa::10"

tasks:

- name: Check current DNS server addresses

win\_shell: |

$adapter = Get-DnsClientServerAddress | Where-Object { $\_.InterfaceAlias -eq 'Ethernet0' }

$dnsAddresses = $adapter.ServerAddresses

$dnsAddresses | Sort-Object | ForEach-Object { $\_.Trim() }

register: current\_dns

changed\_when: false

- name: Set DNS server addresses

win\_dns\_client:

adapter\_names: 'Ethernet0'

dns\_servers: "{{ desired\_dns\_servers }}"

when: "'10.11.113.10' not in current\_dns.stdout\_lines and '2001:db8:c1c1:abfa::10' not in current\_dns.stdout\_lines"

changed\_when: true

- name: Join KKK-RODC to domain

hosts: KKK-RODC

gather\_facts: false

vars:

ansible\_user: Administrator

ansible\_become\_user: Administrator

tasks:

- name: Joining domain

microsoft.ad.membership:

domain\_admin\_user: KKK\Administrator

domain\_admin\_password: Password123

state: domain

dns\_domain\_name: kkk.com

register: domain

- name: Reboot

win\_reboot:

when: domain.reboot\_required

- name: Promoting KKK-RODC to RODC

hosts: KKK-RODC

vars:

ansible\_user: Administrator

ansible\_become\_user: Administrator

gather\_facts: false

tasks:

- name: Installing ADDS

win\_feature:

name: AD-Domain-Services

include\_management\_tools: True

include\_all\_subfeatures: True

- name: Promoting to RODC

microsoft.ad.domain\_controller:

dns\_domain\_name: kkk.com

domain\_admin\_user: KKK\Administrator

domain\_admin\_password: Password123

state: domain\_controller

safe\_mode\_password: Password123

read\_only: True

site\_name: Default-First-Site-Name

register: rodc

- name: Reboot

win\_reboot:

when: rodc.reboot\_required

8.2.3 Dhcp.yaml playbook

---

- name: DHCP configuration

hosts: KKK-ADDS

gather\_facts: false

tasks:

- name: Install DHCP feature and management tools

ansible.windows.win\_feature:

name: DHCP, RSAT-DHCP

state: present

include\_management\_tools: true

register: dhcp\_feature\_result

- name: Reboot the server if needed

ansible.windows.win\_reboot:

when: dhcp\_feature\_result.reboot\_required

- name: Check if DHCP module is available

ansible.windows.win\_shell: |

if (Get-Module -ListAvailable -Name DhcpServer) {

Write-Output "Module Found"

} else {

Write-Output "Module Not Found"

}

register: module\_check

changed\_when: false

- name: Authorize DHCP server to domain

ansible.windows.win\_shell: |

if (!(Get-DhcpServerInDC | Where-Object { $\_.DnsName -eq "KKK-ADDS.kkk.com" })) {

Add-DhcpServerInDC -DnsName "KKK-ADDS.kkk.com" -IPAddress "10.11.113.10"

}

changed\_when: false

- name: Notify server manager that Post-install is done

ansible.windows.win\_shell: |

$currentState = (Get-ItemProperty -Path registry::HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\ServerManager\Roles\12 -Name ConfigurationState).ConfigurationState

if ($currentState -ne 2) {

Set-ItemProperty -Path registry::HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\ServerManager\Roles\12 -Name ConfigurationState -Value 2

}

when: dhcp\_feature\_result.changed or not dhcp\_feature\_result.changed

changed\_when: false

- name: Register clients to DNS server

ansible.windows.win\_shell: |

$credential = New-Object System.Management.Automation.PSCredential("KKK\Administrator", (ConvertTo-SecureString "Passw0rd" -AsPlainText -Force))

$existingCredential = Get-DhcpServerDnsCredential -ErrorAction SilentlyContinue

if (!$existingCredential) {

Set-DhcpServerDnsCredential -Credential $credential -ComputerName "KKK-ADDS.kkk.com"

