

VISOKO UČILIŠTE ALGEBRA

PROJEKTNI ZADATAK

Sistemska inženjerstvo - praktikum

Antonio Janach

Zagreb, svibanj 2021.

Sadržaj

1.	Uvod	1
2.	Zahtjevi infrastrukture	2
2.1.	Virtualne mašine po lokacijama	2
2.2.	Mrežna konfiguracija po lokacijama	3
2.3.	Konfiguracija funkcionalnosti po virtualnim mašinama	3
3.	Opis infrastrukture	6
4.	Topologija infrastrukture	7
5.	Priprema infrastrukture	8
5.1.	Mrežna konfiguracija po lokacijama	8
6.	Razrada projekta - projektno rješenje	9
6.1.	Podizanje domene na lokaciji A	9
6.2.	Dodavanje ostalih Windows računala u domenu na lokaciji A	10
6.3.	Postavljanje trajnog DNS namespace-a na Ubuntu poslužitelj	11
6.4.	Instalacija root CA i subordinate CA – Server1 i Server2	12
6.4.1.	Kreiranje certifikata za Web poslužitelj	13
6.4.2.	Kreiranje certifikata za autentifikaciju korisnika	14
6.6.	Konfiguracija NTP server na ServerDC	18
6.7.	VPN konfiguracija između Ubuntu poslužitelja	20
6.7.1.	OpenVPN konfiguracija na server strani	21
6.7.2.	OpenVPN konfiguracija na klijentskoj strani	24
6.9.	Konfiguracija domene na lokaciji B	27
6.10.	Konfiguracija DHCP poslužitelja na SERVERDC i RODC	29
6.10.1.	Konfiguracija DHCP poslužitelja na SERVERDC	29
6.10.2.	Konfiguracija DHCP poslužitelja na RODC	30
6.11.	Storage spaces konfiguracija i konfiguracija deduplikacije na Server1 i Server2	31
6.12.	DFS + R konfiguracija na SERVER3 i SERVER4 poslužitelju	32
6.13.	Diskovna i iSCSI konfiguracija na Linux1 poslužitelju	34
6.14.	Windows File Server Cluster konfiguracija na Server1 i Server2	36
6.15.	Konfiguracija CentOS poslužitelja - lokacija A	40
6.15.1.	Dodavanje FreeIPA servisa i sync s Active Directory-om	40
6.15.2.	Instalacija Samba servisa i postavljanje inkrementalnog backup-a	44
6.15.3.	Wordpress instalacija na Linux1 poslužitelju	50
6.15.4.	Konfiguracija mail servera na Linux 1 poslužitelj	54
6.16.	Konfiguracija CentOS poslužitelja - lokacija B	57
6.16.1.	Wordpress instalacija na Linux2 poslužitelju	57
6.17.	Slanje log zapisa s Linux2 na Linux1 poslužitelj	61
7.	Zaključak	62
	Popis slika	63
	Literatura	65

1. Uvod

Potrebno je podići infrastrukturu na virtualnoj okolini tako da se međusobno povežu lokacija A i lokacija B koristeći VPN. Za poslužitelje koji se nalaze na lokacijama potrebno je podići sve servise i zahtjeve kako bi se zadovoljili svi uvjeti.

2. Zahtjevi infrastrukture

2.1. Virtualne mašine po lokacijama

Tvrtka Ime Prezime Usluge d.o.o. podiže infrastrukturu od nule, na svoje dvije lokacije (nazovimo ih Lokacija A i Lokacija B). Svaki student za sebe je sistem inženjer zadužen za implementaciju svih koraka projekta. Potrebno je napraviti deployment sljedećih virtualnih računala na Lokaciji A:

- OpenVPN1 – Linux računalo (Ubuntu 20.10)
- Linux1 – Linux računalo (CentOS 8.3)
- ServerDC – Windows 2019 poslužitelj
- Server1 – Windows 2019 poslužitelj
- Server2 – Windows 2019 poslužitelj
-

Isto tako, potrebno je napraviti deployment slijedećih virtualnih računala na Lokaciji B:

- OpenVPN2 – Linux računalo (Ubuntu 20.10)
- RODC – Windows poslužitelj
- Server3 – Windows 2019 poslužitelj Server 4 – Windows 2019 poslužitelj
- Linux2 – Linux računalo (CentOS 8.3)

Izvan lokacije imamo još jedno računalo, Client1 – Windows 10 desktop. To ćemo računalo koristiti za pristup virtualkama na Lokacijama A i B i potencijalno istraživanje problema. Prije spajanja na OpenVPN strojeve, Client1 treba preimenovati u Windowsima u client1-iprezime.

2.2. Mrežna konfiguracija po lokacijama

OpenVPN mašine imaju tri virtualne mrežne kartice. Prva virtualna mrežna kartica će se koristiti za site-to-site VPN između OpenVPN1 i OpenVPN2. Druga virtualna mreža je intra-site virtualna mreža za komunikaciju s ostalim virtualnim mašinama (Windows, Linux). Treća mrežna kartica je spojena na Internet. Koristite ju isključivo i samo za instalaciju i update, nakon čega ju je potrebno isključiti u OS-u.

Primjera radi, druga virtualna mrežna kartica u OpenVPN1 koristi se za komunikaciju sa Linux1, ServerDC, Server1 i SERVER2.

ServerDC i RODC imaju po jednu mrežnu karticu, za intra-site komunikaciju s preostalim virtualnim mašinama na svakoj od lokacija.

Server1, Server2, Server3 i Server4 imaju po dvije mrežne kartice. Prva mrežna kartica je za intra-site komunikaciju. Druga mrežna kartica je za komunikaciju s pripadajućim parom u failover clusteru.

Linux1 i Linux2 mašine imaju po dvije mrežne kartice. Prva je za intra-site komunikaciju, a druga je spojena na Internet. Drugu mrežnu karticu koristite isključivo i samo za instalaciju i update, nakon čega ju je potrebno isključiti u OS-u.

Client1 mašina ima tri mrežne kartice. Prva mrežna kartica je spojena na site-to-site mrežu između dvije lokacije i mora moći pingati OpenVPN1 i OpenVPN2. Druga mrežna kartica je u mreži od Lokacije A. Treća mrežna kartica je u mreži od Lokacije B. Te dvije kartice trebaju biti isključene, osim za potrebe testiranja funkcionalnosti na Lokacijama A i B. Client1 mašina ne smije ni u kojem trenutku biti preko OpenVPN mašina biti spojena na Internet.

Mašine na Lokaciji A kao default GW koriste OpenVPN1. Mašine na Lokaciji B kao default GW koriste OpenVPN2

2.3. Konfiguracija funkcionalnosti po virtualnim mašinama

OpenVPN1 i OpenVPN2 – OpenVPN poslužitelji za Lokaciju A i B. Između njih je potrebno konfigurirati site-to-site OpenVPN korištenjem maksimalne sigurnosne razine koju OpenVPN nudi. Ovi poslužitelji moraju dopuštati routing subneta na Lokaciji A prema Lokaciji B i obrnuto. Prevedeno, virtualne mašine na Lokaciji A preko OpenVPN1 moraju moći pristupiti svim virtualnim mašinama na Lokaciji B, za sve potrebne protokole i servise koji su zadani projektom.

Obje mašine moraju imati konfiguriran firewall tako da propušta minimalnu potrebnu razinu servisa i prometa. Pošto se radi o Ubuntu mašinama, deinstalirajte ufw i instalirajte firewalld kao traženo firewall rješenje. Također, OpenVPN1 i OpenVPN2 moraju imati instaliran nginx ili HAproxy, kako odaberete. Koje god rješenje od ta dva odaberete, trebaju raditi reverse proxying/load balancing HTTPS prometa na L1 i L2 mašine na portu 10443. Konkretno, OpenVPN1 nginx ili HAproxy moraju raditi reverse proxying/load balancing na Linux1 i Linux2, pri čemu Linux2 mora biti port-forwardan kroz OpenVPN2 kroz neki proizvoljan port. Vrijedi i obrnuto – OpenVPN2 nginx ili HAproxy moraju raditi reverse proxying/load balancing na Linux 2 i Linux 1, pri čemu Linux1 mora biti port-forwardan kroz OpenVPN1 kroz neki proizvoljni port. Isti scenarij treba biti konfiguriran i na portu 12443, ali se reverse proxying/load balancing treba raditi za HTTPS promet IIS servera koje ćete instalirati na Server3 i Server4. HTTP protokol mora biti isključen na svim web-serverima (Linux1, Linux2, Server3, Server4) – koristimo isključivo HTTPS protokol. Minimalna korištena HTTPS razina je TLS1.2, za sve

web servere. Konfigurirajte FQDN OpenVPN1 mašine `openvpn1.imeprezime.local`, a za OpenVPN2 `openvpn2.imeprezime.local`.

ServerDC – domenski kontroler na Lokaciji A. Mora imati podignut DC, DNS i DHCP za podmrežu unutar Lokacije A. Sve mašine na Lokaciji A moraju koristiti ServerDC kao DNS i NTP server. RODC – domenski kontroler na Lokaciji B. Mora imati podignut DC, DNS i DHCP za podmrežu unutar Lokacije B. Sve mašine na Lokaciji B moraju koristiti RODC kao DNS server. Sve mašine na lokaciji B moraju koristiti ServerDC kao NTP server.

Server1 i Server2 – member serveri na Lokaciji A. Na ovim serverima je potrebno napraviti Parity Storage Space + spare koristeći četiri diska. Na tom storage spaceu mora biti napravljen disk koji koristi deduplikaciju, i na njega formatiran ReFS datotečni sustav. Na Server1 je potrebno instalirati root CA, a na Server2 je potrebno instalirati subordinate CA. Nakon toga, potrebno je napraviti dva certificate template-a. Prvi koji će se zvati „Web poslužitelji“, standardnog trajanja certifikata 2 godine, koji ćete iskoristiti za izdavanje certifikata za IIS servere koje ćete instalirati na Server3 i Server4. Drugi će se zvati „Web klijenti“, standardnog trajanja certifikata 1 godinu, koji će izdavati certifikate za autentifikaciju korisnika na Server3 i Server4. Također, potrebno je podići Windows File Server Cluster koristeći Failover Clustering metodologiju iz Windows Servera. Na tom File Server Clusteru potrebno je podići minimalno jedan clustered CSV disk i na njemu podići SMB share. Diskove potrebne za ovaj zadatak montirajte kao iSCSI LUN-ove sa Linux1 mašine, kako je zadano u zadacima za Linux1 mašinu.

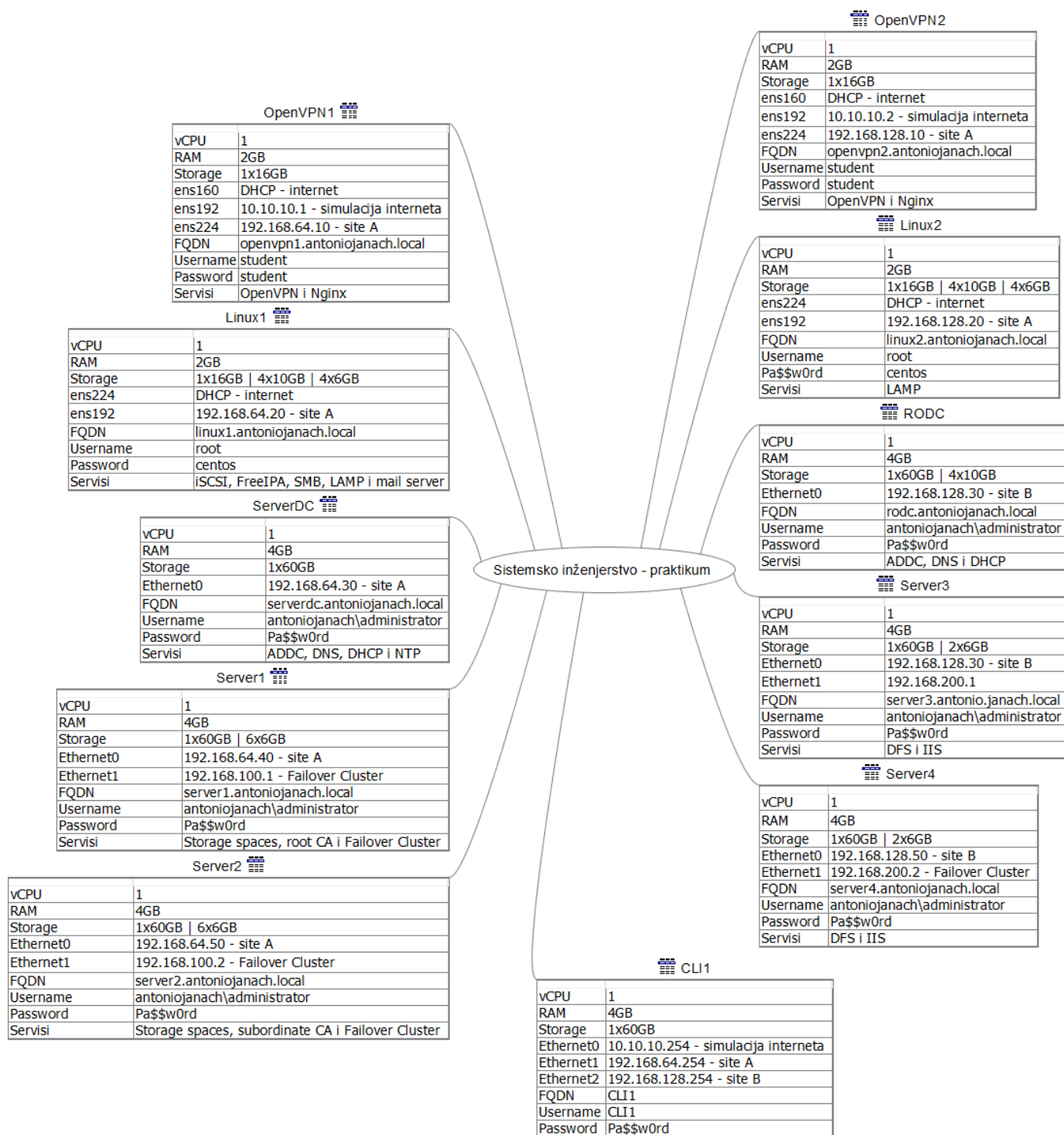
Server3 i Server4 – member serveri na Lokaciji B. Na ovim serverima je potrebno napraviti DFS replikaciju na drugom disku, kojeg ćete montirati kao W disk. Nakon toga, potrebno je instalirati IIS sa svim potrebnim modulima za client certificate authentication. Prevedeno, za ulogiravanje na web stranicu koju ćete napraviti u idućem koraku, na IIS-u na S3/4 potrebno je da se klijent predstavi s certifikatom. Sve ostale autentifikacijske metode moraju biti zabranjene. U IIS-u napravite novu web stranicu imena `ImePrezimeWeb` koja treba posluživati samo jednu tekstualnu datoteku, sadržaja „Ovo je web stranica studenta `Ime Prezime studenta`, na serveru `X`“. Ime Prezime studenta su – očito – ime i prezime studenta, a X je 3 ili 4, ovisno o serveru. Navedenu web stranicu morate bindati na port 443, s certifikatom kojeg će izdati CA 5 infrastruktura na Lokaciji A. HTTP promet mora biti isključen. Web-stranice na portu 443 se trebaju kroz OpenVPN2 port-forwardati kako je opisano u zadatku za OpenVPN1 i OpenVPN2 poslužitelje.

Linux1 mašina ima četiri dodatna diska, koje je potrebno iskoristiti za kreiranje jedne volume grupe imena `iSCSI01`. Unutar te volume grupe potrebno je izrezati tri logička volumena imena `LUN0`, `LUN1` i `LUN2`. Navedene je logičke volumene potrebno kroz iSCSI target prikazati kao LUN broj 0, 1 i 2 prema virtualkama Server1 i Server2 za potrebe podizanja Windows Failover Clusteringa. Napravite potrebno maskiranje u iSCSI konfiguraciji kako bi samo Server1 i Server2 virtualke mogle koristiti ova dva iSCSI LUN-a. Na Linux1 podignite FreeIPA server i syncajte ga s Active Directory-em na ServerDC-u. U direktoriju `/domainshare` napravite SMB share na koji pristup imaju samo korisnici domene `imeprezime.local`. Napravite za taj SMB share backup skriptu koja će raditi inkrementalni backup svaki dan u 02:00:00, i slati taj backup na Linux2, u direktorij `/backup`. Podignite LAMP server i u njega instalirajte zadnju verziju WordPress-a, koji mora biti serviran iz direktorija `/wordpress1`. WordPress mora biti dostupan samo kroz HTTPS/TLS 1.2 konfiguraciju. Podignite na Linux1 i mail server koji će biti MX za domenu `imeprezime.local`, i koji će dopuštati mail relaying samo za Linux strojeve s Lokacije A i Lokacije B.

Linux2 mašina treba posluživati WordPress web-site iz direktorija /wordpress2. U tu svrhu, podignite LAMP server i u njega instalirajte zadnju verziju WordPress-a. WordPress mora biti dostupan samo kroz HTTPS/TLS 1.2 konfiguraciju. Također, Linux2 mašina treba korištenjem TCP protokola slati sve svoje log zapise na Linux1 mašinu. Linux1 mašina ih treba rotirati svaka 2 dana, koristeći kompresiju.

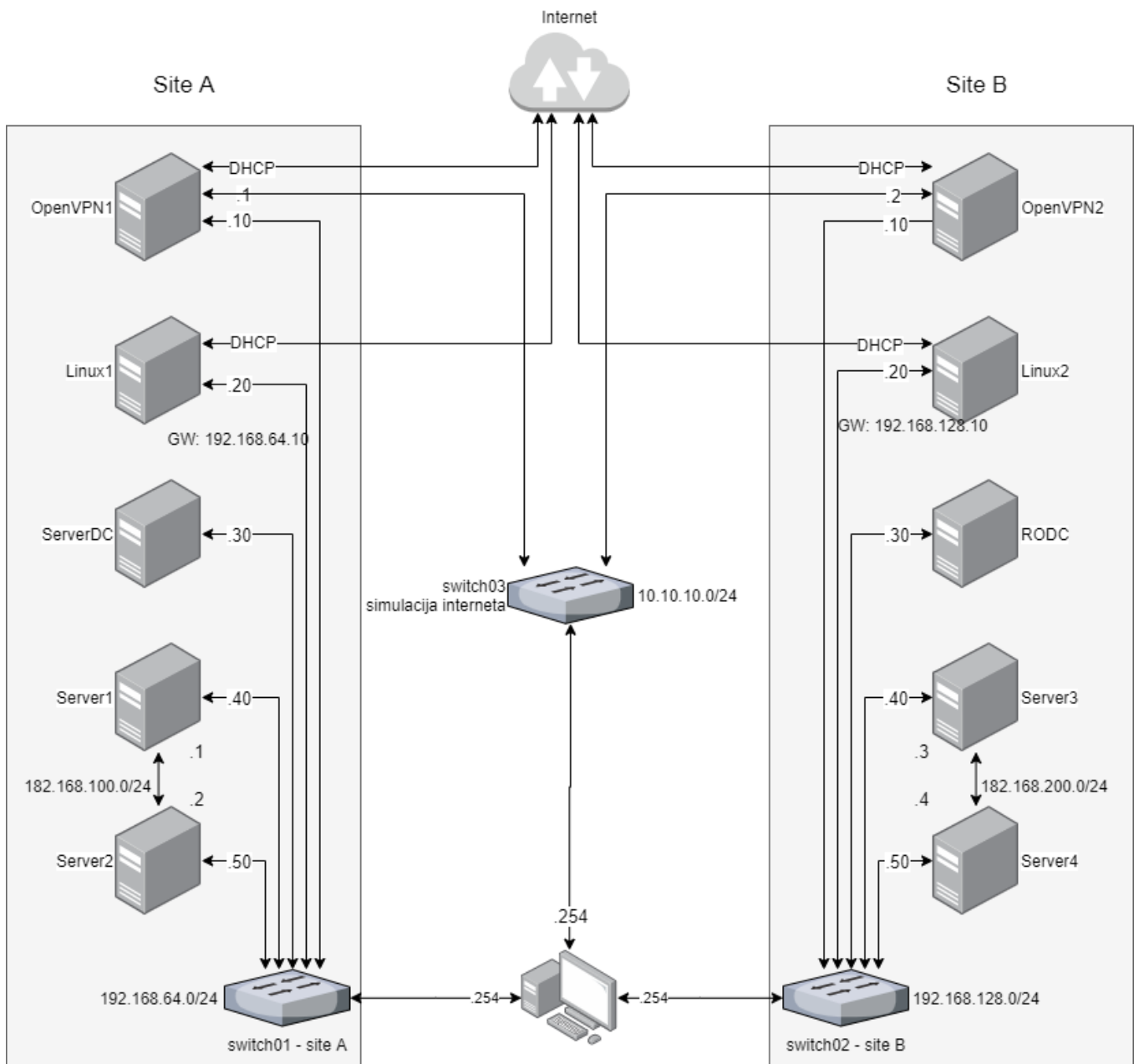
Linux1 i Linux2 mašine moraju imati SELinux u Enforcing modu. Certificate za Linux1 i Linux2 treba izdati na CA infrastrukturi od Lokacije A. Firewall konfiguracija na Linux1 i Linux2 mora biti konfigurirana tako da su samo propušteni servisi koji su potrebni. Konfigurirajte FQDN Linux1 mašine linux1.imeprezime.local, a za Linux2 linux2.imeprezime.local i sukladno tome izdajte certificate. Na sva četiri Linux stroja (OpenVPN1 i 2, Linux1 i 2) napravite pojedinačnu skriptu koja će svaka dva sata provjeriti stanje logova i reportirati sve čudne aktivnosti koje su se dogodile na SSH servisu. Ako skripta primijeti bilo kakve čudne aktivnosti (npr. brute-force napad, pojedinačna neuspješna logiranja i slično), treba poslati e-mail na student@linux1.imeprezime.local sa Subject poljem „Suspicious activity“ i tijelom poruke koje će sadržavati sve po vama sumnjive aktivnosti. Skripte za svoj posao moraju koristiti postfix mail server koji je podešen na linux1.imeprezime.local.

3. Opis infrastrukture



Slika 1: prikaz opisa infrastrukture kroz umnu mapu

4. Topologija infrastrukture



Slika 2: prikaz topologije infrastrukture

5. Priprema infrastrukture

5.1. Mrežna konfiguracija po lokacijama

Za pripremu infrastrukture potrebno je na svakome od računala podesiti IPv4 adrese na mrežnim adapterima. Isto vrijedi za site A i za site B kao i za CLI1 računalo.

OpenVPN Ubuntu poslužitelji imaju tri virtualne mrežne kartice. Prva virtualna mrežna kartica imena ens160 će se koristiti za spajanje na Internet, te se ova virtualna mrežna kartica isključivo koristi za instalaciju potrebnih paketa, kao i za *update* poslužitelja. Druga virtualna mrežna kartica koristi se za *site-to-site* VPN između OpenVPN1 i OpenVPN2. Treća virtualna mrežna kartica koristi se za *intra-site* komunikaciju s ostalim virtualnim računalima (Windows, Linux).

ServerDC i RODC imaju po jednu mrežnu karticu, za *intra-site* komunikaciju s preostalim virtualnim mašinama na svakoj od lokacija.

Server1, Server2, Server3 i Server4 imaju po dvije mrežne kartice. Prva je za *intra-site* komunikaciju. Druga mrežna kartica je za komunikaciju s pripadajućim parom u *failover clusteru*.

Linux1 i Linux2 CentOS poslužitelji imaju po dvije mrežne kartice. Prva je za *intra-site* komunikaciju, a druga je spojena na Internet. Drugu mrežna kartica spojena je na Internet i koristi se isključivo za instalaciju dodatnih paketa koji su potrebni za rješavanje projekta, kao i za *update*.

Cli1 računalo ima tri mrežne kartice. Prva mrežna kartica spojena je na site-to-site mrežu između dvije lokacije i mora moći *ping* naredbom komunicirati sa OpenVPN1 i OpenVPN2. Druga mrežna kartica je u mreži od lokacije A. Treća mrežna kartica je u mreži od lokacije B. Te dvije kartice trebaju biti isključene, osim za potrebne testiranja funkcionalnosti na lokacijama A i B. Cli1 računalo ne smije ni u kojem trenutku biti preko OpenVPN poslužitelja biti spojeno na Internet.

Poslužitelji na lokaciji A kao *default gateway* koriste OpenVPN1. Mašine na lokaciji B kao *default gateway* koriste OpenVPN2.

Mrežna konfiguracija po lokacijama konfigurirana je na način kao što je opisano u poglavlju „3. Opis infrastrukture“. Te za detaljnu topologiju infrastrukture pogledati „4. Topologija infrastrukture“. Nakon uspješne konfiguracije potrebno je *updateati* Linux računala. Ovime završava priprema infrastrukture te će u daljnjem dokumentu biti obrađena tema koja govori o razradi cijelog projekta u detalje.

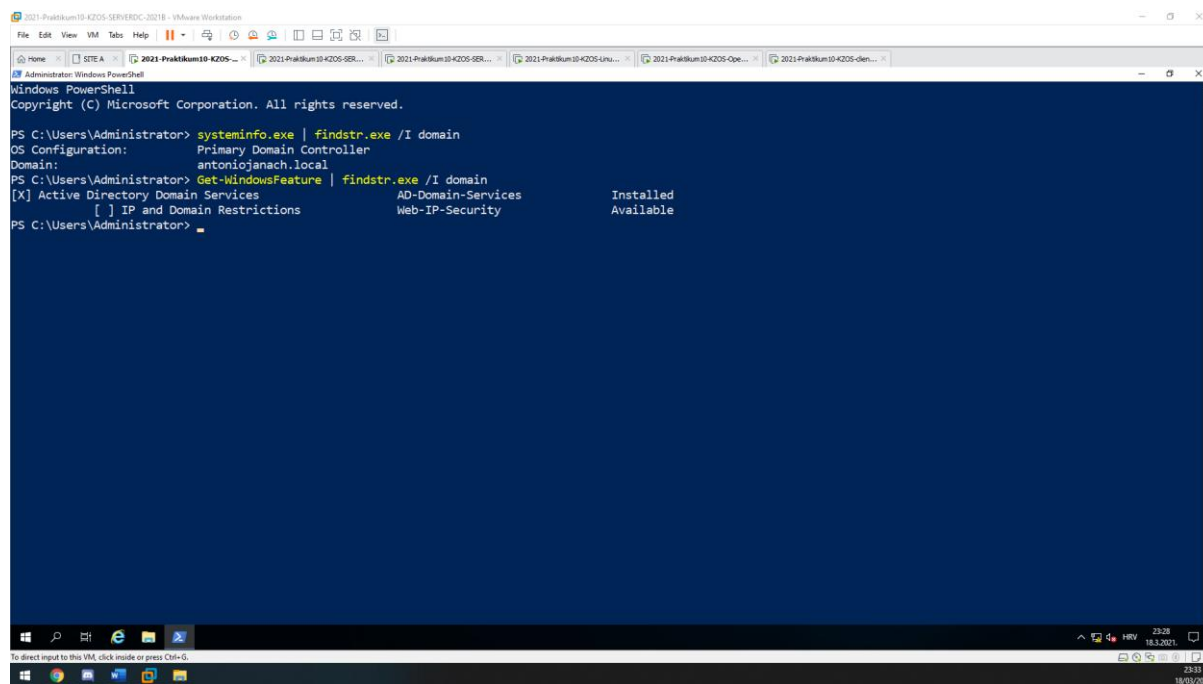
6. Razrada projekta - projektno rješenje

6.1. Podizanje domene na lokaciji A

Active Directory skup je servisa kojim se upravlja identitetima i kontrolom pristupa resursima na mreži. *Active Directory Domain Services* (AD DS) je imenički servis koji omogućuje administratorima izraditi organizacijske jedinice koje se nazivaju domenama.