}

when: dhcp\_feature\_result.changed or not dhcp\_feature\_result.changed

changed\_when: false

- name: Configure IPv4 scopes

win\_shell: |

$scope = Get-DhcpServerv4Scope -ScopeId "{{ item.scopeid }}" -ErrorAction SilentlyContinue

if (-not $scope) {

Add-DhcpServerv4Scope -Name "{{ item.name }}" -StartRange "{{ item.start }}" -EndRange "{{ item.end }}" -SubnetMask "{{ item.netmask }}" -State Active

Set-DhcpServerv4OptionValue `

-ScopeId "{{ item.scopeid }}" `

-ComputerName "KKK-ADDS.kkk.com" `

-DnsServer "{{ item.dns }}" `

-WinsServer "{{ item.wins }}" `

-DnsDomain "kkk.com" `

-Router "{{ item.gateway }}"

} else {

Write-Host "Scope {{ item.scopeid }} already exists, skipping creation."

}

loop: "{{ dhcp.scopev4 }}"

loop\_control:

label: "{{ item.name }}"

changed\_when: false

- name: Configure IPv6 scopes

ansible.windows.win\_shell: |

$prefix = "{{ item.prefix }}"

$exstart = "{{ item.exstart }}"

$exend = "{{ item.exend }}"

if (!(Get-DhcpServerv6Scope -Prefix "{{ item.prefix }}" -ErrorAction SilentlyContinue)) {

Add-DhcpServerv6Scope -Prefix "{{ item.prefix }}" -Name "{{ item.name }}"

Add-DhcpServerv6ExclusionRange -Prefix $prefix -StartRange $exstart -EndRange $exend

}

loop: "{{ dhcp.scopev6 }}"

loop\_control:

label: "{{ item.name }}"

when: dhcp\_feature\_result.changed or not dhcp\_feature\_result.changed

changed\_when: false

8.2.4 Dns.yaml playbook

---

- name: Add KKK-RODC address to KKK-ADDS dns servers

hosts: KKK-ADDS

gather\_facts: false

vars:

ansible\_user: Administrator

ansible\_become\_user: Administrator

desired\_dns\_servers:

- 10.11.113.10

- "2001:db8:c1c1:abfa::10"

- 10.11.113.11

- "2001:db8:c1c1:abfa::11"

tasks:

- name: Check current DNS server addresses

win\_shell: |

$adapter = Get-DnsClientServerAddress | Where-Object { $\_.InterfaceAlias -eq 'Ethernet0' }

$dnsAddresses = $adapter.ServerAddresses

$dnsAddresses | Sort-Object | ForEach-Object { $\_.Trim() }

register: current\_dns

changed\_when: false

- name: Set DNS server addresses

win\_dns\_client:

adapter\_names: 'Ethernet0'

dns\_servers: "{{ desired\_dns\_servers }}"

when: "'10.11.113.11' not in current\_dns.stdout\_lines and '2001:db8:c1c1:abfa::11' not in current\_dns.stdout\_lines"

changed\_when: true

- name: DNS zone and record creation

hosts: KKK-ADDS

gather\_facts: false

tasks:

- name: Check if DNS zone exists

win\_shell: |

if (Get-DnsServerZone -Name 113.11.10.in-addr.arpa -ErrorAction SilentlyContinue) {

Write-Output "ZoneExists"

} else {

Write-Output "ZoneDoesNotExist"

}

register: dns\_zone\_check

changed\_when: false

- name: Check if DNS A record exists

win\_shell: "nslookup 10.11.113.5"

register: nslookup\_check

changed\_when: false

- name: Check if DNS AAAA record exists

win\_shell: |

nslookup 2001:db8:c1c1:abfa::5

register: nslookup\_aaaa\_check

changed\_when: false

- name: Check if CNAME record exists

win\_shell: |

nslookup -type=CNAME mail.kkk.com

register: nslookup\_cname\_check

changed\_when: false

- name: Check if DNS forwarder exists

win\_shell: |

$forwarder = Get-DnsServerForwarder

if ($forwarder) {

Write-Output "ForwarderExists"

} else {

Write-Output "ForwarderDoesNotExist"

}

register: dns\_forwarder\_check

changed\_when: false

- name: Create DNS zones

win\_shell: >

Add-DnsServerPrimaryZone -Name "{{ item.name }}" -ReplicationScope "Forest"

loop: "{{ dns.reversezones }}"

when: '"ZoneDoesNotExist" in dns\_zone\_check.stdout'

- name: Create A records

win\_shell: >

Add-DnsServerResourceRecordA -ZoneName "{{item.zonename}}" -Name "{{ item.name }}" -IPv4Address "{{ item.value }}" -CreatePtr

loop: "{{ dns.a\_records }}"

when: "'kkk-lin.kkk.com' not in nslookup\_check.stdout"

- name: Create AAAA records

win\_shell: >

Add-DnsServerResourceRecordAAAA -ZoneName "{{item.zonename}}" -Name "{{ item.name }}" -IPv6Address "{{ item.value }}" -CreatePtr

loop: "{{ dns.aaaa\_records }}"

when: "'kkk-lin.kkk.com' not in nslookup\_aaaa\_check.stdout"

- name: Create CNAME records

win\_shell: >

Add-DnsServerResourceRecordCNAME -ZoneName "{{item.zonename}}" -Name "{{ item.name }}" -HostNameAlias "{{ item.value }}"

loop: "{{ dns.cname\_records }}"

when: "'kkk-lin.kkk.com' not in nslookup\_cname\_check.stdout"

- name: Add DNS forwarder

win\_shell: >

Add-DnsServerForwarder -IPAddress 8.8.8.8 -PassThru

when: '"ForwarderDoesNotExist" in dns\_forwarder\_check.stdout'

8.2.5 Fsrm.yaml playbook

- name: Install and configure FSRM

hosts: server

gather\_facts: false

tasks:

- name: Install FSRM

ansible.windows.win\_feature:

name: FS-Resource-Manager

state: present

include\_management\_tools: true

notify:

- Create FSRM quotas

- Create file screen groups

- Create file screens

handlers:

- name: Create FSRM quotas

ansible.windows.win\_powershell:

script: |

New-FsrmQuota -Path "C:\Adminisztracio" -Description "Limit usage to 50 GB." -Size 50GB

New-FsrmQuota -Path "C:\It" -Description "Limit usage to 100 GB." -Size 100GB

New-FsrmQuota -Path "C:\Kiado" -Description "Limit usage to 50 GB." -Size 50GB

New-FsrmQuota -Path "C:\Marketing" -Description "Limit usage to 80 GB." -Size 80GB

New-FsrmQuota -Path "C:\Vezeto" -Description "Limit usage to 20 GB." -Size 20GB

New-FsrmQuota -Path "C:\Zene" -Description "Limit usage to 300 GB." -Size 300GB

New-FsrmQuota -Path "C:\KKK" -Description "Limit usage to 50 GB." -Size 50GB

- name: Create file screen groups

ansible.windows.win\_powershell:

script: |

New-FsrmFileGroup -Name "G-Adminisztracio" -IncludePattern "\*.\*" -ExcludePattern @("\*.doc", "\*.docx", "\*.xls", "\*.xlsx", "\*.csv", "\*.pdf", "\*.txt", "\*.ppt", "\*.pptx", "\*.jpg", "\*.jpeg")

New-FsrmFileGroup -Name "G-It" -IncludePattern "\*.exe"

New-FsrmFileGroup -Name "G-Kiado" -IncludePattern "\*.\*" -ExcludePattern @("\*.doc", "\*.docx", "\*.jpg", "\*.pdf", "\*.txt", "\*.ppt", "\*.pptx", "\*.mp3", "\*.wav", "\*.mp4", "\*.avi", "\*.xcf", "\*.psd")

New-FsrmFileGroup -Name "G-Marketing" -IncludePattern "\*.\*" -ExcludePattern @("\*.doc", "\*.docx", "\*.jpg", "\*.webp", "\*.jfif", "\*.png", "\*.pdf", "\*.txt", "\*.ppt", "\*.pptx", "\*.mp4", "\*.avi", "\*.xcf", "\*.psd", "\*.html", "\*.css", "\*.htm", "\*.php", "\*.js", "\*.ts")