Kad je instalacije AD DS uloge završena potrebno je propagirati ServerDC u Domain Controller. Kod propagacije Domain Controller-a odabire se opcija dodavanje forest-a i u polje Root domain name upisuje se antoniojanach.local. U Domain Controller opcijama functional level je Windows Server 2016, SERVERDC će ujedno koristiti DNS ulogu. U polje DSRM(Directory Services Restore Mode) upisati lozinku Pa\$šw0rd. U nadolazećim koracima odabiru se default-ne postavke te kad se SERVERDC propagira bit će automatski pokrenuti s funkcionalnom domenom.

Prema zahtjevima koji su navedeni u projektu podignuta je AD DS uloga na ServerDC poslužitelju. Ime domene koja se nalazi u *forest-u* glasi: „antoniojanach.local“.



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

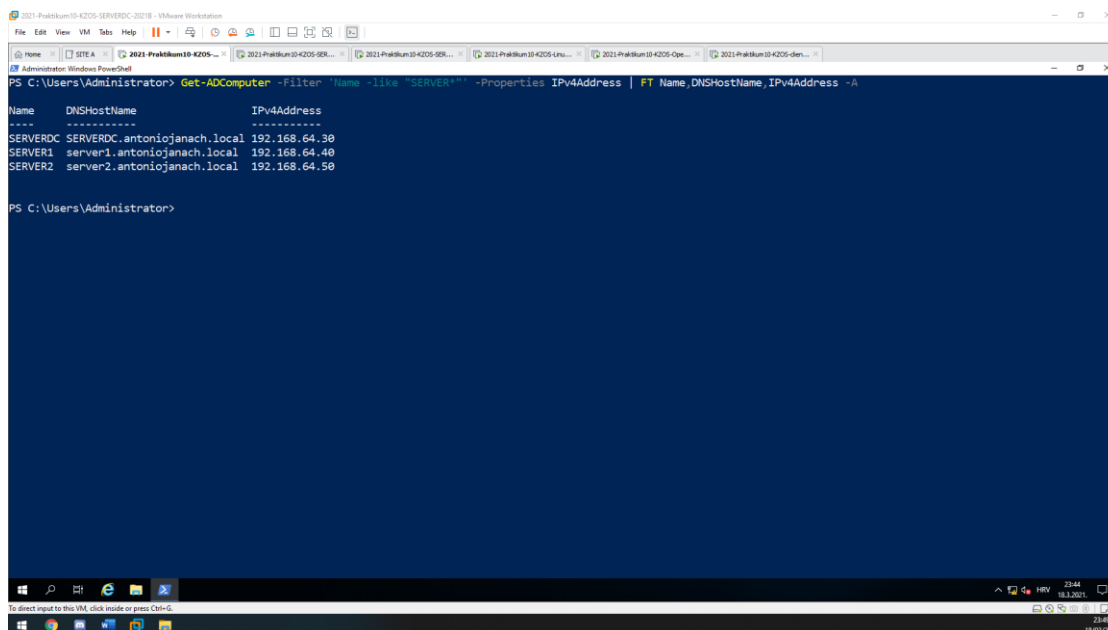
PS C:\Users\Administrator> systeminfo.exe | findstr.exe /I domain
OS Configuration:      Primary Domain Controller
Domain:                antoniojanach.local
PS C:\Users\Administrator> Get-WindowsFeature | findstr.exe /I domain
[X] Active Directory Domain Services      AD-Domain-Services      Installed
[ ] IP and Domain Restrictions           Web-IP-Security          Available
PS C:\Users\Administrator>
```

Slika 3: prikaz instalacije AD DS uloge kao i prikaz kreirane domene imena „antoniojanach.local“

U sljedećem poglavlju bit će obrađena tema gdje se opisuje dodavanje Server1 i Server2 poslužitelja u domenu antoniojanach.local.

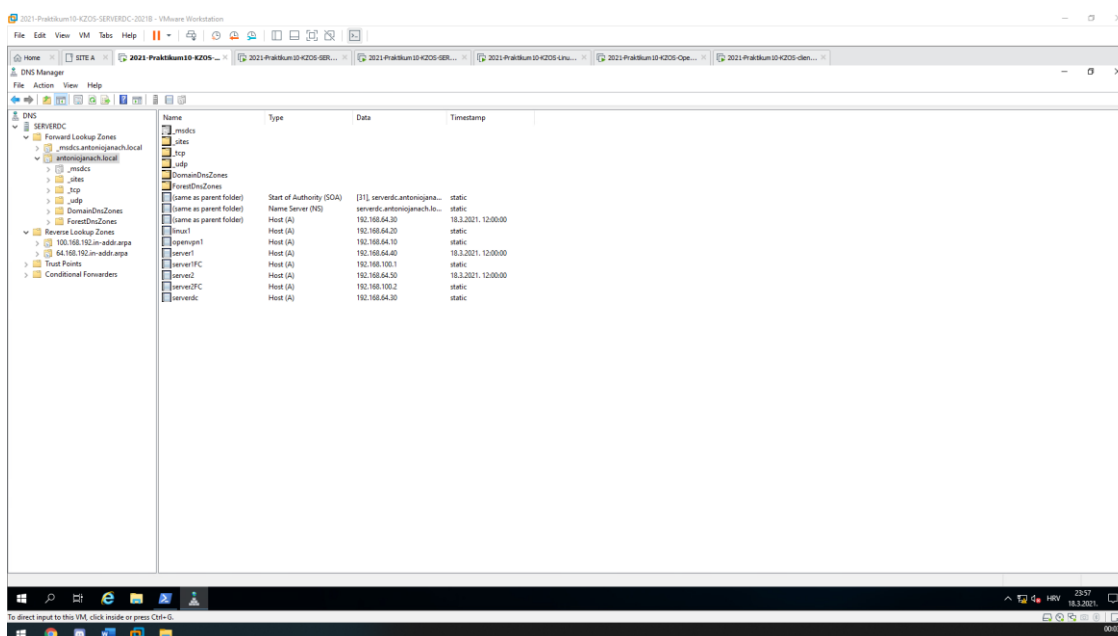
6.2. Dodavanje ostalih Windows računala u domenu na lokaciji A

Funkcionalnom domenom ostala računala sprema su biti dodana u domenu. Metoda za dodavanje poslužitelja u domenu ista je za sve Windows poslužitelje. Na lokaciji A potrebno je dodati Server1 i Server2 u domenu „antoniojanach.local“.



Slika 4: prikaz dodanih poslužitelja Server1 i Server2 u domenu "antoniojanach.local"

Kad su Server1 i Server2 poslužitelji dodani u domenu potrebno je na ServerDC računalu dodati reverznu primarnu zonu za 192.168.64.0/24. Isto tako potrebno je ručno dodati OpenVPN1 i Linux1 poslužitelje u *forward* primarnu zonu „antoniojanach.local“. Ne smijemo zaboraviti dodati mrežne adaptere na Server1 i Server2 poslužiteljima koji se koriste za *Failover Cluster* ulogu.

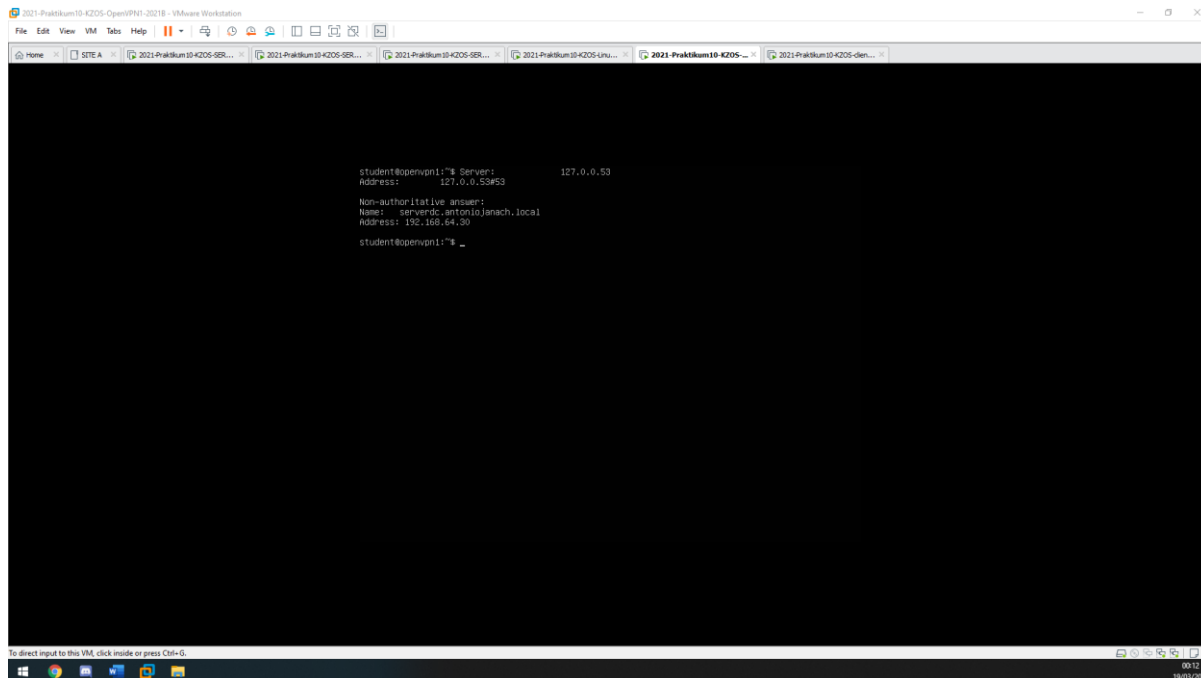


Slika 5: prikaz DNS zapisa i reverznih primarnih zona

Sljedeće poglavlje govori o postavljenju trajnog DNS nameserver-a na Ubuntu poslužitelju.

6.3. Postavljanje trajnog DNS namespace-a na Ubuntu poslužitelj

U projektu je primijećeno da korištenjem nslookup naredbe na Ubuntu poslužitelju nije resolv-an DNS server. Output koji vraća nslookup naredba je „Non-authorative answer“. Rješenje se krije u dodavanju DNS server IP adrese u resolv.conf datoteku.



```
student@openvpn1:~$ nslookup 127.0.0.53
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   serverdc.antoniojanach.local
Address: 192.168.64.30

student@openvpn1:~$ _
```

Slika 6: prikaz problema

Kako bi se otklonio ovaj problem potrebno je instalirati resolvconf servis na Ubuntu poslužitelj. Preuzeti servis potrebno pokrenuti da se pokreće kao i omogućiti pokretanje tog servisa kad se i računalo pokreće u protivnom DNS resolv s Ubuntu poslužitelja neće raditi prema DNS serveru (ServerDC).