New-FsrmFileGroup -Name "G-Vezeto" -IncludePattern "\*.\*" -ExcludePattern @("\*.doc", "\*.docx", "\*.xls", "\*.xlsx", "\*.csv", "\*.pdf", "\*.txt", "\*.ppt", "\*.pptx", "\*.jpg", "\*.jpeg")

New-FsrmFileGroup -Name "G-Zene" -IncludePattern @("\*.mp3", "\*.wav", "\*.flac", "\*.aac", "\*.ogg", "\*.mid", "\*.als", "\*.ptf", "\*.ptx", "\*.docx", "\*.pptx")

New-FsrmFileGroup -Name "G-KKK" -IncludePattern "\*.\*" -ExcludePattern @("\*.docx", "\*.pptx", "\*.pdf", "\*.txt", "\*.xlsx")

- name: Create file screens

ansible.windows.win\_powershell:

script: |

New-FsrmFileScreen -Path "C:\Marketing" -Description "Marketing" -IncludeGroup "G-Marketing" -Active

New-FsrmFileScreen -Path "C:\Adminisztracio" -Description "Adminisztracio" -IncludeGroup "G-Adminisztracio" -Active

New-FsrmFileScreen -Path "C:\It" -Description "It" -IncludeGroup "G-It" -Active

New-FsrmFileScreen -Path "C:\Kiado" -Description "Kiado" -IncludeGroup "G-Kiado" -Active

New-FsrmFileScreen -Path "C:\Vezeto" -Description "Vezeto" -IncludeGroup "G-Vezeto" -Active

New-FsrmFileScreen -Path "C:\Zene" -Description "Zene" -IncludeGroup "G-Zene" -Active

New-FsrmFileScreen -Path "C:\KKK" -Description "KKK" -IncludeGroup "G-KKK" -Active

8.2.6 Inventory.yaml playbook

all:

vars:

ansible\_user: Administrator@KKK.COM

ansible\_password: Password123

ansible\_connection: winrm

ansible\_port: 5985

ansible\_winrm\_transport: basic

ansible\_winrm\_server\_cert\_validation: ignore

ansible\_become\_method: runas

ansible\_become\_user: KKK\Administrator

ansible\_become\_password: Password123

addfs:

- name: Files

path: C:\Files

state: directory

full: IT

change: IT

read: Everyone

- name: Adminisztracio

path: C:\Adminisztracio

state: directory

full: IT

change: Adminisztracio

read: Vezeto

- name: It

path: C:\It

state: directory

full: IT

change: IT

read: Vezeto

- name: Kiado

path: C:\Kiado

state: directory

full: IT

change: Kiado

read: Vezeto

- name: Marketing

path: C:\Marketing

state: directory

full: IT

change: Marketing

read: Vezeto

- name: Vezeto

path: C:\Vezeto

state: directory

full: IT

change: Vezeto

read: Vezeto

- name: Zene

path: C:\Zene

state: directory

full: IT

change: Zene

read: Vezeto

- name: KKK

path: C:\KKK

state: directory

full: IT

change: IT

read: Everyone

dhcp:

scopev4:

- name: "ADMINISZTRACIO"

start: "10.11.47.2"

end: "10.11.47.28"

netmask: "255.255.255.224"

scopeid: "10.11.47.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.47.1"

- name: "MARKETING"

start: "10.11.73.2"

end: "10.11.73.60"

netmask: "255.255.255.192"

scopeid: "10.11.73.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.73.1"

- name: "VEZETO"

start: "10.11.71.2"

end: "10.11.71.12"

netmask: "255.255.255.240"

scopeid: "10.11.71.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.71.1"

- name: "KIADO"

start: "10.11.97.2"

end: "10.11.97.28"

netmask: "255.255.255.224"

scopeid: "10.11.97.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.97.1"

- name: "ZENE"

start: "10.11.19.2"

end: "10.11.19.60"

netmask: "255.255.255.192"

scopeid: "10.11.19.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.19.1"

- name: "WIFI"

start: "10.11.140.2"

end: "10.11.143.250"

netmask: "255.255.252.0"

scopeid: "10.11.140.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.140.1"