```
#instalacija servisa:
Sudo apt-get install resolvconf -y
#pokretanje servis i omogućiti da se servis pali s pokretanjem računala:
Sudo systemctl start resolvconf.service
Sudo systemctl enable resolvconf.service
#dodavanje nameserver IP adrese od ServerDC domenskog poslužitelja u datoteku:
Echo -e „nameserver 192.168.64.30“ >> /etc/resolvconf/resolv.conf.d/head
#trajni zapis DNS servera u resolv.conf datoteku koristeći sljedeće naredbe:
Sudo resolvconf --enable-updates
Sudo resolvconf -u
```

Ovim poglavljem završava postavljanje trajnog DNS namespace-a na Ubuntu poslužitelj.

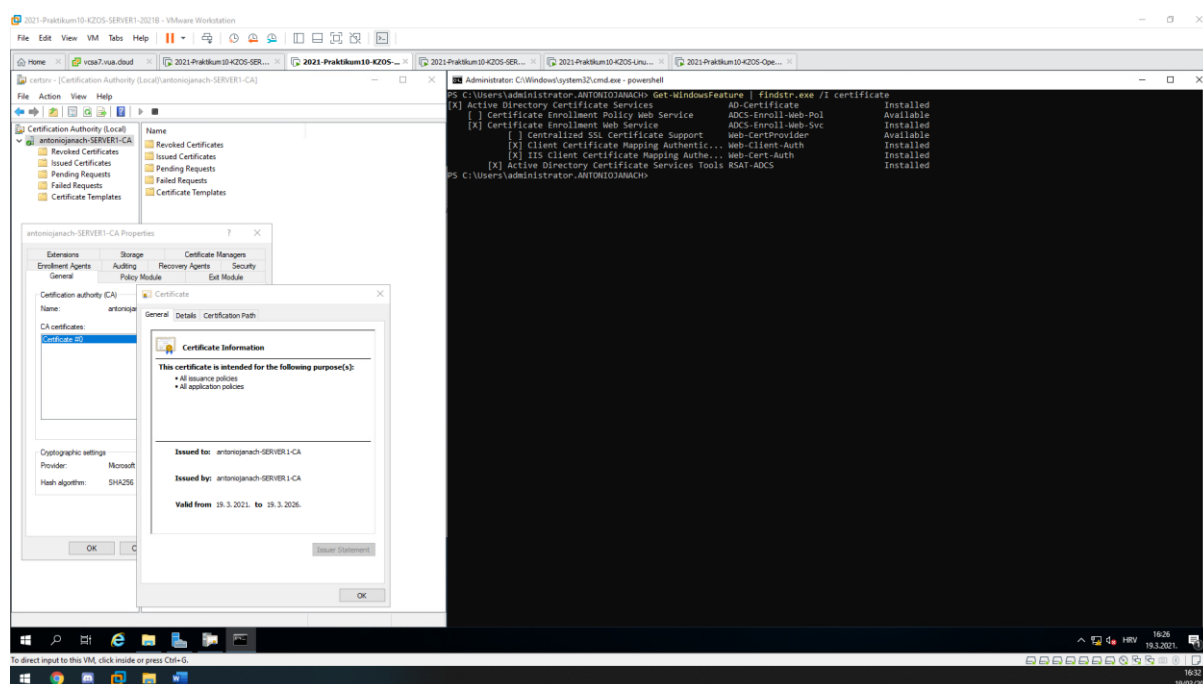
6.4. Instalacija root CA i subordinate CA – Server1 i Server2

Tema ovoga poglavlja obrađuje temu Active Directory certifikacijski servisi (Certificate services) – AD CS. Oni su skup uloga koje se instaliraju na Windows poslužitelj, a implementacija PKI (Public Key Infrastructure) infrastrukture. PKI je skup mehanizama za potvrdu identiteta svakog člana koji sudjeluje u kriptiranoj mrežnoj komunikaciji. PKI infrastruktura može biti vrlo jednostavna (s jednim samostalnim poslužiteljem) ili vrlo složena, s mnogo hijerarhijski poslužitelja integriranih u Active Directory.

Opis infrastrukture koja se želi postići, a to je da je potrebno na Server1 instalirati Root CA, a na Server2 je potrebno instalirati subordinate CA.

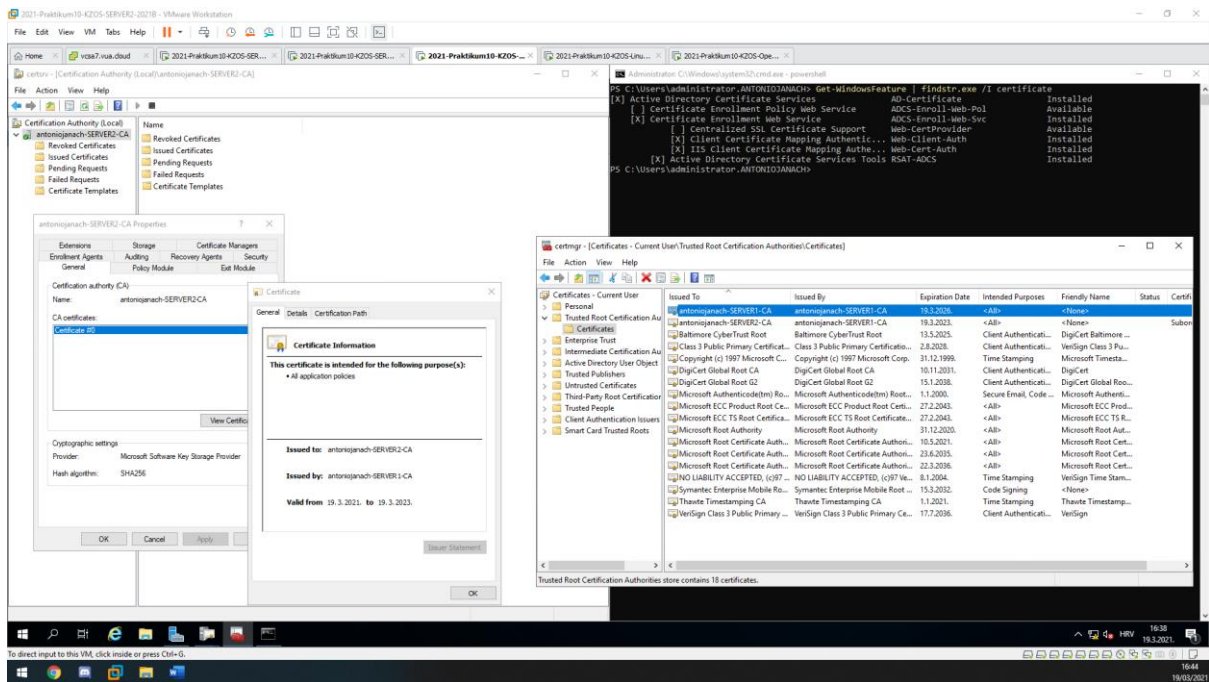
Server1: ima instaliranu root CA ulogu, i root CA je najviša razina hijerarhije i služi kao sidro povjerenja. Da bi se moglo vjerovati certifikatu krajnjeg entiteta, korijenski CA do kojeg se lanac povjerenja ugrađuje u operacijski sustav, a to je preglednik, uređaj ili bilo što drugo što potvrđuje certifikat.

Server2: ima instaliranu ulogu subordinate CA, svrha subordinate CA definirati i odobriti vrste certifikata koji se mogu zatražiti od root CA.



Slika 7: prikaz instalacije AD CS uloge kao root CA na Server1

Kao što se vidi na „Slika 7“ Server1 je root CA zato što je sam sebi potpisao certifikat. Sad kad je dovršena konfiguracija root CA poslužitelja, Server2 je spreman za konfiguraciju subordinate CA uloge. Bitno je napomenuti da oba računala prije postavljanja ove konfiguracije moraju biti u domeni „antoniojanach.local“.



Slika 8: prikaz instalacije AD CS uloga kao subordinate CA na Server2

Kao što se vidi na „Slika 8“ Server2 je subordinate CA zato što je root CA iz generirane request (.req) datoteke potpisao certifikat. Koji je instaliran na Server2 poslužitelju.

6.4.1. Kreiranje certifikata za Web poslužitelj

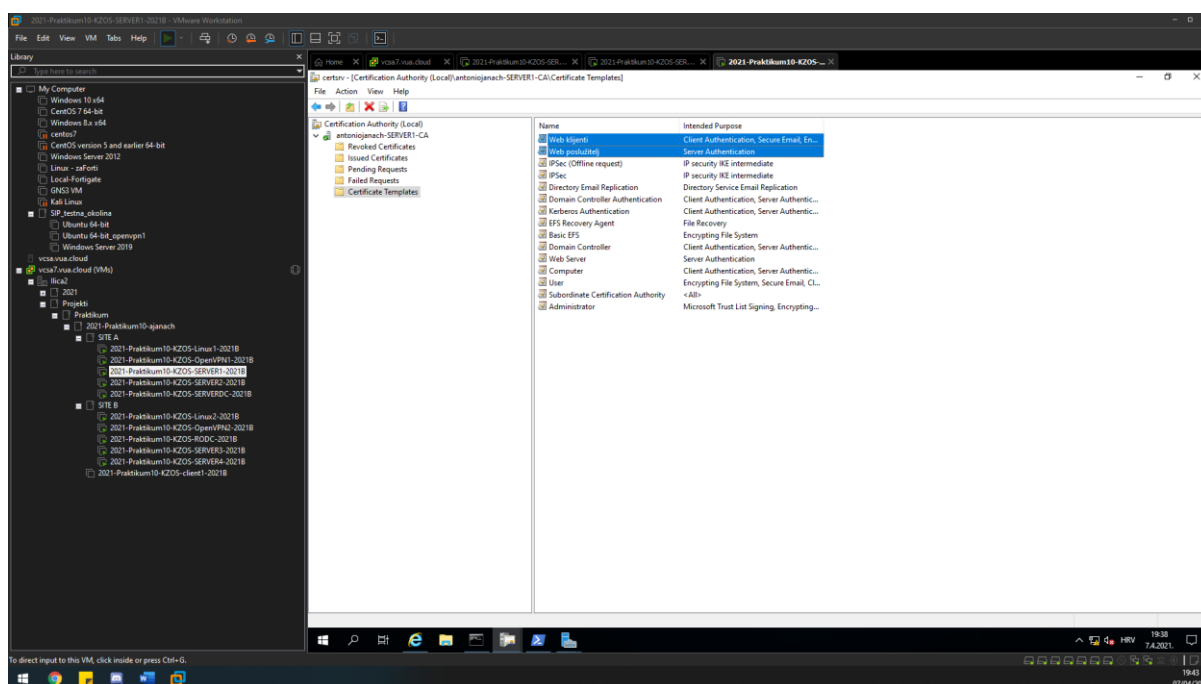
Kao što je u zahtjevu navedeno, nakon instalacije „root CA“ i „subordinate CA“ potrebno je kreirati *certificate template* koji će se zvati „Web poslužitelji“ standardnog trajanja od dvije godine. Taj certifikat koristit će se za izdavanje certifikata za IIS servere koji su će biti instalirani na Server3 i Server4.

Kod kreiranja *certificate template-a* potrebno je iz postojećih certifikacijskih predložaka odabrati „Web server“ predložak, zatim ga duplicirati. U kartici „General“ potrebno je dati ime predlošku (Web poslužitelj) i odabrati opciju „Publish certificate in Active Directory“. Sljedeće što je potrebno, a to je u kartici „Security“ dodati Server3 i Server4 kao „Computer“ objekt i odabrati opciju u kartici dozvola za „Enroll“ i „Autoenroll“. Kad su ove postavke primijenjene taj predložak je potrebno izdati koristeći opciju „New certificate template to issue“.

6.4.2. Kreiranje certifikata za autentifikaciju korisnika

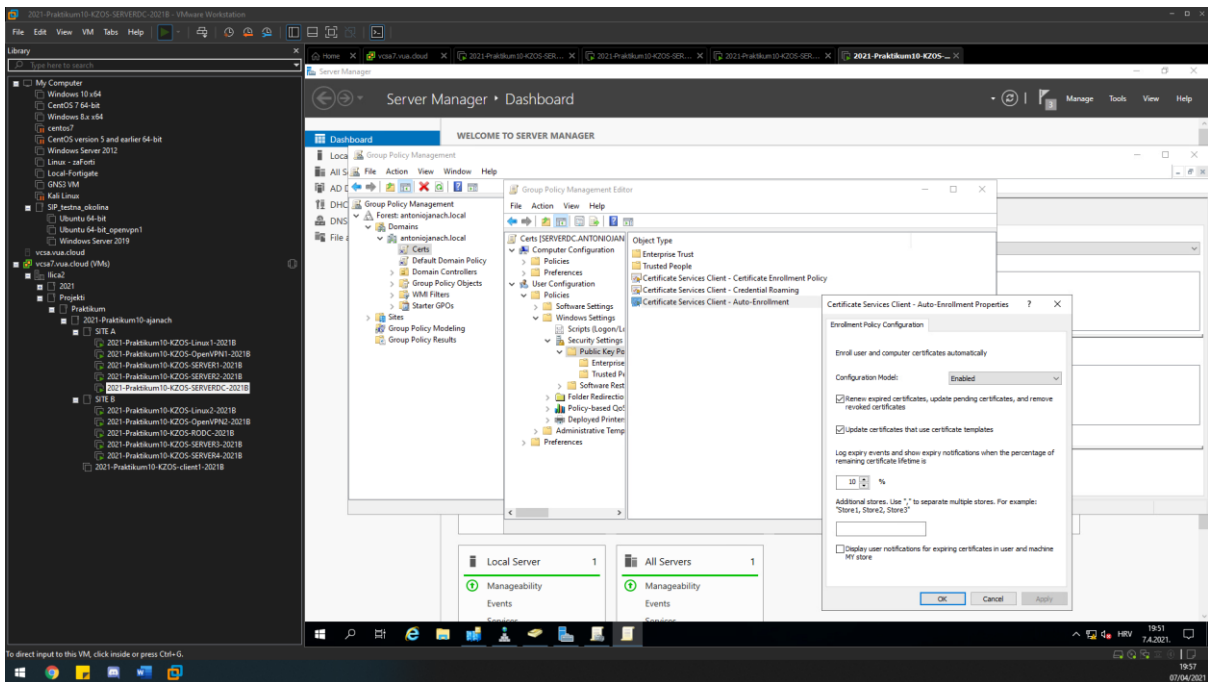
Nakon kreiranog predloška imena „Web poslužitelji“ potrebno je kreirati još jedan predložak koji će izdavati certifikate za autentifikaciju korisnika na Server3 i Server4. Certifikacijski template potrebno je nazvati „Web klijenti“ i standardnog trajanja od jedne godine.

Kod kreiranja *certificate template-a* potrebno je iz postojećih certifikacijskih predložaka odabrati „User“ predložak, zatim ga duplicirati. U kartici „General“ potrebno je dati ime predlošku (Web klijenti) i odabrati opciju „Publish certificate in Active Directory“. Zatim u kartici „Security“ pod pravima od „Domain Users“ nužno je dodati pravo na „Autoenroll“. Kad su ove postavke primijenjene taj predložak je potrebno izdati koristeći opciju „New certificate template to issue“.



Slika 9: prikaz kreiranog certifikata za Web poslužitelj i autentifikaciju korisnika

Kreiranje certifikacijski predložak potrebno je izdati u AD tako što se je potrebno pozicionirati u „Group Policy Management“ na SERVERDC poslužitelju. Potrebno je kreirati novi GPO imena „Certs“ i urediti stavku „Certificate Services Client - Auto-Enrollment“.



Slika 10: prikaz postavka "Certificate Services Client - Auto Enrollment Properties"

Nakon što je certifikacijski predložak izdan u AD potrebno je pokrenuti sljedeću komandu (gpupdate /force) koristeći CMD na svim Windows poslužiteljima.

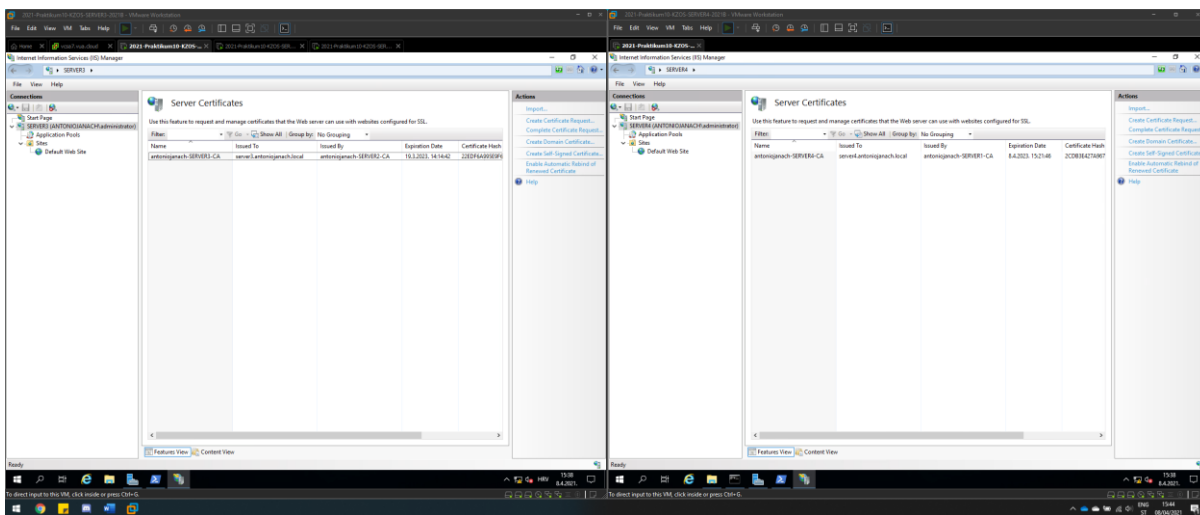
Ovime su certifikacijski predlošci uspješno konfigurirani i propagirani na sve poslužitelje.

6.5. IIS konfiguracija na SERVER3 i SERVER4

Internet Information Services ili skraćeno IIS je *Microsoft-ov* Web poslužitelj koji je jedna od uloga Windows servera. IIS ima mogućnost autentifikacije od kojih je najvažnije autentifikacija pomoću certifikata i integrirana Windows autentifikacija zbog koje se korisnici ne moraju ručno prijavljivati da bi pristupili Web stranici. Već Windows autentifikacija omogućuje klijentskim računalima da pošalju IIS poslužitelju informacije o korisniku. Integrirana Web-autentifikacija se često koristi kako bi se pristup omogućio zaposlenicima koji su za to ovlašteni.

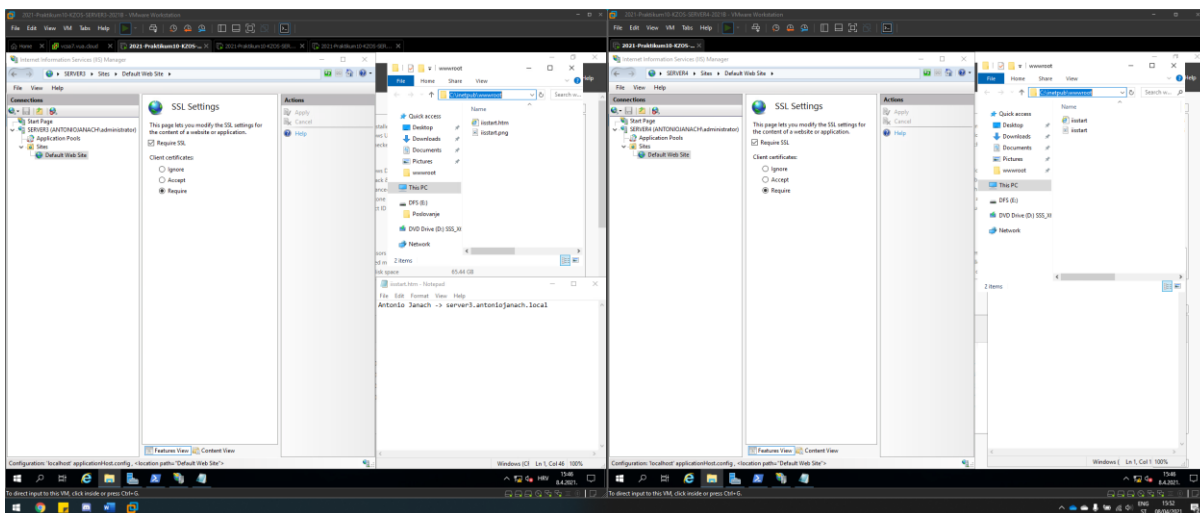
Za početak na SERVER3 i SERVER4 potrebno je instalirati IIS ulogu koristeći „Server Manager“. Zatim je u kartici „Role Services“ nužno odabrati „Client Certificate Mapping Authentication“ i „IIS Client Certificate Mapping Authentication“.

Kad su IIS uloge instalirane na SERVER3 i SERVER4 poslužitelj, potrebno je kreirati domenske certifikate za Web poslužitelj pomoću SERVER2 poslužitelja koji ima ulogu „Subordinate CA“ kako bi se postigao „chain of trust“.



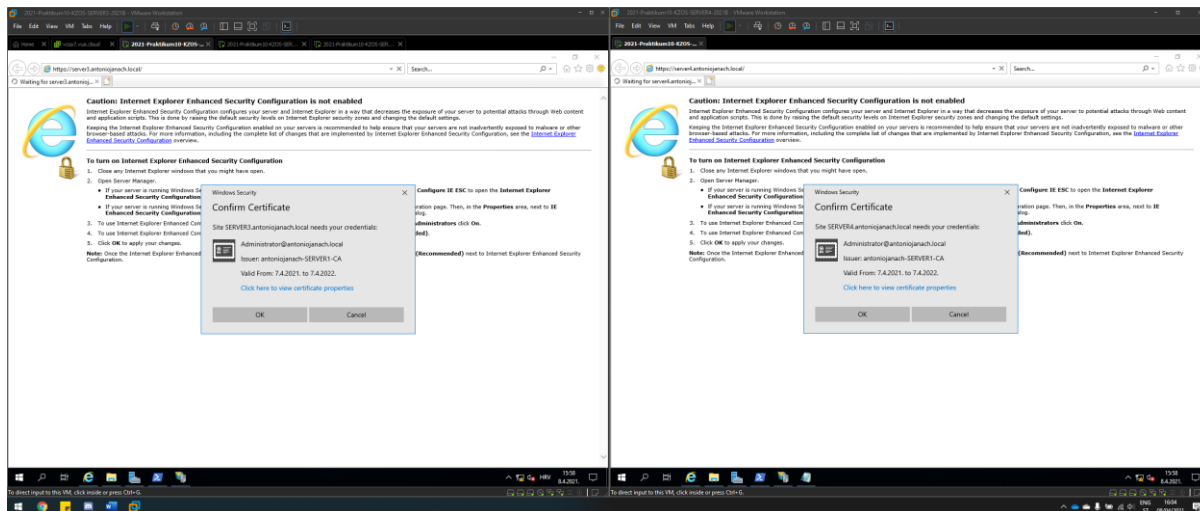
Slika 11: prikaz kreiranog domenskog certifikata na SERVER3 i SERVER4 poslužitelju

Na SERVER3 poslužitelju certifikat je izdan od SERVER1 poslužitelja koji ima ulogu „root CA“, a na SERVER4 poslužitelju certifikat je izdan od SERVER2 poslužitelja koji ima ulogu „subordinate CA“.



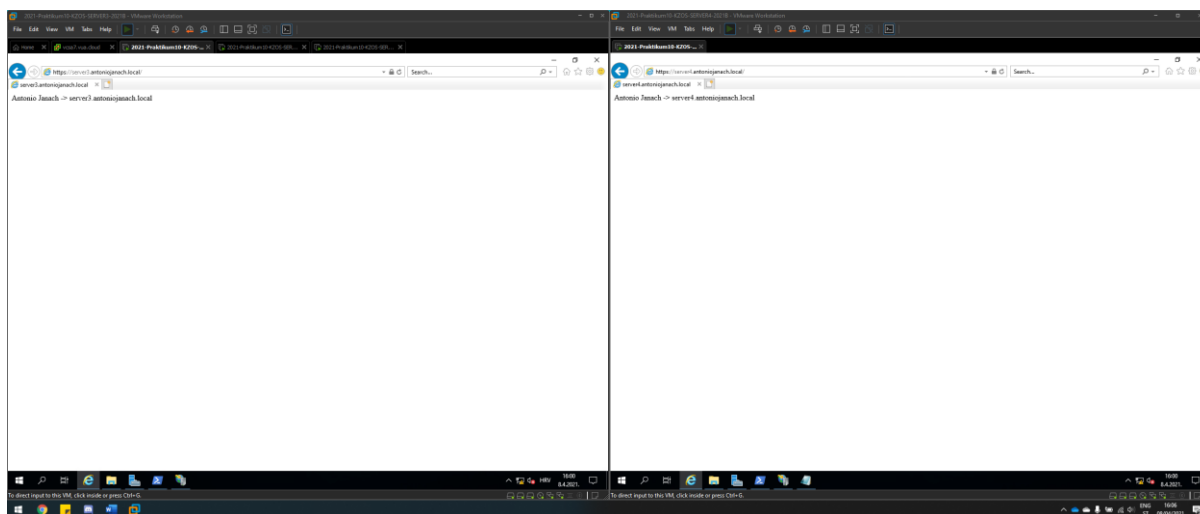
Slika 12: prikaz SSL postavki i promjene sadržaja zadane IIS Web stranice na oba poslužitelja

Kao što je prikazano na slici iznad SSL postavke podešenu su tako da je klijentski certifikat nužan za pristup Web stranici. Na ovaj način svaki puta kad će netko iz domene pristupiti Web stranici, prije prikaza sadržaja pojavit će se prozor „Confirm Certificate“, što znači da je uspješno konfigurirana autentifikacija pomoću certifikata. Također za zadanu Web stranicu potrebno je podesiti postavke „Site Bindings“ te dodati SSL certifikat (postavke je potrebno primijeniti na oba poslužitelja).



Slika 13: prikaz prozora autentifikacije korisnika pomoću certifikata

Kad se prozor za autentifikaciju korisnika za pristupanje Web sadržaju prikaže na ekranu, potrebno je pritisnuti „OK“, i ako se korisnik nalazi u domeni on će moći pristupiti Web sadržaju.



Slika 14: pristupanje stranici koristeći SSL certifikat

IIS konfiguracija na SERVER3 i SERVER4 poslužitelju uspješno je konfigurirana tako da koristi SSL certifikat za pristup pomoću „https“ protokola i koristeći autentifikaciju korisnika pomoću certifikata za pristup Web sadržaju. Također Web sadržaju je moguće pristupiti s bilo kojeg poslužitelja/računala koji je domenski korisnik, što bi značilo da SERVER1, SERVER2, SERVERDC, SERVER3, SERVER4 mogu pristupiti Web sadržaju.

6.6. Konfiguracija NTP server na ServerDC

Kao mrežni administrator, sustavi ili pomoćno osoblje od vitalnog je značaja da se sati poslužitelja i klijentskog računala i ostalih uređaja sinkroniziraju jer neke pogreške u vremenu mogu značajno utjecati na administrativne zadatke.

Iako se možda ne čini istinitim, ako vrijeme poslužitelja nije sinkronizirano s uređajima. Označava se pogrešno vrijeme za zadatke kontrole pristupa ili nadzor koji nije točan. Cjelokupni postupak rasporeda u sustavu Windows Server 2016 upravlja se zahvaljujući usluzi koja se zove NTP.

NTP (Network Time Protocol) jedan je od najstarijih protokola koji rade na uređajima. NTP protokol koristi UDP port 123 prema zadanim postavkama.

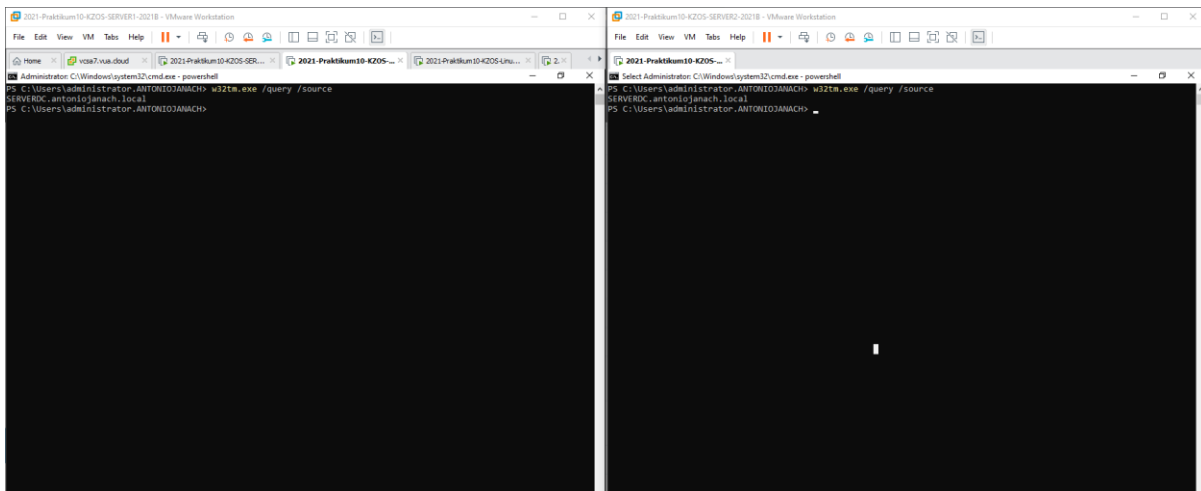
Kao što je u zadatku navedeno ServerDC mora imati ulogu omogućenu NTP server ulogu. Kako bi se ova uloga omogućila na ServerDC uspješno konfigurirala potrebno je pokrenuti sljedeće naredbe na ServerDC poslužitelju koristeći PowerShell.

```
Set-ItemProperty -Path  
"HKLM:\SYSTEM\CurrentControlSet\Services\w32time\TimeProviders\NtpServer" -Name "Enabled" -  
Value 1  
Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\services\W32Time\Config" -Name  
"AnnounceFlags" -Value 5  
Restart-Service w32Time
```

Kad se ove naredbe uspješno pokrenu na ServerDC nužno je propustiti port 123 koji NTP koristi. Kako bi se propustio port 123 potrebno je pokrenuti sljedeće dvije koristeći PowerShell.

```
netsh advfirewall firewall add rule name="NTP TCP port 123" dir=in action=allow  
protocol=tcp localport=123  
netsh advfirewall firewall add rule name="NTP TCP port 123" dir=out action=allow  
protocol=tcp localport=123
```

Svi ostali Windows serveri koji su u domeni bi trebali pokupiti konfiguraciju gdje je NTP server „serverdc.antoniojanach.local“. Za provjeru potrebno je upisati naredbu W32tm /query /source gdje bi izlaz naredbe trebao biti “serverdc.antoniojanach.local”



Slika 15: prikaz izlance naredbe koja prikazuje da je ServerDC NTP server

Sljedeće što je potrebno, a to je da je na Linux1 i OpenVPN1 poslužitelje potrebno postaviti da koriste NTP server koji je podešen na ServerDC poslužitelju.

Konfiguracija na Linux1 poslužitelju:

```
#instalacije chrony servisa:
Dnf install chrony -y
#konfiguracija chrony.conf datoteke:
Echo „pool serverdc.antoniojanach.local iburst“ > /etc/chrony.conf
#pokretanje servisa i omogućavanje servisa kod pokretanja sustava:
Systemctl enable -now chronyd
Systemctl start -now chronyd
#verificiranje statusa:
Chronyc sources
```

Konfiguracija na OpenVPN1 poslužitelju:

```
#instalacija chrony servisa:
Sudo apt-get install chrony -y
#konfiguracija chrony.conf datoteke:
Sudo echo „pool serverdc.antoniojanach.local iburst“ > /etc/chrony/chrony.conf
#pokretanje servisa i omogućavanje paljenje servisa kod pokretanja sustava:
Sudo Systemctl start chrony
Sudo systemctl enable chrony
#verificiranje sustava:
Chronyc sources
```

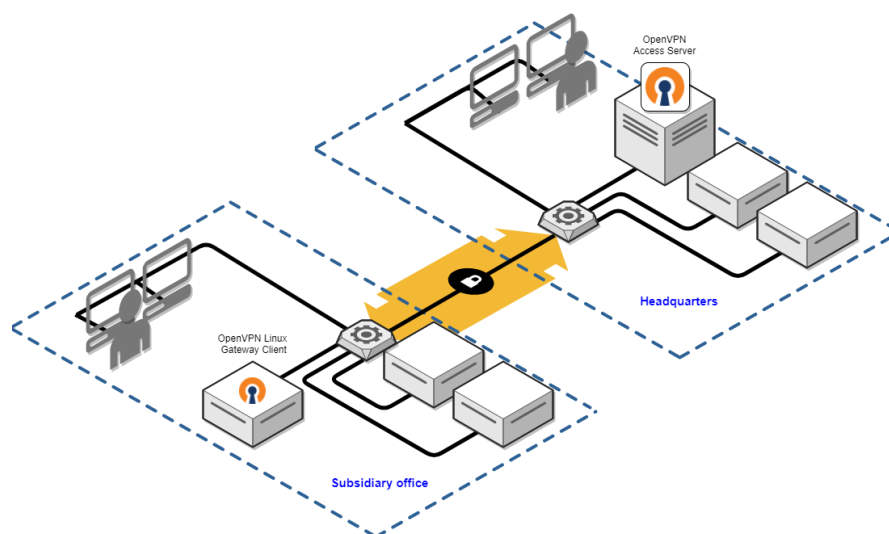
6.7. VPN konfiguracija između Ubuntu poslužitelja

Virtualne privatne mreže mogu se podijeliti na *Site-to-site* VPN i *Remote-access* VPN, a glavna razlika je da u *Site-to-site* VPN-u korisnik nije svjestan postojanja VPN-a, dok je kod *Remote-access* VPN-a svjestan budući da mora upisati svoje korisničko ime i lozinku kako bi se povezao s drugom mrežom. Nadalje, VPN rješenja za udaljeni pristup povezuju udaljene korisnike ili neke manje urede s lokalnom mrežom poduzeća, a povezivanje se obavlja preko modemske veze koristeći Internet, no našem slučaju je to da koristimo povezivanje VPN-om koristeći direktan link između dvaju računala.

Mehanizmi za sigurnost su autentifikacija i enkripcija. Autentifikacija osigurava ograničene provjere pristupa, odnosno dokazuje identitet između korisnika koji se nalazi na krajevima tunela. Koristeći autentifikaciju poslužitelju u zaštićenoj mreži proslijeđuju se potrebni podaci kada klijent želi uspostaviti komunikaciju. Enkripcija služi za očuvanje povjerljivosti i integritet podataka, a označava postupak kodiranja podataka tako da ih mogu pročitati samo oni korisnici koji imaju potrebnu lozinku, dakle oni kojima su podaci namijenjeni.

Kako je zadano u zadatku VPN za konfiguraciju VPN-a koristit će se servis OpenVPN. U ovome slučaju OpenVPN server je "openvpn1.antoniojanach.local", a "openvpn2.antoniojanach.local" je klijent.

Potrebno je izdati certificate pomoću easy-rsa te napraviti konfiguracijski file server.conf u kojem se navode svi izdani certifikati s definiranim log datotekama i postavkama. Na klijentskoj strani potrebno je kreirati konfiguracijsku datoteku imena client.ovpn te navesti sve certifikate i ostale postavke za spajanje na OOS2 računalo.



Slika 16: ilustracijski prikaz site-to-site VPN-a

6.7.1. OpenVPN konfiguracija na server strani

Prije pokretanja skripte potrebno je pokrenuti komandu sljedeću komandu kako bi postali root korisnik, naime na Ubuntu je po zadano isključen root korisnik i nije se moguće prijaviti/ulogirati kao root zato se kao zamjena koristi sudo naredba.

```
Sudo -i
```

Prije instalacije openvpn i easy-rsa paketa potrebno je napraviti ažuriranje svih paketa poslužitelja:

```
apt-get update -y  
apt-get install openvpn easy-rsa -y
```

Da bi se na OpenVPN poslužitelju izradio PKI direktorij, mora se popuniti datoteka koja se zove „vars“ s nekim zadanim vrijednostima.

```
echo -e "set_var EASYRSA_REQ_COUNTRY "HR"  
set_var EASYRSA_REQ_PROVINENCE "Zagreb"  
set_var EASYRSA_REQ_CITY "Zagreb"  
set_var EASYRSA_REQ_ORG "antoniojanach.local"  
set_var EASYRSA_REQ_EMAIL "Janach.antonio@gmail.com"  
set_var EASYRSA_REQ_OU ""  
set_var EASYRSA_ALGO "ec"  
set_var EASY_DIGEST "sha256"" > /home/student/easy-rsa/vars
```

Nakon popunjavanja vars datoteke, može se nastaviti s izradom PKI direktorija. Kako bi to bilo uspješno izvršeno potrebno se je pozicionirati u „easy-rsa“ direktorij i pokrenuti skriptu s init-pki opcijom.

```
cd /home/student/easy-rsa  
./easysrsa init-pki
```

Sada je potrebno stvoriti osnovni par javnih i privatnih ključeva za izdavanje ostalih certifikata te je potrebno ponovno pokrenuti skriptu easysrsa s opcijom build-ca. Ovom skriptom izdani su ca.crt i ca.key ključevi. ca.crt je javna datoteka certifikata CA-a. ca.key je privatni ključ koji CA koristi za potpisivanje certifikata za poslužitelje i klijente.

```
./easysrsa build-ca
```

Kad su kreirani osnovni par ključeva ca.key i ca.crt sada je potrebno sa skriptom easysrsa potrebno generirati privatni serverski ključ server.key i zahtjev za potpisivanje certifikata „server.req“. Sljedećom naredbom potpisujemo zahtjev za izdavanje certifikata te se kreira datoteka naziva server.crt.

```
./easysrsa gen-req server server  
./easysrsa sign-req server server
```

Kao što je u predhodnom paragrafu opisano, za klijentsku stranu potrebno je generirati .key i .crt.

```
./easysrsa gen-req client client  
./easysrsa gen-req client client
```

Sljedeće što je potrebno za funkcionalan VPN je konfiguriranje OpenVPN kriptografskog materijala, za dodatni nivo sigurnosti dodan je zajednički tajni ključ koji će poslužitelj i svi klijenti koristiti s OpenVPN-ovom *tls-crypt* direktivom. Ova opcija se koristi za prikrivanje TSL certifikata koji se koristi kada se poslužitelj i klijent u početku međusobno povežu.

```
Openvpn --genkey --secret /etc/openvpn/server/ta.key
```

Kad su generirani svi potrebni certifikati i ključevi uz kriptografski ključ za serversku stranu potrebno ih je premjestiti na lokaciju `/etc/openvpn/server`.

```
cp /home/student/easy-rsa/pki/ca.crt /etc/openvpn/server/  
cp /home/student/easy-rsa/pki/server.crt /etc/openvpn/server/  
cp /home/student/easy-rsa/pki/private/server.key /etc/openvpn/server/
```

Sad `openvpn1` ima sve potrebne certifikate i ključeve. Sljedeće što je potrebno, a to je konfiguracija OpenVPN-a koristeći konfiguracijsku datoteku. Prvo se je potrebno pozicionirati na putanju `/etc/openvpn/server` i kreirati `server.conf` datoteku u koju je potrebno upisati sljedeće:

```
Cd /etc/openvpn/server  
Touch server.conf  
Echo -e „port 1194  
Proto udp  
Dev tun  
Ca /etc/openvpn/server/ca.crt  
Cert /etc/openvpn/server/server.crt  
Key /etc/openvpn/server/server.key  
Dh none  
Server 10.8.0.0 255.255.255.0  
Ifconfig-pool-persist /var/log/openvpn/ipp.txt  
Push „route 192.168.64.0 255.255.255.0“  
Push „route 192.168.128.0 255.255.255.0“  
Client-config-dir /etc/openvpn/ccd #mkdir /etc/openvpn/ccd, touch openvpn2.antoniojanach.local  
#echo -e „route 192.168.128.0 255.255.255.0“ > openvpn2.antoniojanach.local  
Route 192.168.128.0 255.255.255.0  
Push „redirect-gateway def1 bypass-dhcp“  
Push „dhcp-option DNS 192.168.64.30“  
Push „dhcp-option DNS 192.168.128.30“  
Keepalive 10 120  
Tls-crypt /etc/openvpn/server/ta.key  
Cipher AES-256-GCM  
Auth SHA256  
User nobody  
Group nobody  
Persist-key  
Persist-tun  
Status /var/log/openvpn/openvpn-status.log  
Verb 3  
Explicit-exit-notify 1“ > /etc/openvpn/server/server.conf
```


Prilagođavanje mrežne konfiguracije OpenVPN poslužitelja, jer postoje neki aspekti mrežne konfiguracije koje treba prilagoditi kako bi OpenVPN mogao ispravno usmjeravati promet kroz VPN. Prvi aspekt je prosljeđivanje IP-a, metoda za određivanje kamo treba usmjeravati IP promet. Da bi se podesila zadana postavka prosljeđivanja IP-a OpenVPN poslužitelja potrebno je otvoriti `/etc/sysctl.conf` i upisati sljedeće:

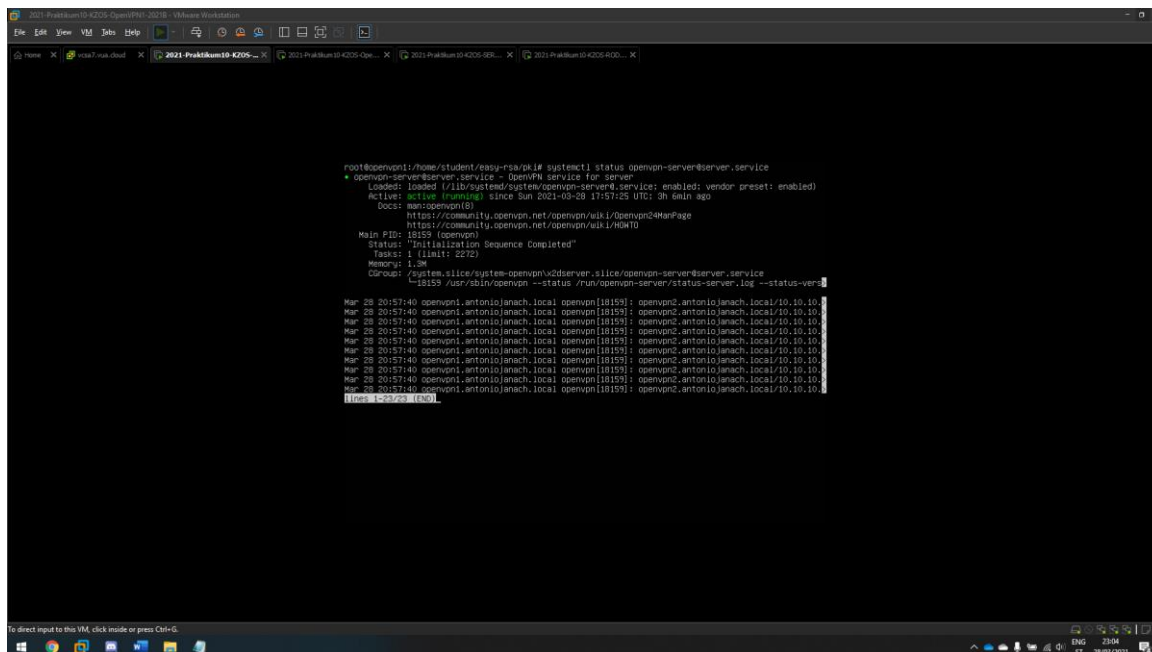
```
Echo -e „net.ipv4.ip_forward = 1“ >> /etc/sysctl.conf
```

Prilagođavanje konfiguracije vatrozida, do sada je instaliran OpenVPN, konfiguriran s generiranim ključevima i certifikatima potrebnim za pristup klijentskom VPN-u. Međutim još nisu dane upute o tome kamo klijent treba poslati dolazni promet. Propustiti je potrebno: OpenVPN servis, uključiti masquerade i propustiti komunikaciju preko `ens192` mrežnog adaptera IP adrese `10.10.10.0/24`.

```
firewall-cmd --zone=public --add-service=openvpn --permanent
firewall-cmd --add-masquerade --permanent
firewall-cmd --permanent --direct --passthrough ipv4 -t nat -A POSTROUTING -s 10.8.0.0/24 -o ens224 -j MASQUERADE
firewall-cmd --permanent --direct --passthrough ipv4 -t nat -A POSTROUTING -s 10.10.10.0/24 -o ens192 -j MASQUERADE
firewall-cmd --reload
```

Sad su svi preduvjeti spremi na pokretanje OpenVPN servisa, OpenVPN radi kao `systemd` usluga. OpenVPN će biti konfiguriran za pokretanje prilikom pokretanja računala, tako da se može povezati s bilo kojim klijentom u bilo kojem trenutku sve dok servis radi.

```
Systemctl -f enable openvpn-server@server.service
Systemctl start openvpn-server@server.service
```



Slika 17: openvpn servis je uspješno pokrenut

Ovime završava konfiguracija OpenVPN-a na serverskoj strani. Sada je serverska strana OpenVPN-a spremna na za spajanje s klijentima.

6.7.2. OpenVPN konfiguracija na klijentskoj strani

Prije pokretanja skripte potrebno je pokrenuti komandu sljedeću komandu kako bi postali root korisnik, naime na Ubuntu je po zadano me isključen root korisnik i nije se moguće prijaviti/ulogirati kao root zato se kao zamjena koristi sudo naredba.

```
Sudo -i
```

Prije instalacije OpenVPN paketa potrebno je napraviti ažuriranje poslužitelja.

```
apt-get update -y
apt-get install openvpn -y
```

Kad su generirani svi potrebni certifikati i ključevi uz kriptografski ključ za klijentsku stranu potrebno ih je premjestiti na lokaciju /etc/openvpn/client sa openvpn1 poslužitelja na openvpn2 poslužitelj tako da je potrebno ove naredbe pokrenuti na openvpn1 serverskoj strani OpenVPN-a:

```
Scp /home/student/easy-rsa/pki/private/client.key student@10.10.10.2:/etc/openvpn/client
Scp /home/student/easy-rsa/pki/ca.crt student@10.10.10.2:/etc/openvpn/client
Scp /home/student/easy-rsa/pki/issued/client.crt student@10.10.10.2:/etc/openvpn/client
Scp /etc/openvpn/server/ta.key student@10.10.10.2:/etc/openvpn/client
```

Na putanji /etc/openvpn/client je potrebno kreirati datoteku imena client.ovpn koja sadrži sljedeću konfiguraciju klijentske strane za OpenVPN:

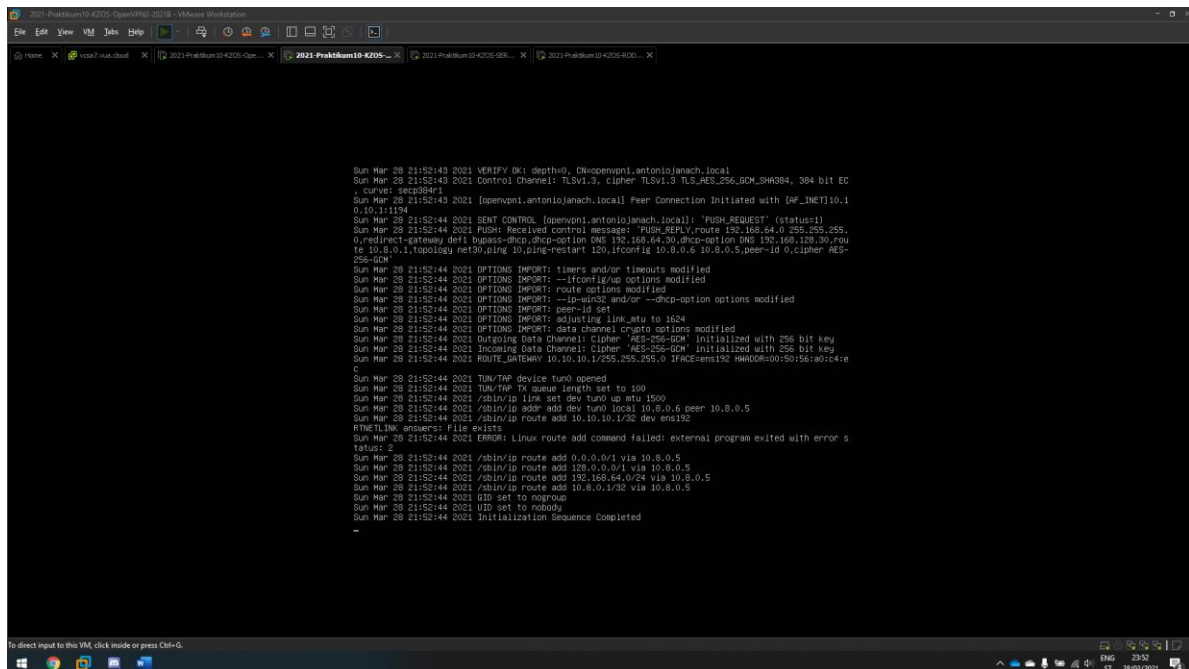
```
Touch client.ovpn
Echo -e „client
Dev tun
Proto udp
Remote 10.10.10.1 1194
Resolv-retry infinite
Nobind
User nobody
Group nogroup
Persist-key
Persist-tun
Ca /etc/openvpn/client/ca.crt
Cert /etc/openvpn/client/client.crt
Key /etc/openvpn/client/client.key
Remote-cert-tls server
Tls-crypt /etc/openvpn/client/ta.key
Cipher AES-256-GCM
Auth SHA256
Key-direction 1
Verb 3“ > /etc/openvpn/client/client.ovpn
```

Prilagođavanje konfiguracije vatrozida, do sada je instaliran OpenVPN, konfiguriran s generiranim ključevima i certifikatima potrebnim za pristup serverskom VPN-u. Međutim još nisu dane upute o tome kamo klijent treba poslati odlazni promet. Propustiti je potrebno: OpenVPN servis, uključiti masquerade i propustiti komunikaciju preko ens192 mrežnog adaptera IP adrese 10.10.10.0/24.

```
firewall-cmd -zone=public -add-service=openvpn -permanent
firewall-cmd -add-masquerade -permanent
firewall-cmd -permanent -direct -passthrough ipv4 -t nat -A POSTROUTING -s 10.8.0.0/24 -o ens224 -j MASQUERADE
firewall-cmd -permanent -direct -passthrough ipv4 -t nat -A POSTROUTING -s 10.10.10.0/24 -o ens192 -j MASQUERADE
firewall-cmd -reload
```

Klijentski poslužitelj openvpn2 spreman je za povezivanje koristeći sljedeće naredbe:

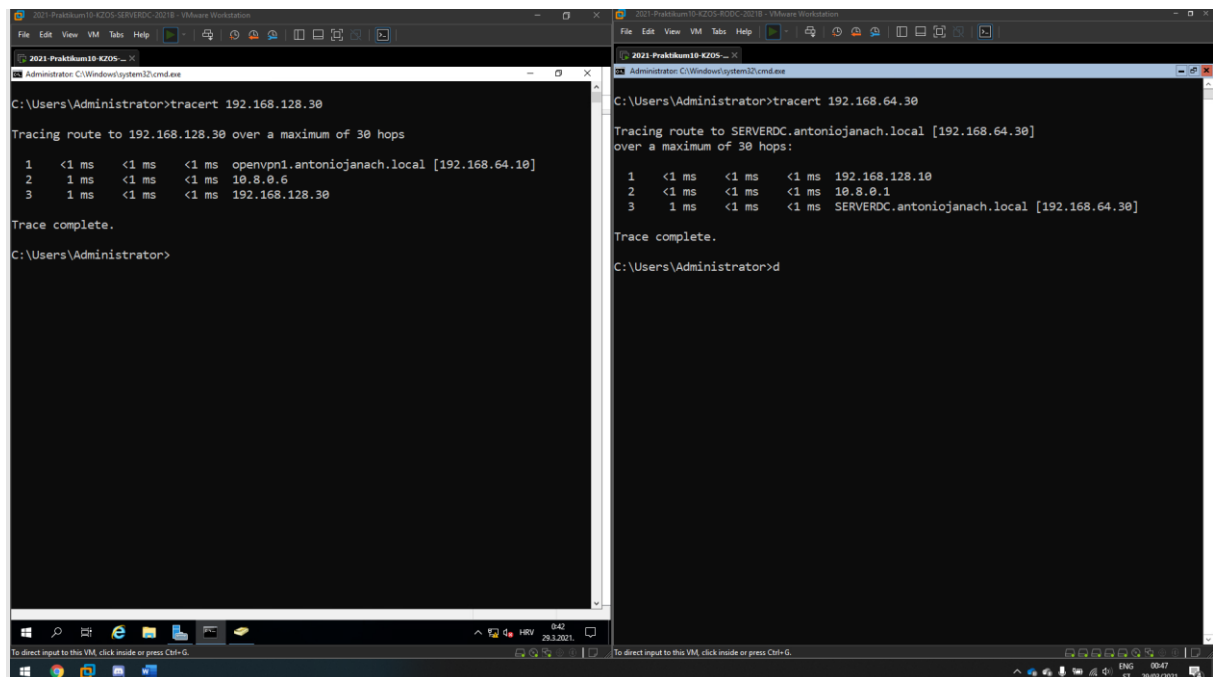
```
Openvpn --config /etc/openvpn/client/client.ovpn
```



```
Sun Mar 28 21:52:43 2021 VERIFY OK: depth=0, DN=openvpn1.antoniojanach.local
Sun Mar 28 21:52:43 2021 Control Channel: TLSv1.3, cipher TLSv1.3 TLS_AES_256_GCM_SHA384, 384 bit EC
Sun Mar 28 21:52:43 2021 [openvpn1.antoniojanach.local] Peer Connection Initiated with [AF_INET]10.10.10.111:1194
Sun Mar 28 21:52:44 2021 SENT CONTROL [openvpn1.antoniojanach.local]: 'PUSH_REQUEST' (status=1)
Sun Mar 28 21:52:44 2021 PUSH: Received control message: 'PUSH_REPLY,route 192.168.64.0 255.255.255.0,redirect-firewall def1,bypass-dhcp,script-option DNS 192.168.64.30,script-option DNS 192.168.123.30,route 10.8.0.1,topology net30,ping 10,ping-restart 120,ifconfig 10.8.0.6 10.8.0.5,peer-id 0,cipher AES-256-GCM'
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: timers and/or timeouts modified
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: --ifconfig/up options modified
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: route options modified
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: --ipaddr2 and/or --ppp-option options modified
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: peer-id set
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: adjusting link_mtu to 1624
Sun Mar 28 21:52:44 2021 OPTIONS IMPORT: data channel crypto options modified
Sun Mar 28 21:52:44 2021 Outgoing Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit Key
Sun Mar 28 21:52:44 2021 Incoming Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit Key
Sun Mar 28 21:52:44 2021 ROUTE_GATEWAY 10.10.1.255.255.255.0 IFRACE=ens192 HWADDR=00:50:56:a0:c4:e4
Sun Mar 28 21:52:44 2021 TUN/TAP device tun0 opened
Sun Mar 28 21:52:44 2021 TUN/TAP TX queue length set to 100
Sun Mar 28 21:52:44 2021 /sbin/ip link set dev tun0 up mtu 1500
Sun Mar 28 21:52:44 2021 /sbin/ip addr add dev tun0 local 10.8.0.6 peer 10.8.0.5
Sun Mar 28 21:52:44 2021 /sbin/ip route add 10.10.10.1/32 dev ens192
RTNETLINK answers: File exists
Sun Mar 28 21:52:44 2021 ERROR: Linux route add command failed: external program exited with error s
tatus=2
Sun Mar 28 21:52:44 2021 /sbin/ip route add 0.0.0.0/1 via 10.8.0.5
Sun Mar 28 21:52:44 2021 /sbin/ip route add 128.0.0.0/1 via 10.8.0.5
Sun Mar 28 21:52:44 2021 /sbin/ip route add 192.168.0.0/24 via 10.8.0.5
Sun Mar 28 21:52:44 2021 /sbin/ip route add 10.8.0.1/32 via 10.8.0.5
Sun Mar 28 21:52:44 2021 GID set to nobody
Sun Mar 28 21:52:44 2021 UID set to nobody
Sun Mar 28 21:52:44 2021 Initialization Sequence Completed
```

Slika 18: klijentski openvpn2 se je uspješno spojilo na serverski poslužitelj openvpn2

Sada kada je site-to-site VPN uspješno konfiguriran potrebno je podesiti mrežnu konfiguraciju openvpn1 i openvpn2 poslužitelja kako bi komunikacije između ServerDC i RODC poslužitelja bila moguća.



```
C:\Users\Administrator>tracert 192.168.128.30

Tracing route to 192.168.128.30 over a maximum of 30 hops:

  0  <1 ms  <1 ms  <1 ms  openvpn1.antoniojanach.local [192.168.64.10]
  1  <1 ms  <1 ms  <1 ms  10.8.0.6
  2  <1 ms  <1 ms  <1 ms  192.168.128.30

Trace complete.

C:\Users\Administrator>

C:\Users\Administrator>tracert 192.168.64.30

Tracing route to SERVERDC.antoniojanach.local [192.168.64.30]
over a maximum of 30 hops:

  0  <1 ms  <1 ms  <1 ms  192.168.128.10
  1  <1 ms  <1 ms  <1 ms  10.8.0.1
  2  <1 ms  <1 ms  <1 ms  SERVERDC.antoniojanach.local [192.168.64.30]

Trace complete.

C:\Users\Administrator>
```

Slika 19: prikaz uspješne komunikacije između ServerDC i RODC poslužitelja koristeći VPN vezu

Time je dokazano da paket uspješno prolazi VPN tunelom s lokacije A do lokacije B.

6.8. Postavljanje SSH servisa i razmjena ključeva

Konfiguracijom VPN servisa cijela infrastruktura koja se sastoji od računala može komunicirati međusobno što je idealno vrijeme za razmjenu SSH ključeva. Ključevi se razmjenjuju tako da se je na određenom korisniku root/student potrebno upisati sljedeću komandu za generiranje ključeva.

```
Ssh-keygen
```

Otvora se wizard u terminalu od Linux-a gdje se mora upisati putanja spremanja ključa. Za putanju spremanja ključa odabrana je: /root/.ssh/imeključa. Isto tako i za korisnika student.

Kad je ključ kreiran potrebno ga je razmijeniti s ostalim računalima na sljedeći način:

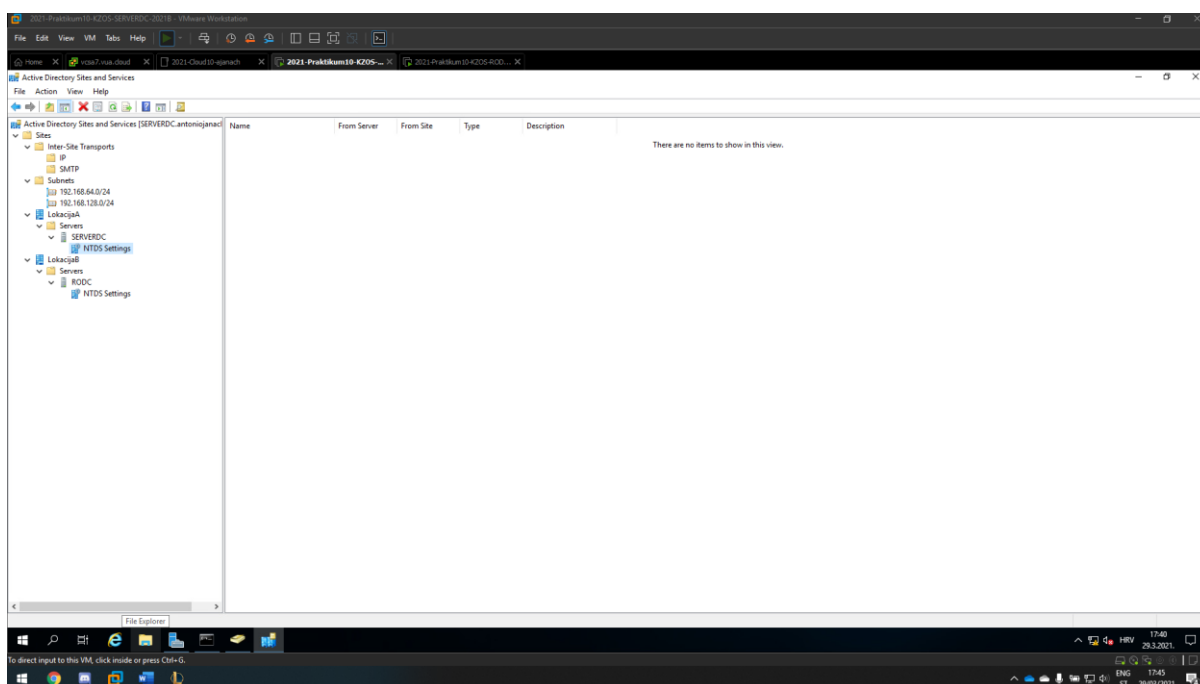
```
Ssh-copy-id -i /root/.ssh/imeključa root@DNS_zapis_drugog_poslužitelja
```

Da bi se ključ mogao razmijeniti s Ubuntu poslužiteljima u konfiguracijskoj datoteci potrebno otkomentirati parametar „PermitRootLogin yes“ da bi se dopustilo spajanje na Ubuntu mašine s root korisnikom.

6.9. Konfiguracija domene na lokaciji B

Preporuka kod većih i bitnih serverskih okruženja jest posjedovanje više DC zbog mogućnosti pada jednog. U okruženju s više DC-a oni međusobno repliciraju „ntds.dit“ podatak. Ova se akcija naziva „Peer replication“. U procesu instalacije moguće je izabrati između „Full“, „Read only DC“ (najčešće na Windows Core edicijama na udaljenim lokacijama i to uglavnom iz sigurnosnih razloga), „Forest root“ (glavni DC u šumi domeni), „Peer dc“ (sekundarni failover kontroler koji sinkronizira ntds.dit zbog dodatne sigurnosti domene).

Za replikaciju informacija podataka između DC-a, potrebno je dodati ostale DC putem „Active directory sites and services“ konzole.

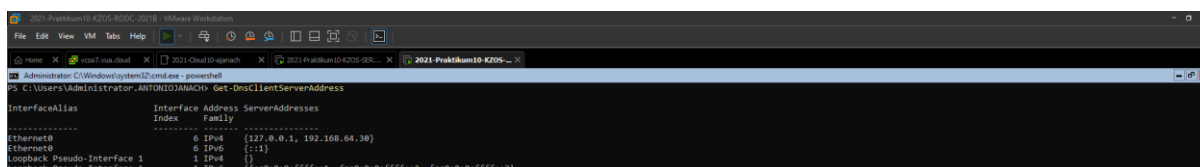


Slika 20: prikaz konfiguracije u "Active directory sites and services" na SERVERDC poslužitelju

Kako bi se dodao RODC kao „Read only domain controller“ potrebno je kroz PowerShell instalirati ulogu „Active Directory Domain services“.