- name: "IT"

start: "10.11.169.2"

end: "10.11.169.12"

netmask: "255.255.255.240"

scopeid: "10.11.169.0"

dns: "10.11.113.10"

wins: "10.11.113.10"

gateway: "10.11.169.1"

scopev6:

- name: "ADMINISZTRACIO"

exstart: "2001:db8:c1c1:abb1:1::"

exend: "2001:db8:c1c1:abb1:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:abb1::"

- name: "MARKETING"

exstart: "2001:db8:c1c1:abf8:1::"

exend: "2001:db8:c1c1:abf8:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:abf8::"

- name: "VEZETO"

exstart: "2001:db8:c1c1:abba:1::"

exend: "2001:db8:c1c1:abba:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:abba::"

- name: "KIADO"

exstart: "2001:db8:c1c1:abf6:1::"

exend: "2001:db8:c1c1:abf6:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:abf6::"

- name: "ZENE"

exstart: "2001:db8:c1c1:ab8d:1::"

exend: "2001:db8:c1c1:ab8d:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:ab8d::"

- name: "WIFI"

exstart: "2001:db8:c1c1:abda:1::"

exend: "2001:db8:c1c1:abda:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:abda::"

- name: "IT"

exstart: "2001:db8:c1c1:abb5:1::"

exend: "2001:db8:c1c1:abb5:ffff:ffff:ffff:ffff"

prefix: "2001:db8:c1c1:abb5::"

dns:

reversezones:

- networkid: "10.11.113.0/24"

name: "113.11.10.in-addr.arpa"

- networkid: "10.11.199.0/24"

name: "199.11.10.in-addr.arpa"

- networkid: "2001:db8:c1c1:abfa::/64"

name: "a.f.b.a.1.c.1.c.8.b.d.0.1.0.0.2.ip6.arpa"

- networkid: "2001:db8:c1c1:ab22::/64"

name: "2.2.b.a.1.c.1.c.8.b.d.0.1.0.0.2.ip6.arpa”

a\_records:

- name: "kkk-lin"

value: "10.11.113.5"

zonename: "kkk.com"

- name: "kkk-vpn"

value: "10.11.199.2"

zonename: "kkk.com"

- name: "kkk-backup"

value: "10.11.113.7"

zonename: "kkk.com"

reverse\_a\_records:

- name: "kkk-adds"

value: "10.11.113.5"

zonename: "kkk.com"

- name: "kkk-vpn"

value: "10.11.199.2"

zonename: "kkk.com"

- name: "kkk-backup"

value: "10.11.113.7"

zonename: "kkk.com"

reverse\_aaaa\_records:

- name: "kkk-lin"

value: "10.11.113.5"

zonename: "kkk.com"

- name: "kkk-vpn"

value: "10.11.199.2"

zonename: "kkk.com"

- name: "kkk-backup"

value: "10.11.113.7"

zonename: "kkk.com"

aaaa\_records:

- name: "kkk-lin"

value: "2001:db8:c1c1:abfa::5"

zonename: "kkk.com"

- name: "kkk-backup"

value: "2001:db8:c1c1:abfa::7"

zonename: "kkk.com"

- name: "kkk-vpn"

value: "2001:db8:c1c1:ab22::2"

zonename: "kkk.com"

cname\_records:

- name: "mail"

value: "kkk-lin.kkk.com"

zonename: "kkk.com"

- name: "log"

value: "kkk-lin.kkk.com"

zonename: "kkk.com"

- name: "python"

value: "kkk-lin.kkk.com"

zonename: "kkk.com"

- name: "radius"

value: "kkk-adds.kkk.com"

zonename: "kkk.com"

- name: "file"

value: "kkk-adds.kkk.com"

zonename: "kkk.com"

- name: "vpn"

value: "kkk-vpn.kkk.com"

zonename: "kkk.com"

mainhq:

children:

server:

hosts:

KKK-ADDS:

ansible\_host: 10.11.113.10

hostname: KKK-ADDS

KKK-RODC:

ansible\_host: 10.11.113.11

hostname: KKK-RODC