```
Install-WindowsFeature -Name ad-domain-services -IncludeManagementTools -  
IncludeAllSubFeature
```

Zatim je potrebno dodati IP adresu od ServerDC računala na RODC poslužitelju.

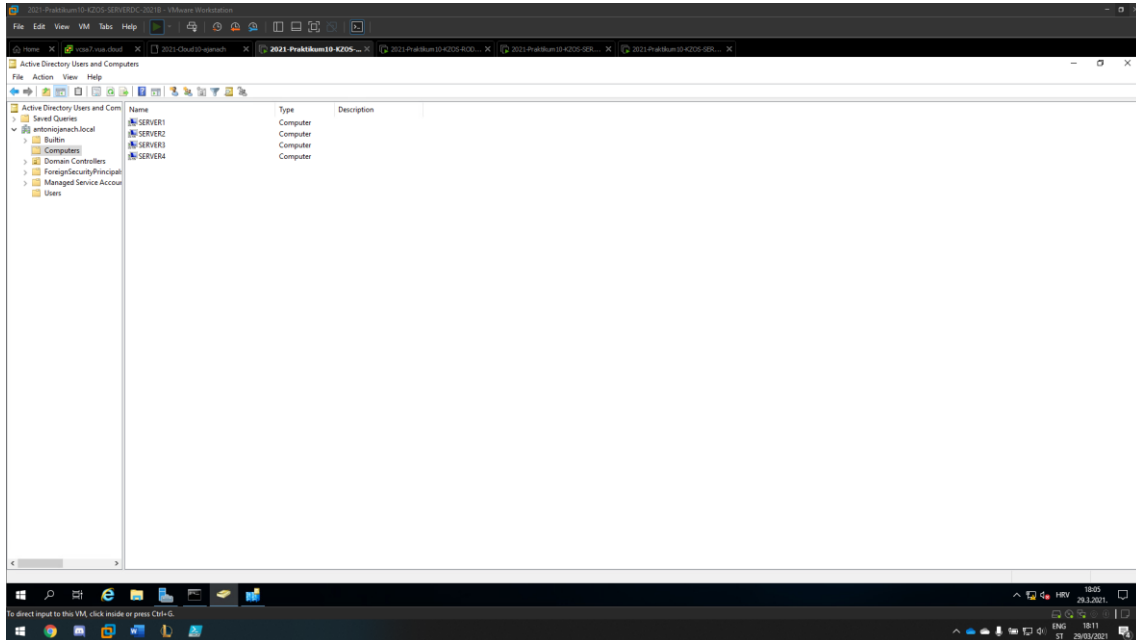


Slika 21: DNS konfiguracija na RODC poslužitelju

Za kraj je potrebno dodati RODC u antoniojanach.local domenu kao „Read only domain controller“.

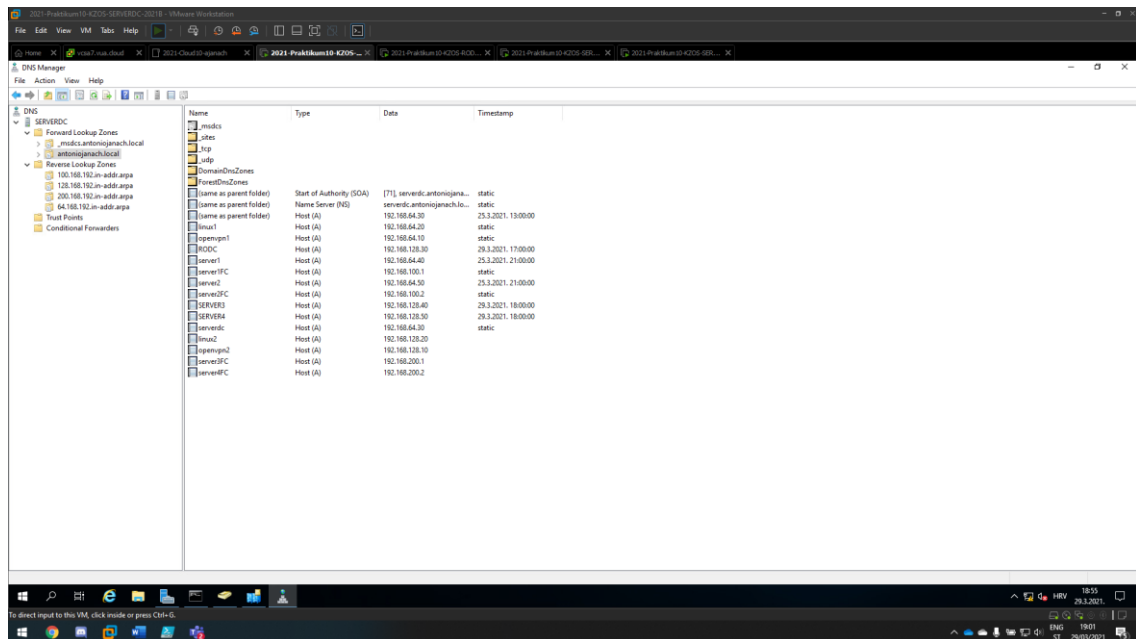
```
Install-ADDSDomainController -Credential (Get-Credential) -DomainName antoniojanach.local -  
InstallDNS:$true -ReadOnlyReplica:$true -Force:$true
```

Kad je RODC uspješno dodan u domenu, potrebno je dodati Server3 i Server4 poslužitelj koristeći DNS IP adresu RODC poslužitelja.



Slika 22: prikaz svih dodanih Windowd poslužitelja u domenu

I za kaj potrebno je dodati DNS zapise svih poslužitelja za lokaciju B.



Slika 23: prikaz DNS konfiguracije na ServerDC poslužitelju

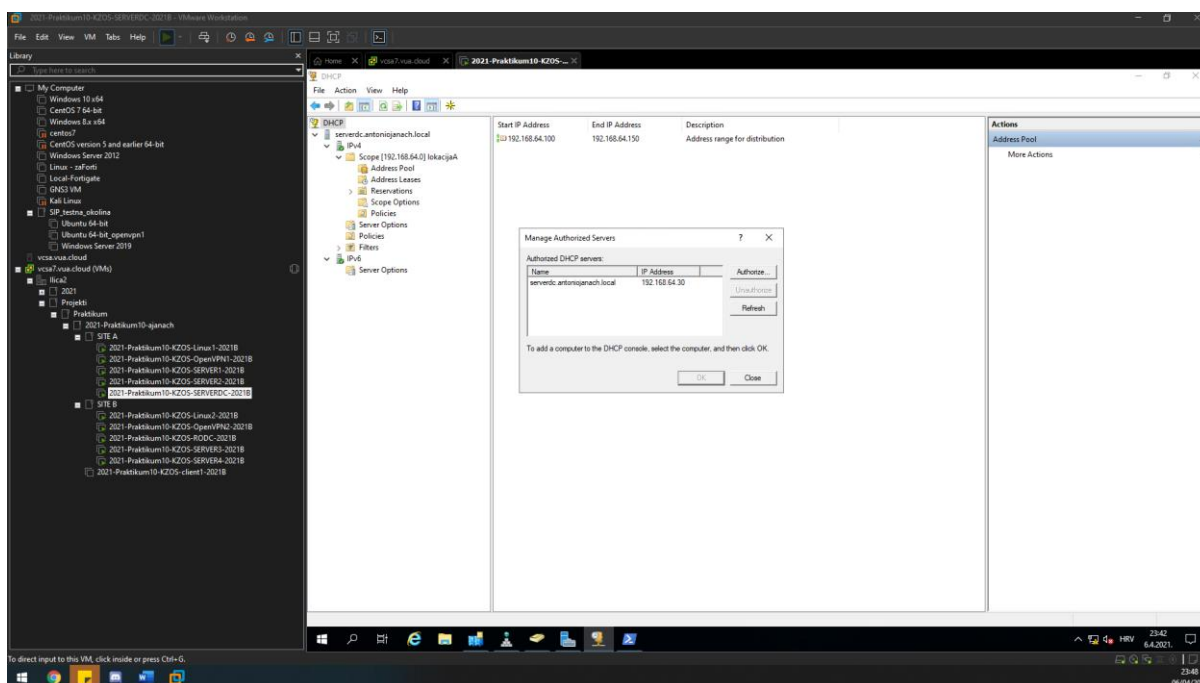
Ovime završava konfiguracija domene na lokaciji B gdje svako računalo može komunicirati s drugim koristeći FQDN poslužitelja.

6.10. Konfiguracija DHCP poslužitelja na SERVERDC i RODC

DHCP služi automatiziranoj dodjeli IP postavki računalima na IP mreži. Te postavke, osim IP adrese i mrežne maske, uključuju zadani usmjernik i adresu DNS poslužitelja. Cilj je da bilo koji korisnik s fizičkim pristupom mreži može dobiti postavke DHCP poslužitelja. Isto tako, ako je DHCP poslužitelj instaliran na domenski kontroler, on se automatski autorizira s AD-om. No, ako je DHCP poslužitelj instaliran na bilo koji drugi server koji je član domene potrebna su mu „Enterprise Admin“ prava za autorizaciju.

6.10.1. Konfiguracija DHCP poslužitelja na SERVERDC

Kod instalacije DHCP uloge nužno je autorizirati DHCP server pute domenskog administratora. Kad je DHCP server uloga instalirana i autorizirana od strane domenskog administratora potrebno je dodati raspon adresa iz mrežnog *subnet-a* lokacije A. Ovim putem dodana je raspon adresa od 192.168.64.100 do 192.168.64.150. Isto tako „Lease Duration“ postavljen je na 8 sati i kao „Gateway“ postavljeno je Linux računalo „openvpn1.antoniojanach.local“.

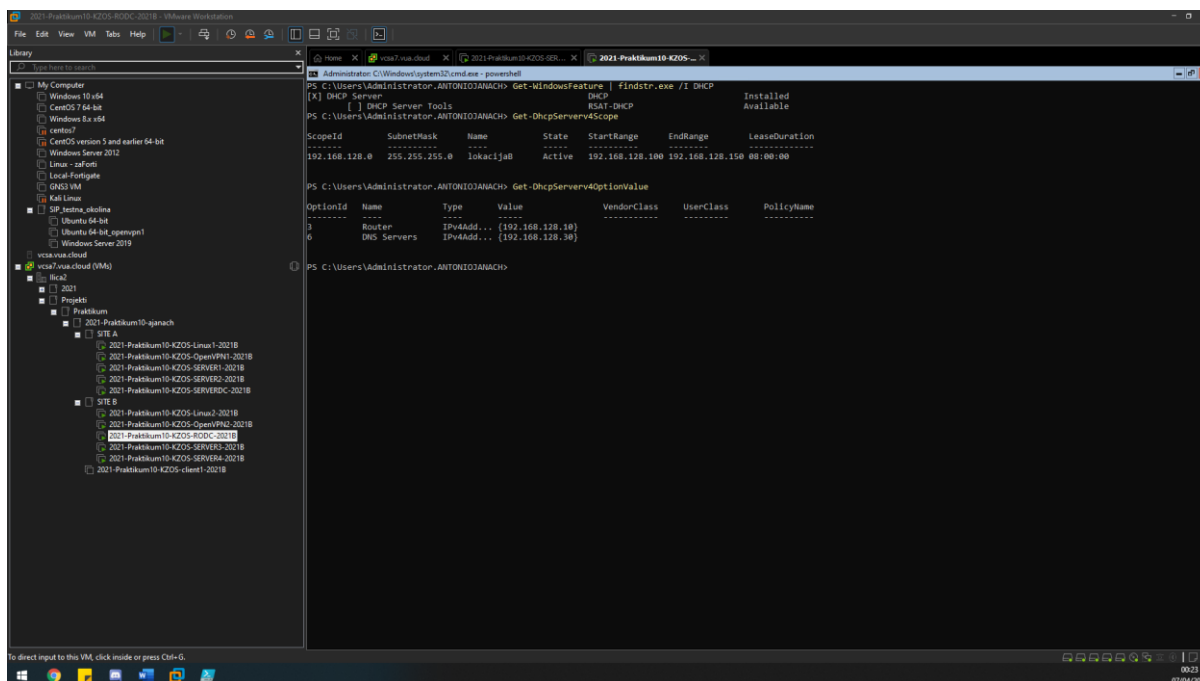


Slika 24: prikaz dodanog raspona adresa i autorizacije DHCP servera

Ovime završava konfiguracija DHCP servera na SERVERDC poslužitelju. Sljedeće što je potrebno, a to je konfigurirati DHCP ulogu na RODC poslužitelj.

6.10.2. Konfiguracija DHCP poslužitelja na RODC

Kao i kod konfiguracije DHCP uloge na SERVERDC poslužitelju vrijedi isti princip, samo što je potrebno sve odraditi kroz PowerShell. Raspon adresa koji je dodan je od 192.168.128.100 do 192.168.128.150. Dodan je DNS server u ovom slučaju IP adresa od RODC poslužitelja(192.168.128.30) i „Gateway“ od Openvpn2 poslužitelja(192.168.128.10). Postavljen je „Lease Duration“ na 8 sati.



Slika 25: prikaz instalirane DHCP uloge i postavke DHCP poslužitelja

U nastavku su prikazani PowerShell CMDlet-i koji su korišteni za konfiguraciju DHCP poslužitelja.

```
#instalacije DHCP uloge na RODC poslužitelj
Install-WindowsFeature -Name DHCP -IncludeAllSubFeature -IncludeManagementTools
#dodavanje novog raspona adresa:
Add-DhcpServerV4Scope -Name "lokacijaB" -StartRange 192.168.128.100 -EndRange 192.168.128.150 -
SubnetMask 255.255.255.0
#dodavanje DNS servera koji je u ovom slučaju RODC i Gateway openvpn2.antoniojanach.local poslužitelja
Set-DhcpServerV4OptionValue -DnsServer 192.168.128.30 -Router 192.168.128.10
#postavljanje Lease Duration-a na 8 sati
Set-DhcpServerV4Scope -ScopeId 192.168.128.0 -LeaseDuration 0.08:00:00
```

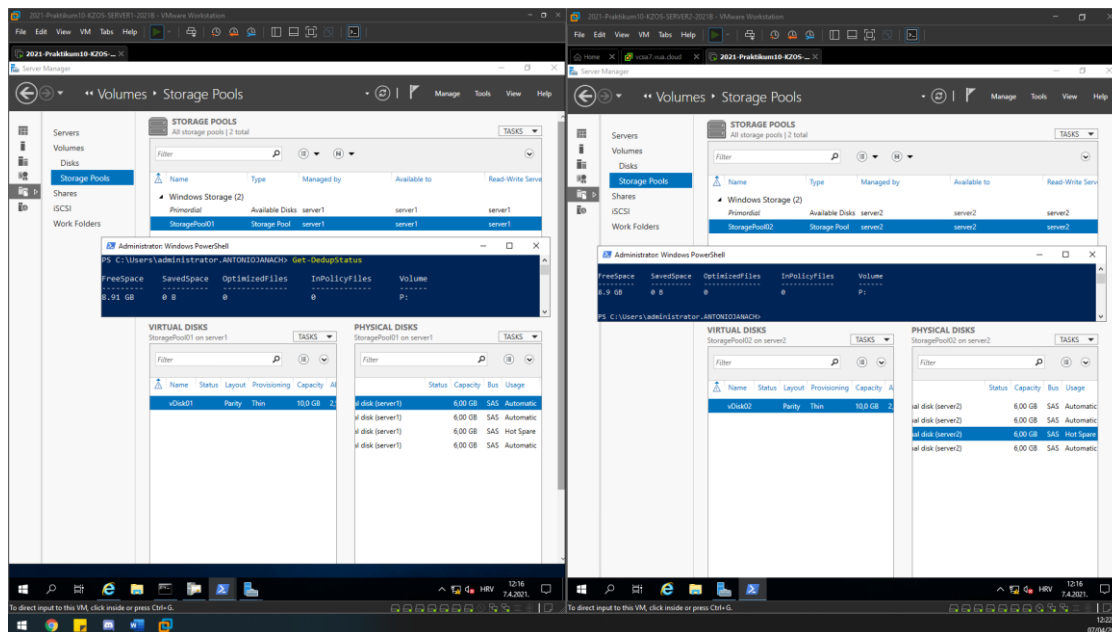
Kao što se vidi iz navedenog, DHCP uloga uspješno je konfigurirana na RODC poslužitelju.

6.11. Storage spaces konfiguracija i konfiguracija deduplikacije na Server1 i Server2

„Storage spaces“ je zamišljen kao fleksibilan način organizacije diskova različitih kapaciteta, pa čak i način priključenja, u redundantna polja. Naravno, idealna situacija je korištenje diskova istih veličina, ali zamisao je, barem u manjim okruženjima, iskoristiti diskove koje imamo na raspolaganju za povećanje kapaciteta poslužitelja ili čak klijentskog računala. Jednom implementiran „Storage spaces“ koristi virtualne diskove za pohranu podataka. Također prema zahtjevima potrebno je uključiti značajku uklanjanja duplikata. Značajka uklanjanja duplikata kao i samo ime govori, riječ je o automatiziranom mehanizmu čija je zadaća povećati učinkovitost iskorištenja diskovnog prostora automatskim uklanjanjem duplih datoteka.

Nakon što je „Storage Pool“ kreiran od četiri diska gdje je jedan „Hot spare“, potrebno je kreirati virtualni disk koji koristi ReFS datotečni sustav.

Kad je kreiran virtualni disk nužno je kreirati novi volumen koji koristi GPT particijsku tablicu. Također nad tim diskom potrebno je uključiti deduplikaciju. Kako bi se uključila deduplikacija potrebno je instalirati tu značajku.



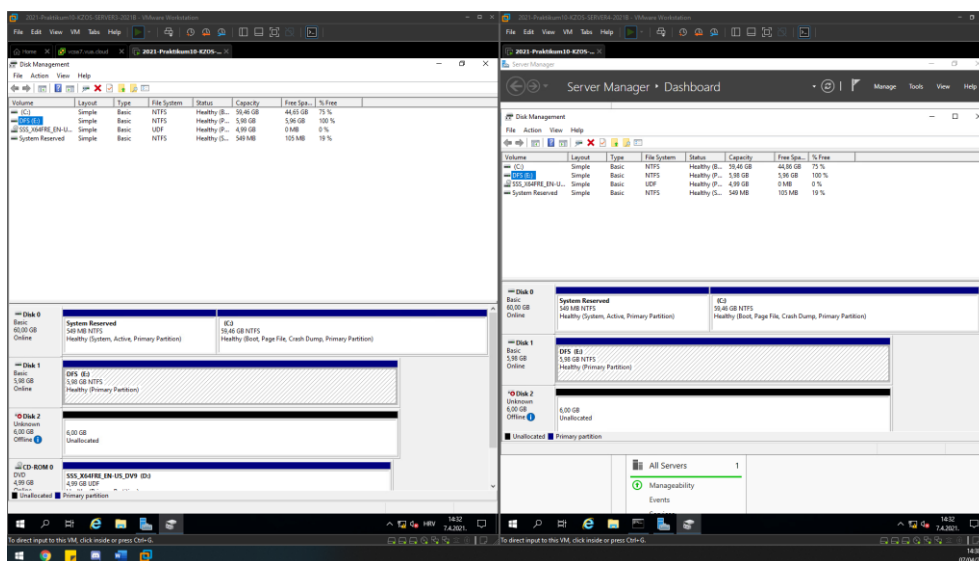
Slika 26: Prikaz kreiranog „Storage pool-a“, virtualnog diska, volumena i uključene deduplikacije nad volumenom

Kao što se vidi iz prethodne slike uspješno je konfigurirana *Storage spaces* konfiguracija te zadovoljava svim uvjetima zahtjeva koji su prethodno navedeni.

6.12. DFS + R konfiguracija na SERVER3 i SERVER4 poslužitelju

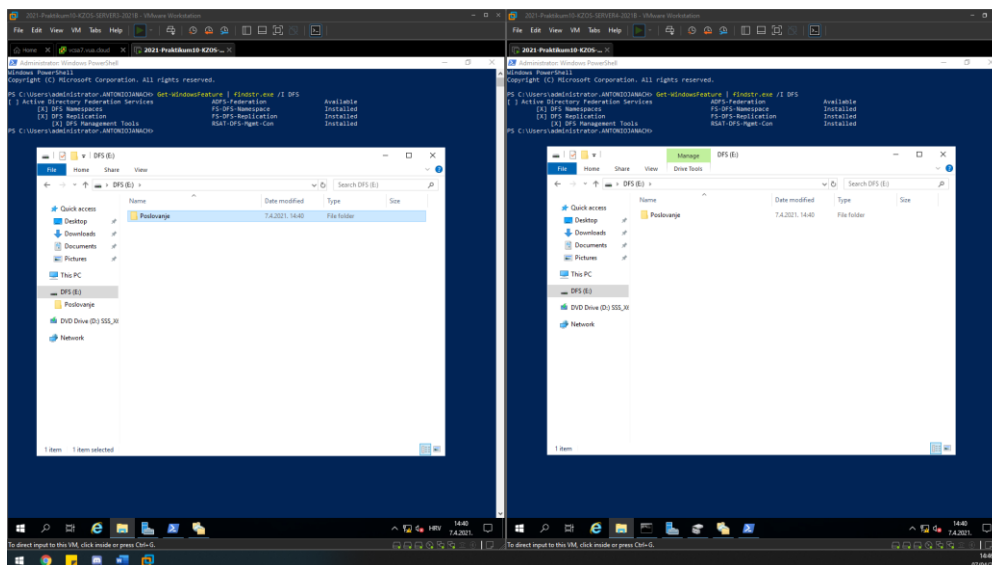
Uloga distribuiranog datotečnog sustava - DFS je da od krajnjeg korisnika djelomično sakri kompleksnost mrežne pohrane podataka. Naime, kako bi korisnik pristupio određenoj mrežnoj mapi, mora znati njezinu UNC (eng. *Universal Naming Convention*) putanju, koja je na Windows sustavima oblika \\FQDN\imeMape. DFS također omogućuje spajanje više mapa s različitih poslužitelja u jednu virtualnu mapu, koju onda predstavlja korisnicima. Na korisniku je samo da otvori početnu lokaciju, a DFS mehanizmi njegove će zahtjeve preusmjeriti na poslužitelj koji datoteke zaista ima.

Prije same instalacije DFS uloge potrebno je kreirati novi volumen na prvom slobodnom disku (disk 1). Da bi kreirali volumen nužno je postaviti disk u online stanje, formatirati ga u GPT partijsku tablicu i potom kreirati volumen.



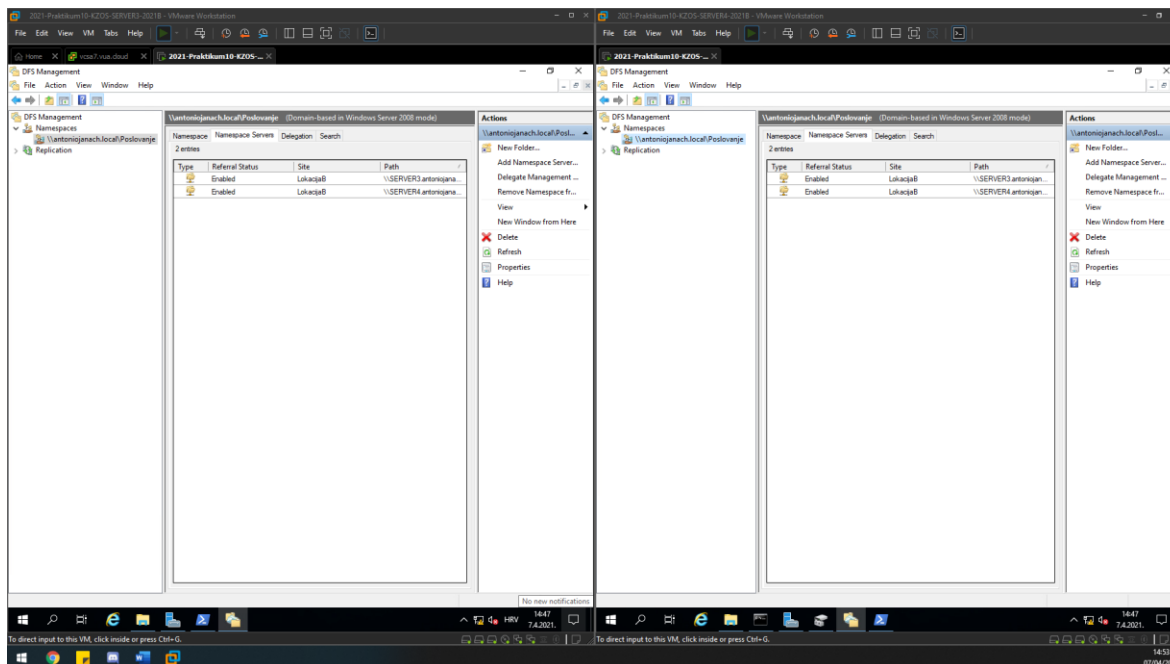
Slika 27: prikaz kreiranog volumena na SERVER3 i SERVER4 poslužitelju imena "DFS"

Kad su diskovi spremni na korištenje potrebno je instalirati DFS ulogu zajedno sa „DFS Namespaces“ i „DFS Replication“ na oba poslužitelja (SERVER3 i SERVER4). Kad se uloga instalira na oba poslužitelja potrebno je kreirati datoteku imena „Poslovanje“ na novokreiranom volumenu.



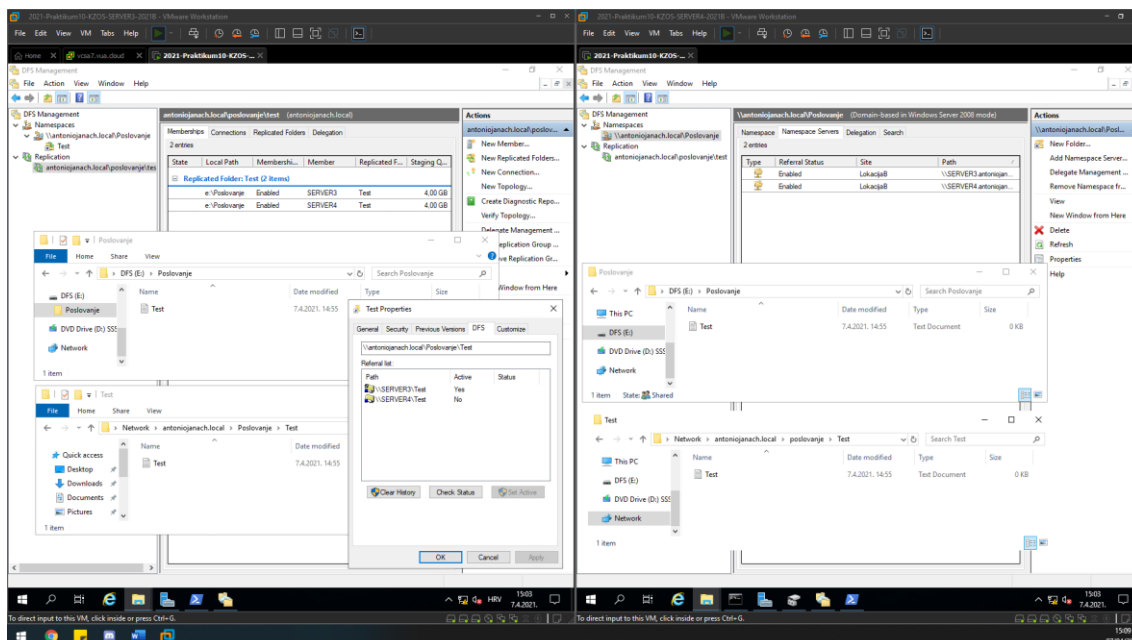
Slika 28: prikaz instalirane DFS uloge i kreirane mape imena „Poslovanje“ na novokreiranom volumenu

Daljnja konfiguracija odvija se u sučelju imena „DFS Management“. Kad se kreira *namespaces* imena „Poslovanje“, tom *namespaces-u* potrebno je dodati *namespace* servere, tj. SERVER4.



Slika 29: prikaz kreiranog DFS namespace-a i dodanog Server4 poslužitelja

Kad je kreiran *namespaces* imena „Poslovanje“ i dodan Server4 poslužitelja, potrebno je tom *namespaces-u* dodati foldere. Svaki folder mora imati folder *target* s oba računala (Server3 i Server4). Kad su kreirani i jedan i drugi *target* folder na tom folderu potrebno je pokrenuti replikaciju. Za primarni server postavljen je Server3.



Slika 30: prikaz uspješne konfiguracije DFS + R na Server3 i Server4 poslužitelju

DFS + R konfiguracija je uspješna i provjerena tako što sam kreirao .txt datoteku imena Test na UNC putanji \\antoniojanach.local\poslovanje\test. Kao što je vidljivo iz prethodne slike replikacija uspješno radi.

6.13. Diskovna i iSCSI konfiguracija na Linux1 poslužitelju

Prema zahtjevu koji je zadan na Linux1 poslužitelju potrebno je kreirati jednu volumen grupu imena „iSCSI01“. Zatim je iz te volumen grupe potrebno izrezati tri logička volumena imena LUN0, LUN1, LUN2. Logičke volumene je potrebno kroz iSCSI Target prikazati kao LUN 0, 1 i 2 prema poslužiteljima Server1 i Server2. Ti udaljeni diskovi koji su predstavljeni s udaljenog iSCSI poslužitelja iskoristit će se za podizanje „Windows Failover Clusteringa“.

U suštini iSCSI je protokol na razini bloka za upravljanje uređajima za pohranu preko TCP / IP mreže, posebno za korištenje na velikim udaljenostima. iSCSI Target je udaljeni tvrdi disk predstavljen s udaljenog iSCSI poslužitelja. Dok s druge strane stoji iSCSI klijent koji se naziva Inicijator, preko kojeg se pristupa spremištu koje dijeli Target.

Problematiku ovoga zahtjeva riješio sam sljedećom skriptom koja je pokrenuta na Linux1 poslužitelju:

```
#!/bin/bash
#kreiranje particija
fdisk /dev/sdb <<EOF n p 1 2048 20971519 t 8e w EOF

fdisk /dev/sdc <<EOF n p 1 2048 20971519 t 8e w EOF

fdisk /dev/sdd <<EOF n p 1 2048 20971519 t 8e w EOF

fdisk /dev/sde <<EOF n p 1 2048 20971519 t 8e w EOF

#kreiranje fizickog volumena:
pvcreate /dev/sdb1
pvcreate /dev/sdc1
pvcreate /dev/sdd1
pvcreate /dev/sde1

#kreiranje grupe volumena:
vgcreate iSCSI01 /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1

#kreiranje logickog volumena:
lvcreate -n LUN0 -L 7GiB iSCSI01
lvcreate -n LUN1 -L 7GiB iSCSI01
lvcreate -n LUN2 -L 7GiB iSCSI01

#instalacija potrebnih servisa:
yum update -y
yum install target* -y
systemctl enable target
systemctl enable target

#konfiguracija u targetcli-u:
targetcli /backstores/block create LUN0 /dev/iSCSI01/LUN0
targetcli /backstores/block create LUN1 /dev/iSCSI01/LUN1
targetcli /backstores/block create LUN2 /dev/iSCSI01/LUN2
targetcli /iscsi create iqn.2021-04.local.antoniojanach:target
targetcli /iscsi/iqn.2021-04.local.antoniojanach:target/tpg1/acls create iqn.2021-04.local.antoniojanach:server1
targetcli /iscsi/iqn.2021-04.local.antoniojanach:target/tpg1/acls create iqn.2021-04.local.antoniojanach:server2
```

```
targetcli /iscsi/iqn.2021-04.local.antoniojanach:target/tpg1/luns create /backstores/block/LUN0
targetcli /iscsi/iqn.2021-04.local.antoniojanach:target/tpg1/luns create /backstores/block/LUN1
targetcli /iscsi/iqn.2021-04.local.antoniojanach:target/tpg1/luns create /backstores/block/LUN2

#kroz firewalld propustiti iSCSI promet:
firewall-cmd --add-service=iscsi-target --permanent
firewall-cmd --add-port=3260/tcp --permanent
firewall-cmd --reload

#kraj skripte:
echo -e "\n skripta je gotova."
```

```
root@linux1:~/Documents
File Edit View Search Terminal Help
root@linux1:~/Documents# targetcli ls
o- /
  o- backstores ..... [Storage Objects: 3]
    | o- block ..... [/dev/iscsi01/LUN0 (7.001B) write-thru activated]
    | | o- alua ..... [ALUA Groups: 1]
    | | | o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    | | | o- LUN0 ..... [/dev/iscsi01/LUN0 (7.001B) write-thru activated]
    | | | o- alua ..... [ALUA Groups: 1]
    | | | | o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    | | | o- LUN2 ..... [/dev/iscsi01/LUN2 (7.001B) write-thru activated]
    | | | o- alua ..... [ALUA Groups: 1]
    | | | | o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
    | o- fileio ..... [Storage Objects: 0]
    o- pscsi ..... [Storage Objects: 0]
    o- ramdisk ..... [Storage Objects: 0]
  o- iscsi ..... [Targets: 1]
    | o- iqn.2021-04.local.antoniojanach:target ..... [TPGs: 1]
    | | o- tpg1 ..... [no-gen-acls, no-auth]
    | | | o- acls ..... [Mapped LUNs: 3]
    | | | | o- iqn.2021-04.local.antoniojanach:server1 ..... [Mapped LUNs: 3]
    | | | | | o- mapped_lun0 ..... [lun0 block/LUN0 (rw)]
    | | | | | o- mapped_lun1 ..... [lun1 block/LUN1 (rw)]
    | | | | | o- mapped_lun2 ..... [lun2 block/LUN2 (rw)]
    | | | | o- iqn.2021-04.local.antoniojanach:server2 ..... [Mapped LUNs: 3]
    | | | | | o- mapped_lun0 ..... [lun0 block/LUN0 (rw)]
    | | | | | o- mapped_lun1 ..... [lun1 block/LUN1 (rw)]
    | | | | | o- mapped_lun2 ..... [lun2 block/LUN2 (rw)]
    | | | o- luns ..... [LUNs: 3]
    | | | | o- lun0 ..... [block/LUN0 (/dev/iscsi01/LUN0) (default_tg_pt_gp)]
    | | | | o- lun1 ..... [block/LUN1 (/dev/iscsi01/LUN1) (default_tg_pt_gp)]
    | | | | o- lun2 ..... [block/LUN2 (/dev/iscsi01/LUN2) (default_tg_pt_gp)]
    | | o- portals ..... [Portals: 1]
    | | | o- 0.0.0.0:3260 ..... [Lun]
    o- loadback ..... [Targets: 0]
root@linux1:~/Documents# ls
iscsi.sh
root@linux1:~/Documents#
```

Slika 31: prikaz završne iSCSI konfiguracije na Linux1 poslužitelju

Kao što je vidljivo u prethodnoj slici iSCSI target je uspješno konfiguriran na Linux1 poslužitelju.

6.14. Windows File Server Cluster konfiguracija na Server1 i Server2

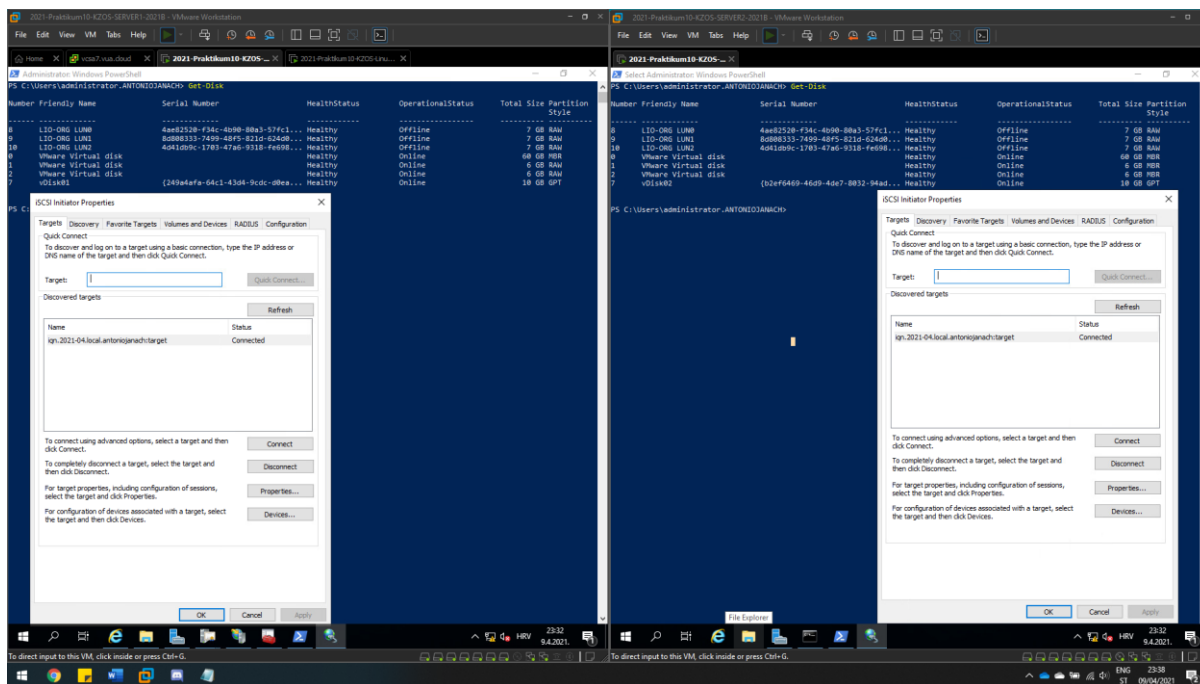
„Failover Clustering“ skupina je neovisnih računala koja zajedno rade na povećanju dostupnosti i skalabilnosti klasteriziranih uloga. Klasterizirani poslužitelji su međusobno povezani i, ako jedan od čvorova klastera padne, drugi čvorovi počinju preuzimati ulogu. Klaster koji implementiramo je baziran na iSCSI skladištu podataka.

Za početak potrebno je na Windows poslužitelje spojiti dijeljene iSCSI diskove koristeći iSCSI inicijator i te diskove postaviti u online stanje, formatirati ih u partijsku tablicu i kreirati volumene.

U konfiguraciji *iSCSI target* poslužitelja *iqn-ovi* poslužitelja koji se spajaju na *iSCSI target* glase:

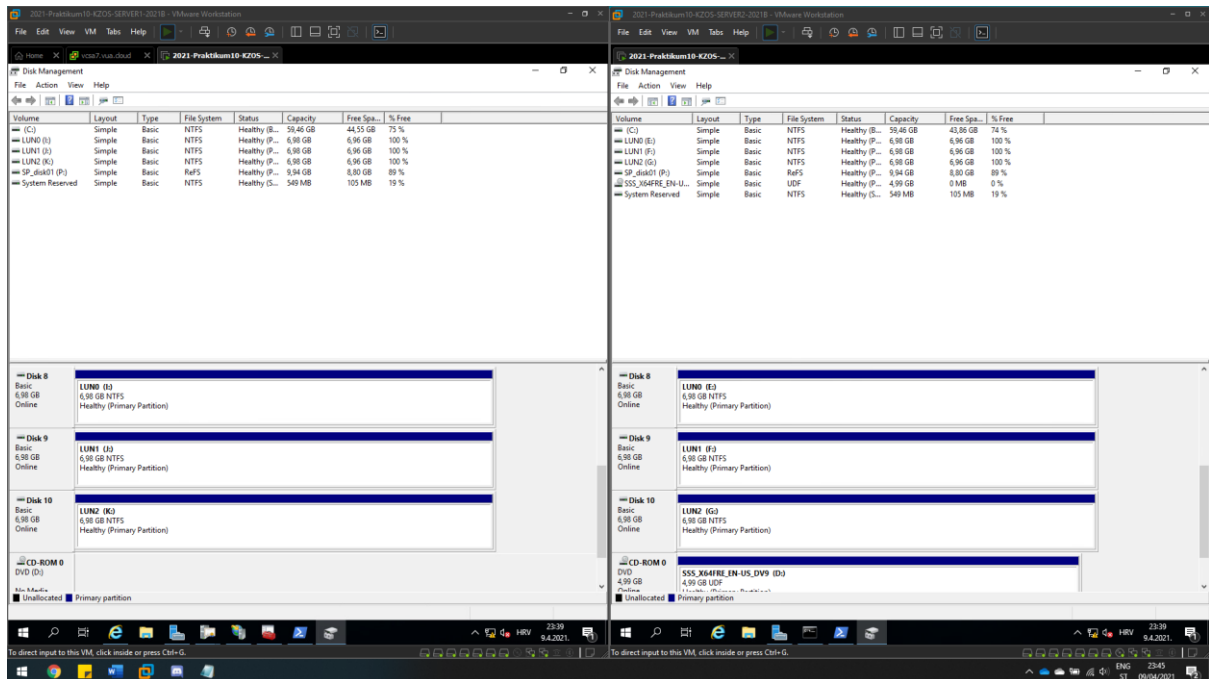
- SERVER1: iqn.2021-04.local.antoniojanach:server1
- SERVER2: iqn.2021-04.local.antoniojanach:server2

Imena *iqn-ova* potrebno je promijeniti u „iSCSI Initiator“ postavkama. Kad su postavke promijenjene u kartici „Initiator Name“ u kartici „Discover“ potrebno je spojiti dijeljene diskove putem *iSCSI Target-a*.



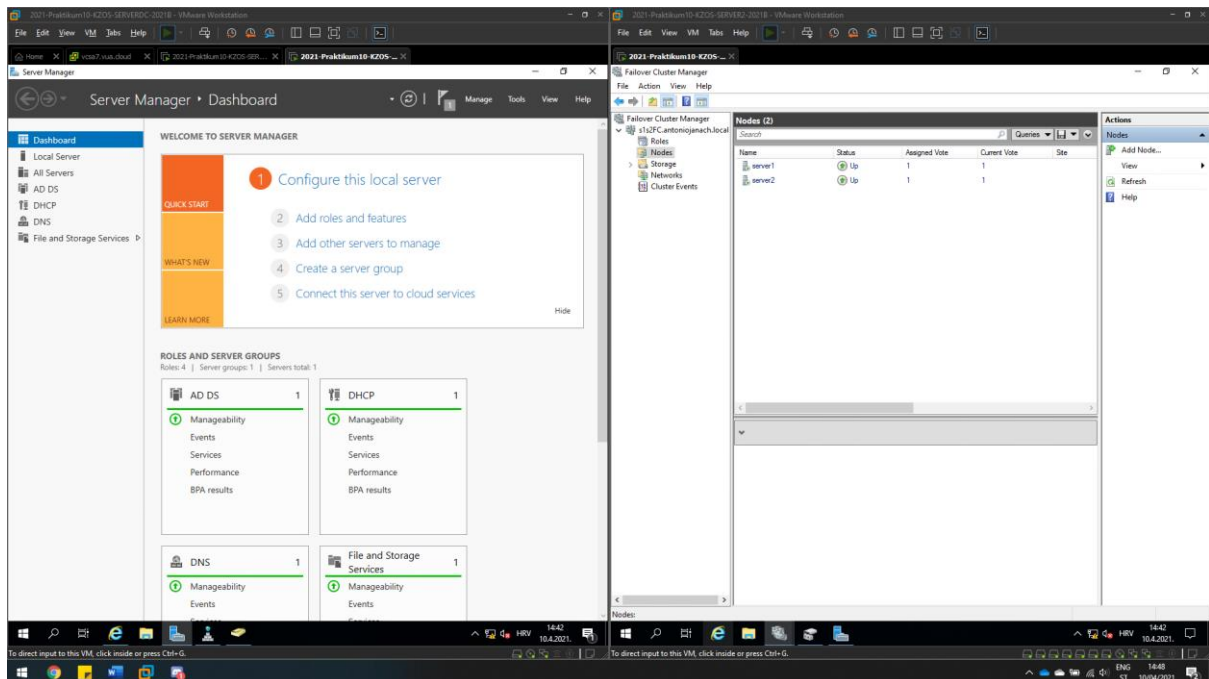
Slika 32: prikaz uspješnog spajanja na iSCSI target koristeći iSCSI inicijator

Sljedeće što je potrebno kad su diskovi spojeni, a to je diskove postaviti u online stanje, formatirati ih u GPT particijsku tablicu i kreirati volumene.



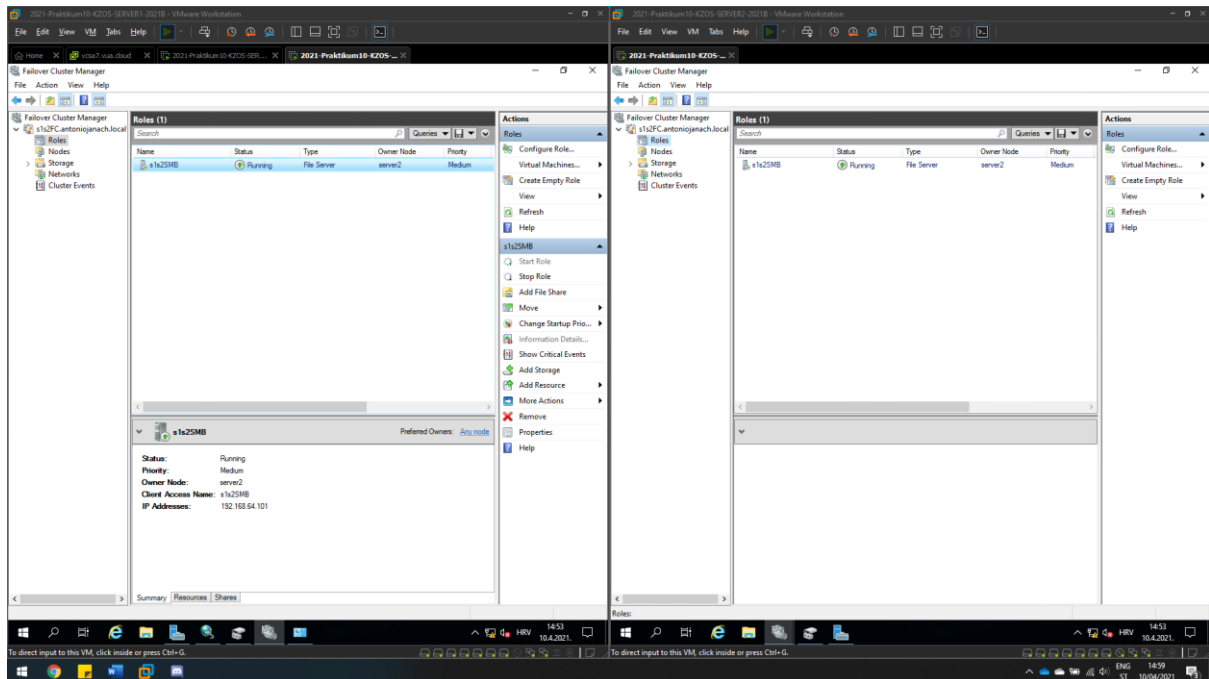
Slika 33: prikaz konfiguracije diskova

Kad su diskovi uspješno konfigurirani, potrebno je instalirati „Failover Clustering“ značajku na oba poslužitelja. Kad je značajka uspješno instalirana u „Failover Cluster Manager“ je potrebno pokrenuti validaciju konfiguracije. Nakon što validacija uspješno provjeri sve komponente, potrebno je kreirati klaster imena „s1s2FC“ u mrežnom rasponu 192.168.64.0/24, IP adrese 192.168.64.100. Također pod tom IP adresom napravljen je DNS zapis.



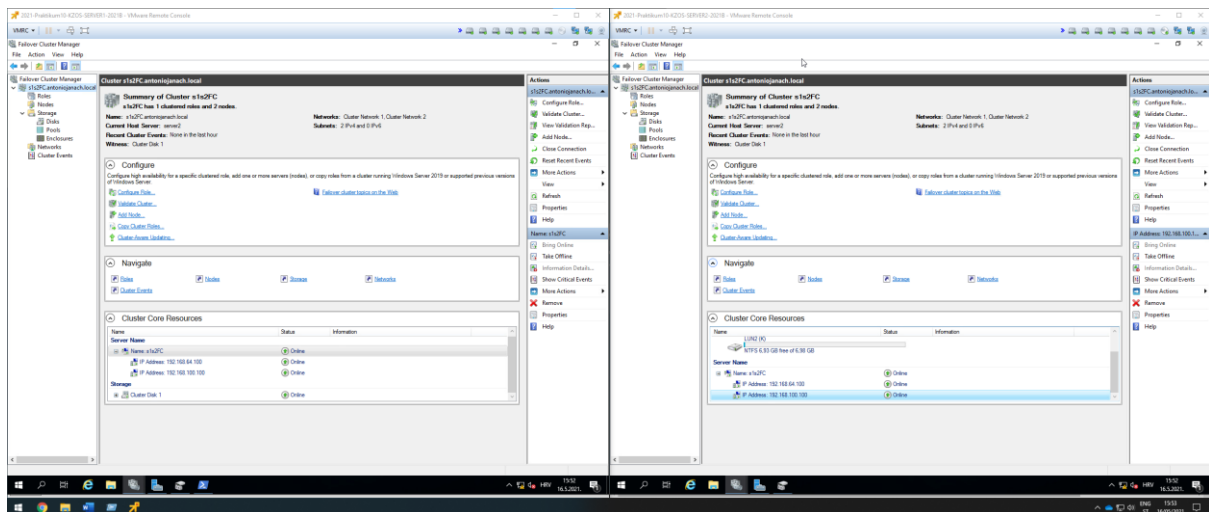
Slika 34: prikaz uspješno kreiranog klastera na SERVER1 i SERVER2 poslužitelju

Sad kad je kreiran klaster, potrebno je konfigurirati File Server ulogu nad tim klasterom. U kartici „Roles“ desnim klikom miša odabrati „Configure Role“ i slijediti upute sa čarobnjaka.

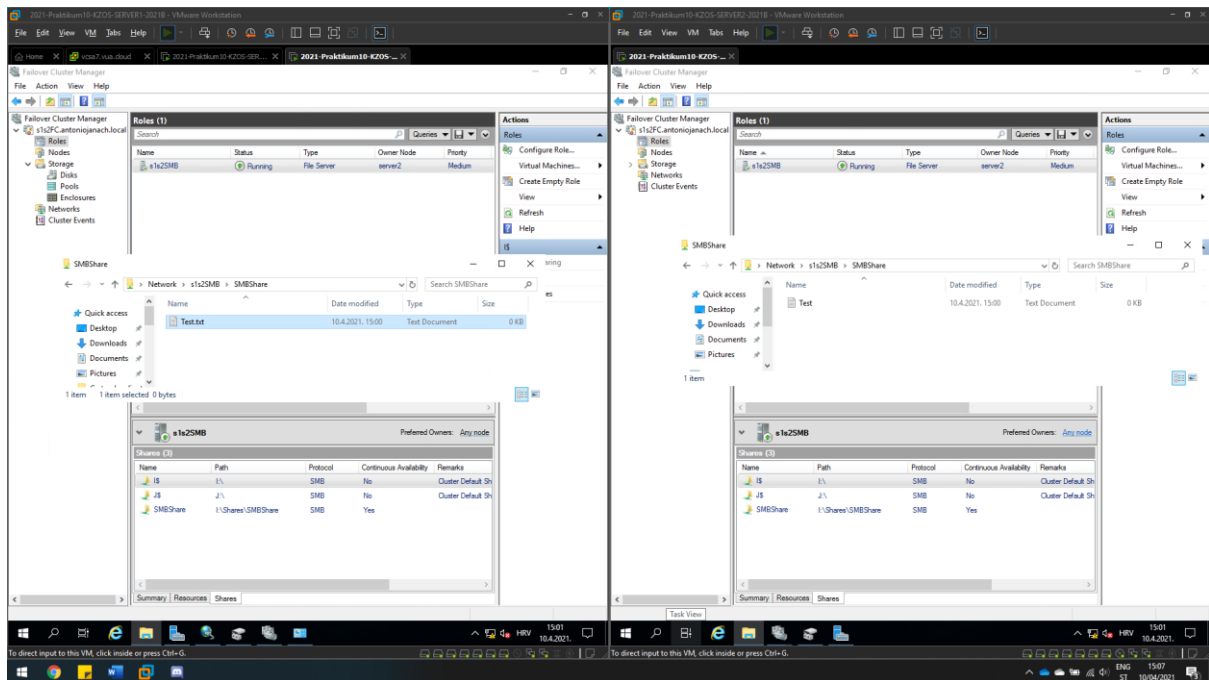


Slika 35: prikaz uspješno konfigurirane uloge "File Server"

Nakon kreiranog klastera potrebno je u „Networks“ dijelu „Failover Cluster“ management role dozvoliti opciju „Allow clients to connect through this network“ kako bi se dodala druga mrežna kartica za komunikaciju s pripadajućim parom u failover clustering-u.



Kad je uloga uspješno konfigurirana preostalo je konfigurirati SMB dijeljeni disk na SMB ulozu. Potrebno je dodati „SMB Share -Quick“. Lokalna putanja diska je: I:\Shares\SMBShare, dok je dijeljena mapa: \\s1s2SMB\SMBShare.



Slika 36: prikaz uspješno kreiranog dijeljenog SMB diska

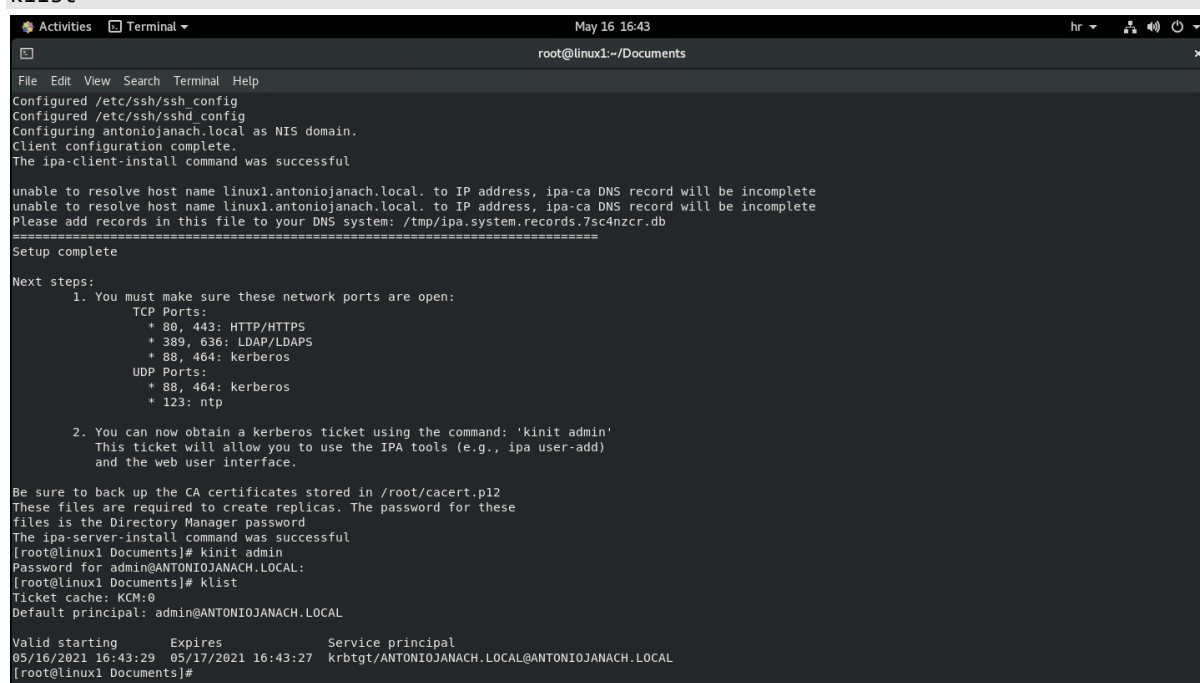
Ovime završava konfiguracija „Windows File Server Cluster“ na Server1 i Server2 poslužitelju.

6.15. Konfiguracija CentOS poslužitelja - lokacija A

6.15.1. Dodavanje FreeIPA servisa i sync s Active Directory-om

Server za FreeIPA servis bit će Linux1 poslužitelj koji se koristi kao domenski servis za Linux poslužitelje. Da bi FreeIPA servis bio konfiguriran prema navedenim zahtjevima potrebno je servis instalirati na sljedeći način:

```
#instalacija FreeIPA paketa:
Yum module -y install idm:DL1/dns
Yum install freeipa-server -y
#postavljanje FreeIPA poslužitelja s DNS-om:
Echo -e „192.168.64.20 linux1.antoniojanach.local“ >> /etc/hosts
Ipa-server-install
#provjera Kerberos tiketa:
Kinit admin
klist
```



```
Activities Terminal May 16 16:43
root@linux1:~/Documents
File Edit View Search Terminal Help
Configured /etc/ssh/ssh_config
Configured /etc/ssh/ssh_config
Configuring antoniojanach.local as NIS domain.
Client configuration complete.
The ipa-client-install command was successful

unable to resolve host name linux1.antoniojanach.local. to IP address, ipa-ca DNS record will be incomplete
unable to resolve host name linux1.antoniojanach.local. to IP address, ipa-ca DNS record will be incomplete
Please add records in this file to your DNS system: /tmp/ipa.system.records.75c4nzcr.db
=====
Setup complete

Next steps:
  1. You must make sure these network ports are open:
      TCP Ports:
        * 80, 443: HTTP/HTTPS
        * 389, 636: LDAP/LDAPS
        * 88, 464: kerberos
      UDP Ports:
        * 88, 464: kerberos
        * 123: ntp

  2. You can now obtain a kerberos ticket using the command: 'kinit admin'
     This ticket will allow you to use the IPA tools (e.g., ipa user-add)
     and the web user interface.

Be sure to back up the CA certificates stored in /root/cacert.p12
These files are required to create replicas. The password for these
files is the Directory Manager password
The ipa-server-install command was successful
[root@linux1 Documents]# kinit admin
Password for admin@ANTONIOJANACH.LOCAL:
[root@linux1 Documents]# klist
Ticket cache: KCM:0
Default principal: admin@ANTONIOJANACH.LOCAL

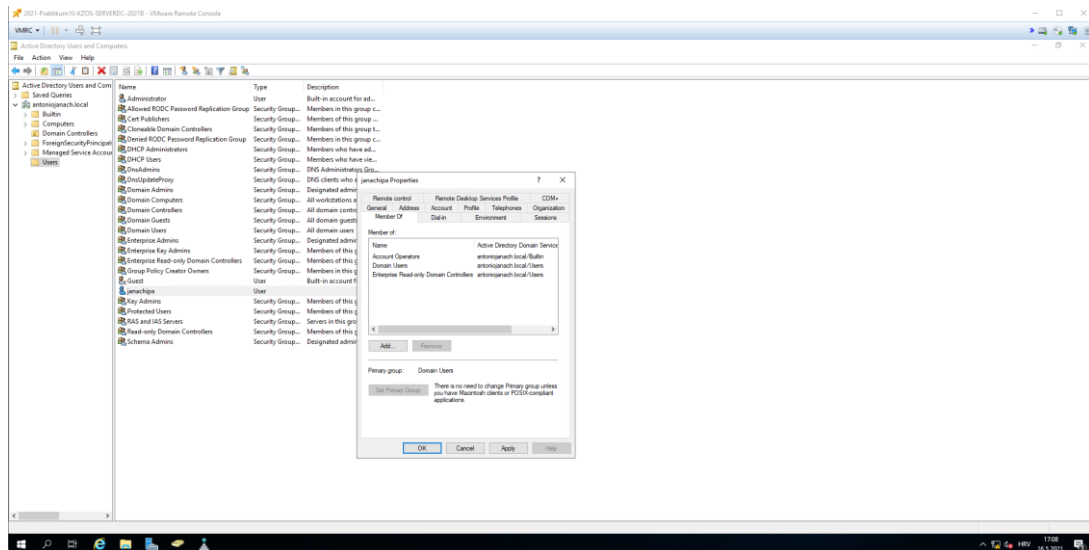
Valid starting Expires Service principal
05/16/2021 16:43:29 05/17/2021 16:43:27 krbtgt/ANTONIOJANACH.LOCAL@ANTONIOJANACH.LOCAL
[root@linux1 Documents]#
```

Slika 37: prikaz uspješne instalacije FreeIPA servisa

Kad je servis instaliran potrebno je kroz firewall propustiti određene portove:

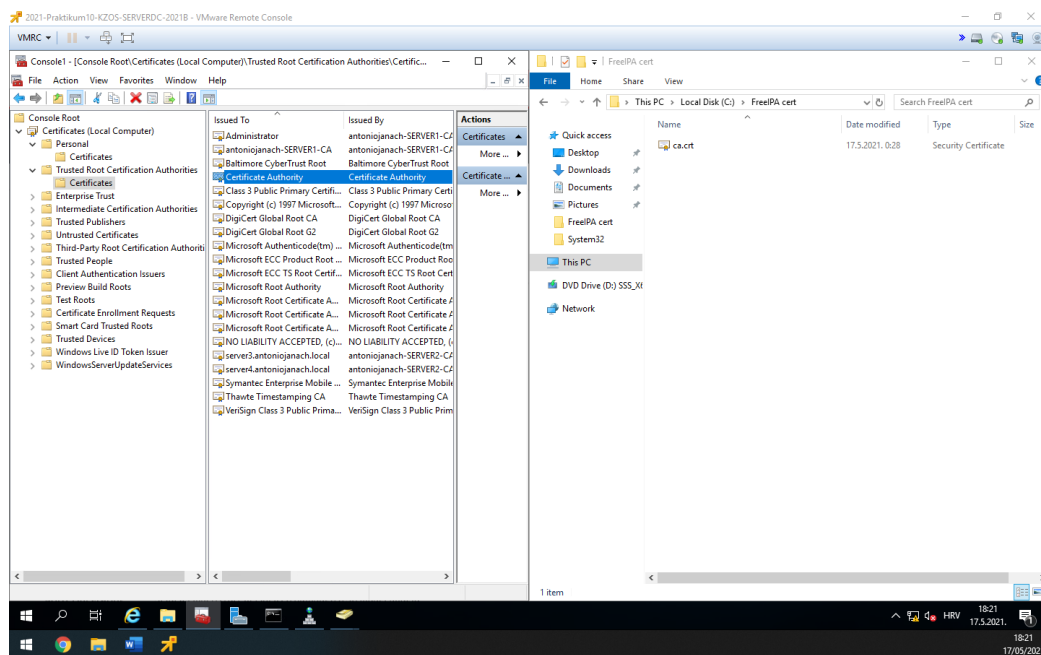
```
#propuštanje portova kroz firewall:
Firewall-cmd --add-service={freeipa-ldap,freeipa-ldaps,dns,ntp} --permanent
Firewall-cmd --reload
```

Da bi FreeIPA radila zajedno s Windows Active Directory imeničkim sustavom nužno je uspostaviti povjerenje između FreeIPA servisa i Windows AD-a. Ova uspostava povjerenja između Linuxa i Windowsa postiže se razmjenom certifikata tako tako da ServerDC i Linux1 imaju certifikat koji je domenski. Prije svega potrebno je na ServerDC računalu nad domenom kreirati novi korisnički račun „janachipa@antoniojanach.local“. Novo kreiranog korisnika potrebno je staviti u sljedeće grupe: „Enterprise Read-Only Domain Controller“ i „Account Operator“. Na ovaj način kreiranom korisniku koji je dodan u navedene grupe omogućen je pristup za replikaciju.



Slika 38: prikaz kreiranog korisnika s dodanim grupama u AD-u

Nakon što je korisnik kreiran i dodijeljena su mu prava potrebno je sa ServerDC poslužitelja preuzeti FreeIPA CA certifikat. Kako bi se preuzeo certifikat potrebno je pomoću web browsera na web lokaciji: <https://linux1.antoniojanach.local/ipa/ui/> preuzeti certifikat. Kad je certifikat preuzet, taj certifikat potrebno je instalirati u „Trusted Root Certification Authorities“. Nakon toga Server DC je spreman za trust. Certifikat nosi ime „Certificate Authority“ i certifikat vrijedi do 2041. godine.

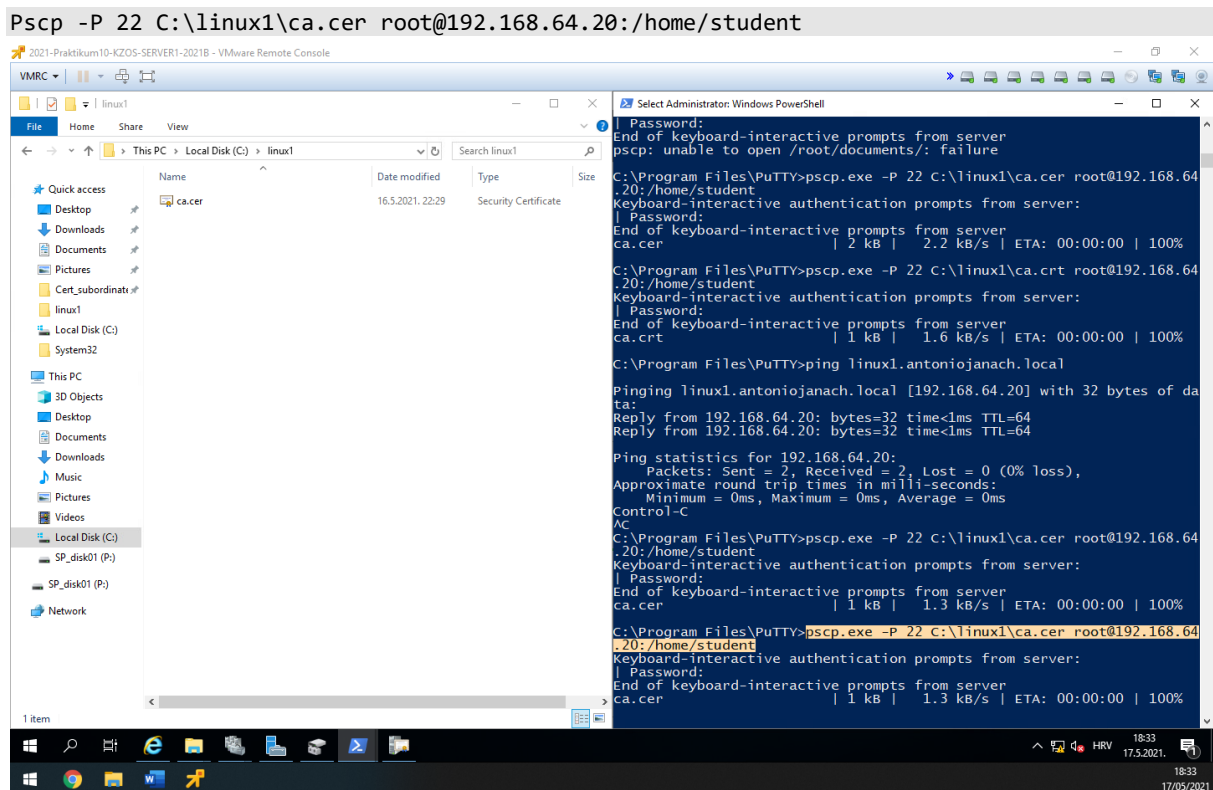


Slika 39 prikaz preuzetog i instaliranog FreeIPA certifikata na ServerDC poslužitelj

Kad je certifikat izgeneriran od FreeIPA servisa i kad je instaliran na ServerDC poslužitelju potrebno je pomoću već instalirane uloge imena „Certificate Authority“ preuzeti CA certifikat od *Active Directory-a*. Kad je certifikat preuzet potrebno je taj certifikat prenijeti na Linux1 poslužitelj. Kako bi se prenijela ta ca.cer datoteka korišten je alat imena „Putty“ i naredba pscp.

Napomena: certifikat je potrebno preuzeti u „Base 64“ metodi kodiranja. Kako bi se preuzeo certifikat korišteno je Web sučelje imena „Microsoft Active Directory Certificate Services“ na sljedećoj Web lokaciji: <https://server1/certsrv/>.

Komanda koja je korištena:



Slika 40: Preuzimanje CA certifikate od strane AD-a i prijenos istog na Linux1 poslužitelj

Prijenosom ca.cer datoteke s ServerDC poslužitelja na Linux1 poslužitelj, potrebno je certifikat dodati u datoteku putanje /etc/openldap/certs i urediti datoteku na putanji /etc/openldap/ldap.conf.

```

Mv /home/student/ca.cer /etc/openldap/certs
Vim /etc/openldap/ldap.conf
    SASL_NOCANON on
    URI ldaps://linux1.antoniojanach.local
    BASE dc=antoniojanach,dclocal
    TLS_CACERT /etc/ipa/ca.crt
    SASL_MECH GSSAPI
    TLS_CACERTDIR /etc/openldap/certs/
    TLS_REQCERT allow
  
```

Sad je sve spremno za pokretanje replikacije sljedećom komandom na Linux1 poslužitelju:

```

Ipa-replica-manage connect --winsync --
binddn='cn=janachipa,cn=users,dc=antoniojanach,dc=local' --bindpw='Pa$$w0rd' --
passsync='Pa$$w0rd' --cacert='/etc/openldap/certs/ca.cer' serverdc.antoniojanach.local -v

```

```

[root@linux1 openldap]# ipa-replica-manage connect --winsync --binddn='cn=free\ ipa,cn=users,dc=antoniojanach,dc=local' --bindpw='Pa$$w0rd' --passsync='Pa$$w0rd' --cacert='/etc/openldap/certs/ca.cer' serverdc.antoniojanach.local -v
Directory Manager password:

Added CA certificate /etc/openldap/certs/ca.cer to certificate database for linux1.antoniojanach.local
ipa: INFO: Failed to connect to AD server serverdc.antoniojanach.local
ipa: INFO: The error was: {'desc': 'Invalid credentials', 'info': '80090308: LdapErr: DSID-0C090446, comment: AcceptSecurityContext error, data 52e, v4563'}
Failed to setup winsync replication
[root@linux1 openldap]# ipa-replica-manage connect --winsync --binddn='cn=Users,dc=antoniojanach,dc=local' --bindpw='Pa$$w0rd' --passsync='Pa$$w0rd' --cacert='/etc/openldap/certs/ca.cer' serverdc.antoniojanach.local -v
Directory Manager password:

Added CA certificate /etc/openldap/certs/ca.cer to certificate database for linux1.antoniojanach.local
ipa: INFO: Failed to connect to AD server serverdc.antoniojanach.local
ipa: INFO: The error was: {'desc': 'Invalid credentials', 'info': '80090308: LdapErr: DSID-0C090446, comment: AcceptSecurityContext error, data 52e, v4563'}
Failed to setup winsync replication
[root@linux1 openldap]# ipa-replica-manage connect --winsync --binddn='cn=janachipa,cn=Users,dc=antoniojanach,dc=local' --bindpw='Pa$$w0rd' --passsync='Pa$$w0rd' --cacert='/etc/openldap/certs/ca.cer' serverdc.antoniojanach.local -v
Directory Manager password:

Added CA certificate /etc/openldap/certs/ca.cer to certificate database for linux1.antoniojanach.local
ipa: INFO: AD Suffix is: DC=antoniojanach,DC=local
The user for the Windows PassSync service is uid=passsync,cn=sysaccounts,cn=etc,dc=antoniojanach,dc=local
Adding Windows PassSync system account
ipa: INFO: Added new sync agreement, waiting for it to become ready . . .
ipa: INFO: Replication Update in progress: FALSE: status: Error (0) Replica acquired successfully: Incremental update started: start: 20210516223727: end: 20210516223727
ipa: INFO: Agreement is ready, starting replication . . .
ipa: WARNING: This configuration ("--winsync") may imply that the log file contains clear text passwords.
Please ensure that these files can be accessed only by trusted accounts.
Log files are under /var/lib/dirsrv/slapd-ANTONIOJANACH-LOCAL/cldb
Starting replication, please wait until this has completed.

Update succeeded

Connected 'linux1.antoniojanach.local' to 'serverdc.antoniojanach.local'
[root@linux1 openldap]# ipa-replica-manage connect --winsync --binddn='cn=ianachipa,cn=Users,dc=antoniojanach,dc=local' --bindpw='Pa$$w0rd' --passsync='Pa$$w0rd'

```

Slika 41: Prikaz uspješno postignute replikacije između Windows AD-a i FreeIPA servisa

Linux1 poslužitelj na kojem je instalirana FreeIPA je uspješno sinkroniziran s „Windows Active Directory“ imeničkim sustavom. Zaključno tome FreeIPA servis ima pristup „Windows Active Directory“ DNS zapisima.

6.15.2. Instalacija Samba servisa i postavljanje inkrementalnog backup-a

Prema zahtjevima potrebno je u direktoriju /domainshare napraviti SMB share na koji pristup imaju samo korisnici domene antoniojanach.local. Zatim je za taj kreirani SMB share potrebno napraviti backup skriptu koja će raditi inkrementalni backup svaki dan u 02:00:00, potom će taj backup slati na Linux2 poslužitelju direktorij /backup.

Za početak nužno je instalirati sve potrebne servise i kreirati direktorij na putanji /domainshare kojem je postavljen pristup da mogu tom direktoriju pristupiti i korisnici koji nisu dio Linux sustava.

```
Yum install samba samba-common samba-client -y
Mkdir /domainshare
Chmod -R 0755 /domainshare
Chown -R nobody:nobody /domainshare
Chcon -t samba_share_t /domainshare
```

Zatim je potrebno konfigurirati file na putanji /etc/samba/smb.conf

```
Vim /etc/samba/smb.conf
[global]
Workgroup = ANTONIOJANACH
Server string = Samba Server %v
Netbio name = linux
Security = user
Map to guest = bad user
dns proxy = no
[domainshare]
Path = /domainshare
Browsable = yes
Writable = yes
Guest ok = yes
Read only = no
```

Testparm

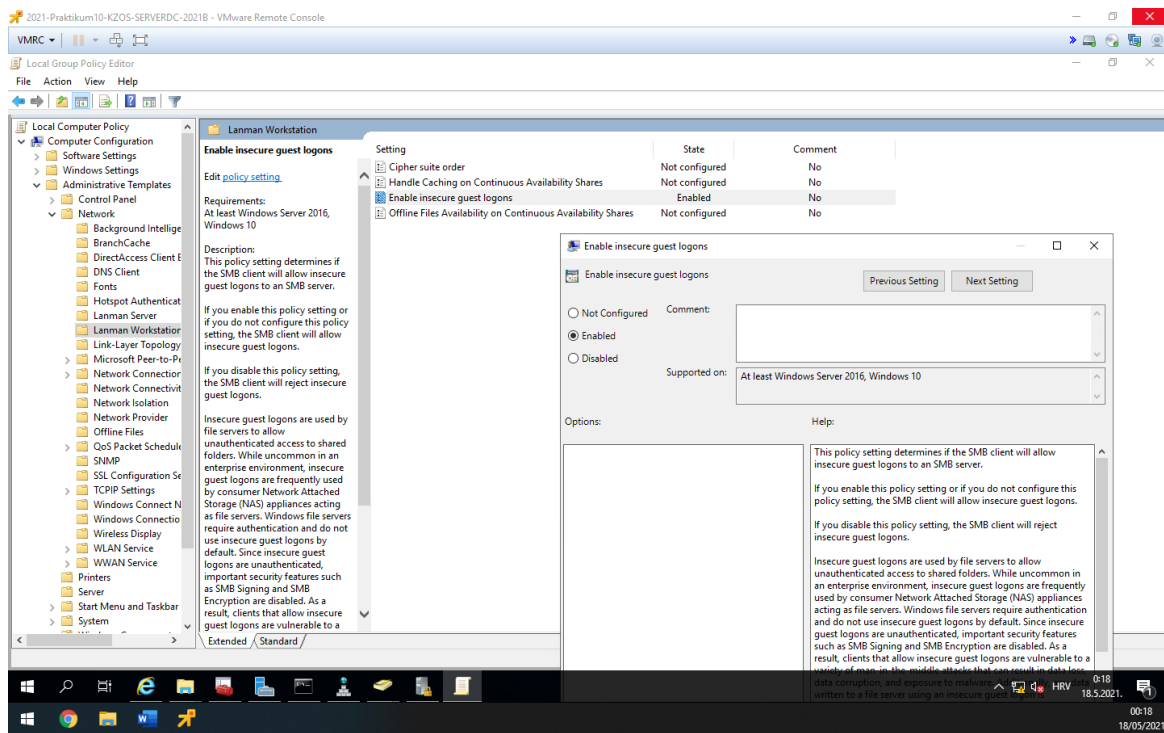
Kad je konfiguracija uspješno konfigurirana za smb.conf file nužno je propustiti Samba servis kroz firewall.

```
Firewall-cmd --add-service=samba --zone=public --permanent
Firewall-cmd --reload
```

Ovime završava konfiguracija Samba servisa na Linux 1 poslužitelju, kad je sve uspješno konfigurirano potrebno je pokrenuti servis.

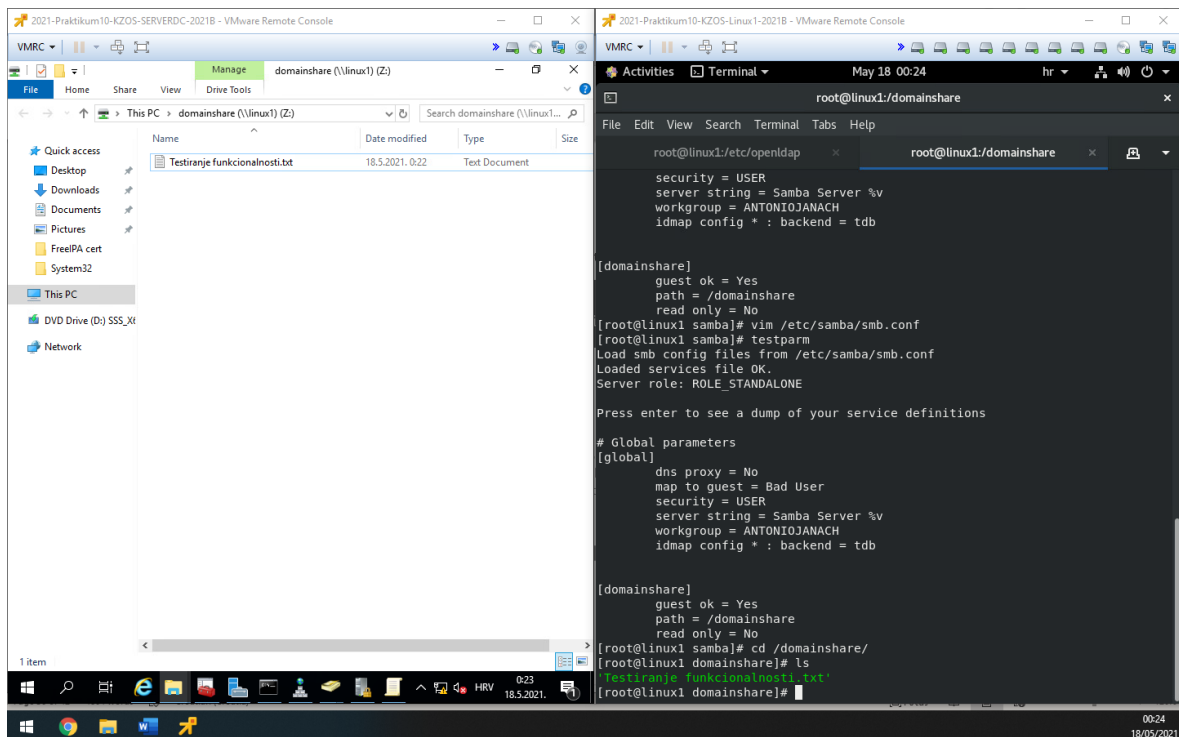
```
Systemctl start smb
Systemctl start nmb
Systemctl enable smb
Systemctl enable nmb
```

Prije nego se isproba funkcionalnost, mapiranje diska neće biti moguće jer je potrebno konfigurirati „Group Policy“. U „Group Policy“ potrebno je omogućiti „Enable insecure guest logons“.



Slika 42: konfiguracija "Group Policy" na ServerDC poslužitelju

Kad je „Group Policy“ postavljen, sada je moguće mapirati disk na računalo koristeći putanju `\\linux1\domainshare`. Ujedno je testirana funkcionalnost tako što je kreiran .txt file na Windows strani te se je taj kreirani .txt file prenio u folder /domainshare na Linux strani.



Slika 43: prikaz uspješne konfiguracije SMB share-a

Kad je SMB share postavljen i disk se može mapirati na Windows poslužitelje, taj isti je potrebno backupirati koristeći skriptu koja će raditi inkrementalni backup svaki dan u 02:00:00 i slati ga na Linux2 u direktorij /backup.

```
#!/bin/bash
# Configuration variables (change as you wish)
src="${1:-/domainshare}"
dst="${2:-/backup}"
remote="${3:-linux2.antoniojanach.local}"
backupDepth=${backupDepth:-7}
timeout=${timeout:-1800}
pathBak0="${pathBak0:-data}"
partialFolderName="${partialFolderName:-rsync-partial}"
rotationLockFileName="${rotationLockFileName:-rsync-rotation-lock}"
pathBakN="${pathBakN:-backup}"
nameBakN="${nameBakN:-backup}"
exclusionFileName="${exclusionFileName:-exclude.txt}"
dateCmd="${dateCmd:-date}"
logName="${logName:-rsync-incremental-backup_$(dateCmd +%Y-%m-%d)_$(dateCmd +%H-%M-%S).log}"
ownFolderName="${ownFolderName:-rsync-incremental-backup}"
logFolderName="${logFolderName:-log}"
interactiveMode="${interactiveMode:-no}"
# Combinate previously defined variables for use (don't touch this)
ownFolderPath="${HOME}/${ownFolderName}"
tempLogPath="${ownFolderPath}/${remote}_${dst//[\/]/}"
exclusionFilePath="${ownFolderPath}/${exclusionFileName}"
remoteDst="${remote}:${dst}"
bak0="${dst}/${pathBak0}"
remoteBak0="${remoteDst}/${pathBak0}"
partialFolderPath="${dst}/${partialFolderName}"
rotationLockFilePath="${dst}/${rotationLockFileName}"
logPath="${dst}/${pathBakN}/${logFolderName}"
remoteLogPath="${remote}:${logPath}"
logFile="${tempLogPath}/${logName}"
# Prepare own folder
mkdir -p "${tempLogPath}"
touch "${logFile}"
touch "${exclusionFilePath}"
writeToLog() {
    echo -e "${1}" | tee -a "${logFile}"
}
writeToLog "*****"
writeToLog "*"
writeToLog "*   rsync-incremental-backup   *"
writeToLog "*"
writeToLog "*****"
# Prepare backup paths
i=1
while [ "${i}" -le "${backupDepth}" ]
do
    export "bak${i}=${dst}/${pathBakN}/${nameBakN}.${i}"
```



```

        true "${(i = i + 1)}"
done
writeToLog "\\n[${dateCmd} -Is] You are going to backup"
writeToLog "\\tfrom:  ${src}"
writeToLog "\\tto:    ${remoteBak0}"
# Prepare ssh parameters for socket connection, reused by following sessions
sshParams=(-o "ControlPath=\"${ownFolderPath}/ssh_connection_socket_%h_%p_%r\" " -o
"ControlMaster=auto" \
-o "ControlPersist=10")
# Prepare rsync transport shell with ssh parameters (escape for proper space handling)
rsyncShellParams=(-e "ssh$(for i in "${sshParams[@]}"; do echo -n " '${i}'"; done)")
batchMode="yes"
if [ "${interactiveMode}" = "yes" ]
then
    batchMode="no"
fi
# Check remote connection and create master socket connection
if ! ssh "${sshParams[@]}" -q -o BatchMode="${batchMode}" -o ConnectTimeout=10 "${remote}"
exit
then
    writeToLog "\\n[${dateCmd} -Is] Remote destination is not reachable"
    exit 1
fi
# Prepare paths at destination
ssh "${sshParams[@]}" "${remote}" "mkdir -p ${dst} ${logPath}"
writeToLog "\\n[${dateCmd} -Is] Old logs sending begins\\n"
# Send old pending logs to destination
rsync "${rsyncShellParams[@]}" -rhvz --remove-source-files --exclude="${logName}" --log-
file="${logFile}" \
    "${tempLogPath}/" "${remoteLogPath}/"
writeToLog "\\n[${dateCmd} -Is] Old logs sending finished"
# Rotate backups if last rsync succeeded ..
if (ssh "${sshParams[@]}" "${remote}" "[ ! -d ${partialFolderPath} ] && [ ! -e
${rotationLockFilePath} ]")
then
    # .. and there is previous data
    if (ssh "${sshParams[@]}" "${remote}" "[ -d ${bak0} ]")
    then
        writeToLog "\\n[${dateCmd} -Is] Backups rotation begins"
        true "${(i = i - 1)}"
        # Remove the oldest backup if exists
        bak="bak${i}"
        ssh "${sshParams[@]}" "${remote}" "rm -rf ${!bak}"
        # Rotate the previous backups
        while [ "${i}" -gt 0 ]
        do
            bakNewPath="bak${i}"
            true "${(i = i - 1)}"
            bakOldPath="bak${i}"
            if (ssh "${sshParams[@]}" "${remote}" "[ -d ${!bakOldPath} ]")
            then

```

```

        ssh "${sshParams[@]}" "${remote}" "mv ${!bakOldPath}
${!bakNewPath}"
        fi
    done
    writeToLog "[$(${dateCmd} -Is)] Backups rotation finished\\n"
else
    writeToLog "\\n[$(${dateCmd} -Is)] No previous data found, there is no
backups to be rotated\\n"
fi
else
    writeToLog "\\n[$(${dateCmd} -Is)] Last backup failed, backups will not be
rotated\\n"
fi

# Set rotation lock file to detect in next run when backup fails
ssh "${sshParams[@]}" "${remote}" "touch ${rotationLockFilePath}"
writeToLog "[$(${dateCmd} -Is)] Backup begins\\n"
# Do the backup
if rsync "${rsyncShellParams[@]}" -achvz --progress --timeout="${timeout}" --delete --no-W
\
    --partial-dir="${partialFolderName}" --link-dest="${bak1}/" --log-file="${logFile}"
--exclude="${ownFolderPath}" \
    --chmod=+r --exclude-from="${exclusionFilePath}" "${src}/" "${remoteBak0}/"
then
    writeToLog "\\n[$(${dateCmd} -Is)] Backup completed successfully\\n"
    # Clear unneeded partials and lock file
    ssh "${sshParams[@]}" "${remote}" "rm -rf ${partialFolderPath}
${rotationLockFilePath}"
    rsyncFail=0
else
    writeToLog "\\n[$(${dateCmd} -Is)] Backup failed, try again later\\n"
    rsyncFail=1
fi

# Send the complete log file to destination
if scp "${sshParams[@]}" "${logFile}" "${remoteLogPath}"
then
    rm "${logFile}"
fi

# Close master socket connection quietly
ssh "${sshParams[@]}" -q -O exit "${remote}"
exit "${rsyncFail}"

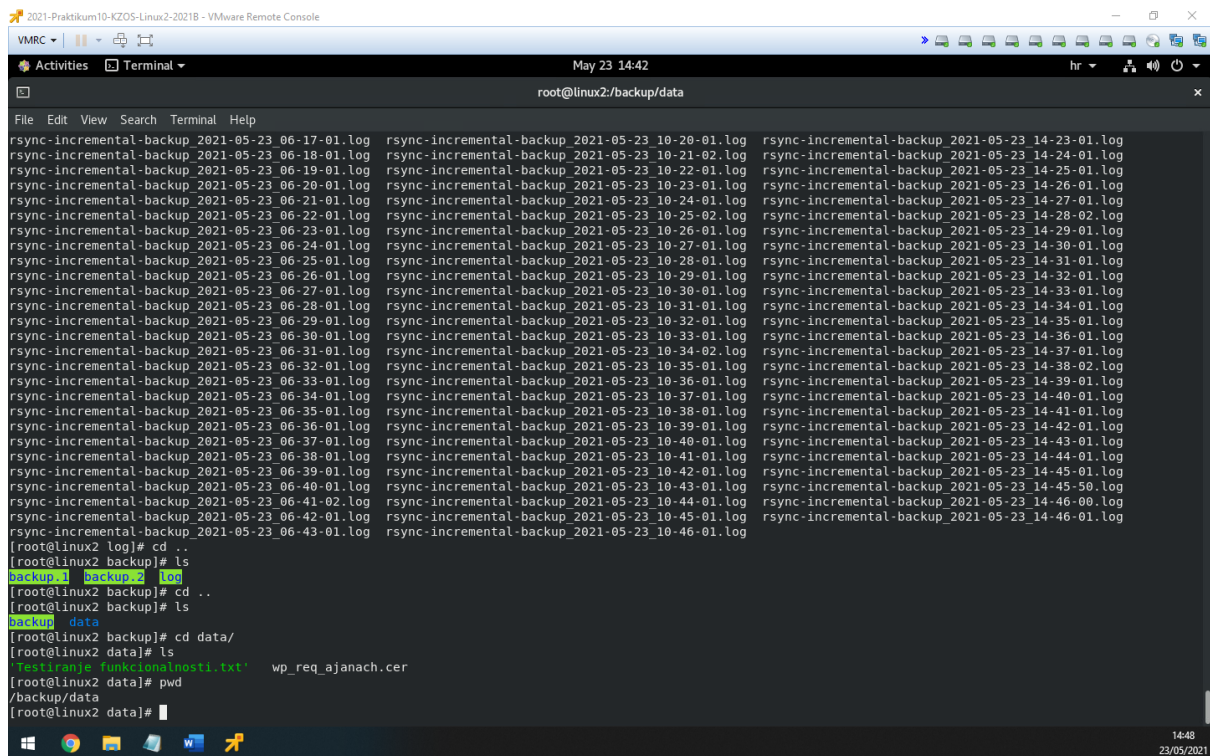
```

Skriptu je potrebno podesiti da se pokreće svaki dan u 02:00:00. To se postiže konfiguracijom crontab-a.

```

Crontab -e
0 2 * * * /backupSkripta/backupSkripta.sh

```



Slika 44: prikaz uspješnog postavljenog inkrementalnog backup-a na SMB share folder

Ovime je uspješno postavljen SMB share na koji se mogu spojiti svi članovi „antoniojanach.local“ domene zajedno sa izvršavanjem inkrementalnog backup-a pomoću skripte i crontab job-a svaki dan u 02:00:00 sati.

6.15.3. Wordpress instalacija na Linux1 poslužitelju

Na Linux1 poslužitelj podignut je LAMP server i u njega je instalirana zadnja verzija WordPress-a. Za instalaciju WordPress-a preduvjet je instalirati php, mariadb, httpd servise i sukladno tome ih konfigurirati na ispravan način. Također kroz firewall propušten je http i https promet. WordPress treba je serviran iz direktorija /wordpress1.

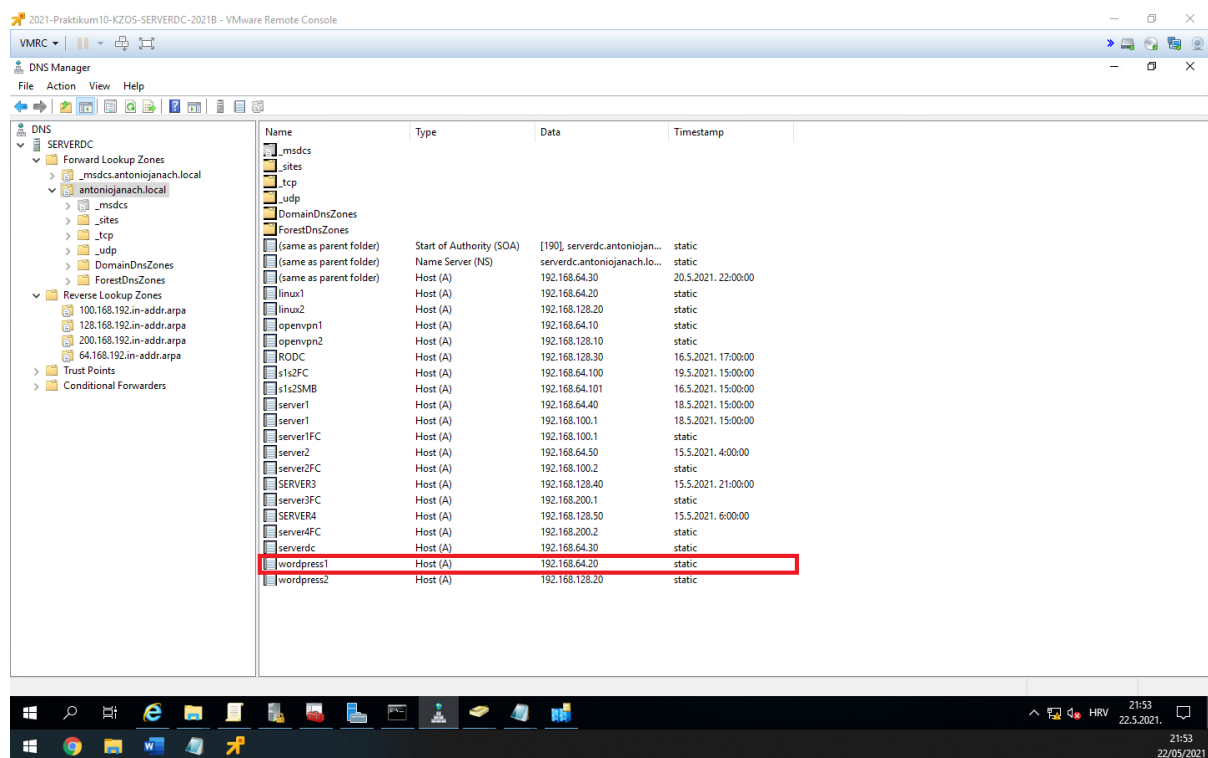
```
#instalacija potrebnih servisa i postavljene postavke da se koristi php verzije 7.4:
yum install mariadb mariadb-server httpd httpd-tools php php-cli php-json php-gd php-
mbstring php-pdo php-xml php-mysqlnd php-pecl-zip wget -y
yum module reset php
yum module enable php:7.4
yum install php php-cli php-json php-gd php-mbstring php-pdo php-xml php-mysqlnd php-pecl-
zip -y
#pokretanje servisa i omogućavanje servisa da se pokreću zajedno sa sustavom:
systemctl start httpd
systemctl enable httpd
systemctl start mariadb
systemctl enable mariadb
#propuštanje http i https prometa kroz firewall:
firewall-cmd --zone-public --add-service={http,https} --permanent
firewall-cmd --reload
#kreiranje direktorija i dodijela prava nad direktorij:
mkdir /wordpress1
chmod -R 775 /wordpress1
chown -R apache:apache /wordpress1
#konfiguracija mysql baze podataka:
mysql_secure_installation
mysql -u root -p
MariaDB [(none)]> CREATE DATABASE wordpress1;
MariaDB [(none)]> CREATE USER `wordpress1`@`localhost` IDENTIFIED BY 'centos';
MariaDB [(none)]> GRANT ALL ON wordpress1.* TO `wordpress1`@`localhost`;
MariaDB [(none)]> FLUSH PRIVILEGES;
MariaDB [(none)]> exit
#preuzimanje najnovije verzije WordPress-a:
Cd /wordpress1
Wget https://wordpress.org/latest.tar.gz
Tar xvzf latest.tar.gz
#Selinux konfiguracija:
Chmod -Rf 775 /wordpress1
Chown -Rf apache:apache /wordpress1
Semanage fcontext -a -t httpd_sys_rw_content_t
    „/wordpress1(/.*)"?"
Restorecon -Rv „/wordpress1“
```

Kako bi WordPress bio dostupan samo kroz HTTPS/TLS, generiran je request za potpisivanje certifikata od strane subordinate CA.

```
Openssl req -new -newkey rsa:2048 -nodes -keyout wp_cert_ajanych.key -out wp_req_ajanych.cer
```

Novo izgenerirani request potrebno je potpisati od strane Server2 poslužitelja koji ima instaliranu „Subordinate CA“ ulogu. Novo generirani request s certifikatom koji je potrebno potpisati je prebačen na Server2 poslužitelj pomoću predhodno kreiranog SMB share-a. Kad je datoteka certifikata prebačena na Server2 poslužitelj potrebno je taj certifikat potpisati tako da se otvori Web preglednik na sljedećoj Web lokaciji: <https://server2.antoniojanach.local/certsrv>. Potpisani certifikat potrebno je preuzeti u „base-64-encoded“ formatu kako bi ga Linux1 poslužitelj znao pročitati.

Potrebno je dodati DNS zapis na ServerDC poslužitelju za Web lokacija na kojem će se učitavati i administrirati WordPress stranica, a web lokacija glasi: <https://wordpress1.antoniojanach.local:10443>.



Slika 45: DNS zapis za Web lokaciju

Sad je potrebno stvoriti novu apache konfiguraciju na putanji /etc/httpd/conf.d/wordpress.conf na sljedeći način:

```
Vim /etc/httpd/conf.d/wordpress.conf
    Listen 10443 https
<VirtualHost *:80>
    ServerAdmin root@localhost
    ServerName wordpress1.antoniojanach.local
    Redirect permanent / https://wordpress1.antoniojanach.local:10443/
</VirtualHost>
<VirtualHost 192.168.64.20:10443>
    ServerAdmin root@localhost
    DocumentRoot /wordpress1/wordpress
    ServerName wordpress1.antoniojanach.local
    #ServerAlias linux1.antoniojanach.local
<Directory /wordpress1/wordpress>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>
    ErrorLog /var/log/httpd/wordpress_error.log
    CustomLog /var/log/httpd/wordpress_access.log common
    SSLEngine on
    SSLCertificateFile /certifikati/certnew.cer
    SSLCertificateKeyFile /certifikati/wp_cert_ajanach.key
    SSLCertificateChainFile /etc/openldap/certs/ca.cer
</VirtualHost>
#kad je konfiguracija spremljena potrebno je ponovno pokrenuti httpd servis:
Systemctl restart httpd
```

Nakon ponovnog pokretanja httpd servisa pojavit će se razni errori i zato je potrebno instalirati sealert da bi se navedeni „Aleart“ riješio. To se događa jer je SELinux u „Enforcing“ stanju.

```
Yum install setroubleshoot setools -y
#pokrenuti komandu sealert kako bi se riješili svi alearti tako što sealert alat predloži
što treba učiniti:
sealert
```

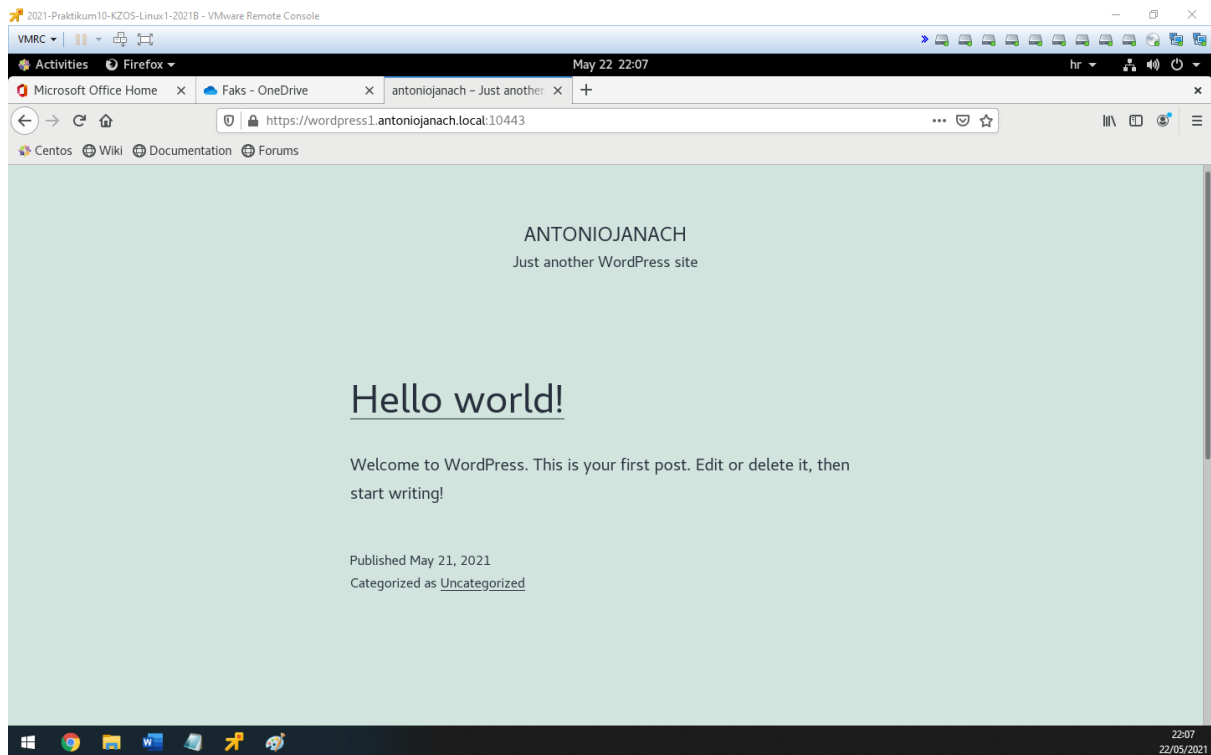
Zatim propustiti port 10443 kroz firewall, ovaj port je korišten zbog konfiguracije „reverse proxying/load balancer“ https prometa na L1 i L2 mašinama.

```
Firewall-cmd --add-service:10443/tcp --permanent
Firewall-cmd --reload
```

Ovime završava inicijalna konfiguracija te preostaje instalacija WordPress-a preko GUI-a pomoću Web preglednika na Web lokaciji: <https://linux1.antoniojanach.local/wordpress>. Certifikat za https je potpisan od strane subordinate CA te je moguće pristupiti https protokolom WordPress stanici.

Zadnja stvar koju je potrebno napraviti, a to je kopirati ca.cer na lokaciju /etc/pki/ca-trust/source/anchors/:

```
Cp ca.cer /etc/pki/ca-trust/source/anchors
update-ca-trust extract
systemctl restart httpd
```

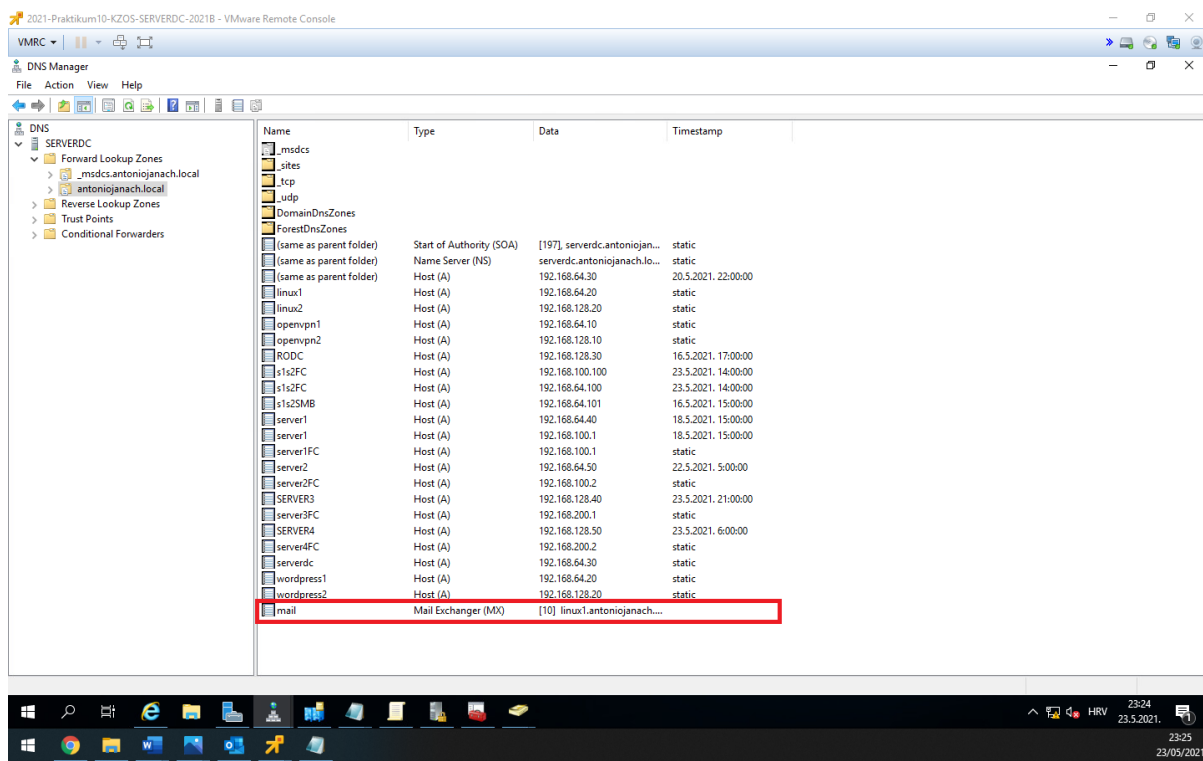


Slika 46: prikaz uspješne instalacije WordPress-a na Linux1 poslužitelj

Nakon uspješne instalacije WordPress-a na Linux1 moguće je pristupiti Web lokaciji koristeći https protokol.

6.15.4. Konfiguracija mail servera na Linux 1 poslužitelj

Cilj je podignuti na Linu1 poslužitelj mail server koji će biti MX za domenu antoniojanach.local. Stoga je prvo potrebno dodati MX zapis u DNS servis koji se nalazi na ServerDC poslužitelju.



Slika 47: MX zapis je dodan u DNS server

Na Linux1 poslužitelju instaliran je postfix servis koji je pokrenut i pokreće se svaki put kad se sustav ponovno pokrene. Zatim je instaliran mailx email klijent i uređena je postfix konfiguracijska datoteka na putanji: /etc/postfix/main.cf.

```
Yum install postfix -y
Systemctl start postfix
Systemctl enable postfix
Yum install mailx -y
Postavljanje postfix konfiguracijske datoteke:
Vim /etc/postfix/main.cf
    compatibility_level = 2
    queue_directory = /var/spool/postfix
    command_directory = /usr/sbin
    daemon_directory = /usr/libexec/postfix
    data_directory = /var/lib/postfix
    mail_owner = postfix
    myhostname = mail.antoniojanach.local
    mydomain = antoniojanach.local
    myorigin = $mydomain
    inet_interfaces = all
    inet_protocols = ipv4
    mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
    unknown_local_recipient_reject_code = 550
```



```

mynetworks = 192.168.64.20, 192.168.64.10, 192.168.128.20, 192.168.128.10,
127.0.0.0/8
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
home_mailbox = Maildir/
debug_peer_level = 2
debugger_command =
    PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
    ddd $daemon_directory/$process_name $process_id & sleep 5
sendmail_path = /usr/sbin/sendmail.postfix
newaliases_path = /usr/bin/newaliases.postfix
mailq_path = /usr/bin/mailq.postfix
setgid_group = postdrop
html_directory = no
manpage_directory = /usr/share/man
sample_directory = /usr/share/doc/postfix/samples
readme_directory = /usr/share/doc/postfix/README_FILES
smtpd_tls_cert_file = /etc/pki/tls/certs/postfix.pem
smtpd_tls_key_file = /etc/pki/tls/private/postfix.key
smtpd_tls_security_level = may
smtp_tls_CApath = /etc/pki/tls/certs
smtp_tls_CAfile = /etc/pki/tls/certs/ca-bundle.crt
smtp_tls_security_level = may
meta_directory = /etc/postfix
shlib_directory = /usr/lib64/postfix

```

Sljedeće što je potrebno, a to je restartati postfix servis i propustiti smtp protokol na firewall-u.

```

Systemctl restart postfix
Firewall-cmd --add-service=smtp --permanent
Firewall-cmd --reload

```

Pokrenuti sealeart komandu za pristup konfiguriranju Maildir direktorija u koji dolaze mailov-i na korisnikov profil.

```

ausearch -c 'local' --raw | audit2allow -M my-local
semodule -X 300 -i my-local.pp

```

Testirati slanje mail-a s root korisnika na student korisnika.

```

Mail -s „test“ student@linux1.antoniojanach.local
Test.
Ctrl + D
EOT

```

```

2021-Praktikum10-KZOS-Linux1-2021B - VMware Remote Console
VMRC | May 24 00:01
student@linux1:~/Maildir/new

File Edit View Search Terminal Help

--A3CB918F2FC7.1621804295@mail.antoniojanach.local
Content-Description: Delivery report
Content-Type: message/delivery-status

Reporting-MTA: dns; mail.antoniojanach.local
X-Postfix-Queue-ID: A3CB918F2FC7
X-Postfix-Sender: rfc822; student@antoniojanach.local
Arrival-Date: Sun, 23 May 2021 23:11:35 +0200 (CEST)

Final-Recipient: rfc822; root@linux1.antoniojanach.local
Original-Recipient: rfc822;root@linux1.antoniojanach.local
Action: failed
Status: 5.4.4
Diagnostic-Code: X-Postfix; Host or domain name not found. Name service error
for name=linux1.antoniojanach.local type=A: Host not found

--A3CB918F2FC7.1621804295@mail.antoniojanach.local
Content-Description: Undelivered Message
Content-Type: message/rfc822
Content-Transfer-Encoding: 8bit

Return-Path: <student@antoniojanach.local>
Received: by mail.antoniojanach.local (Postfix, from userid 1000)
        id A3CB918F2FC7; Sun, 23 May 2021 23:11:35 +0200 (CEST)
Date: Sun, 23 May 2021 23:11:35 +0200
To: root@linux1.antoniojanach.local
Subject: test
User-Agent: Heirloom mailx 12.5 7/5/10
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Message-Id: <20210523211135.A3CB918F2FC7@mail.antoniojanach.local>
From: student <student@antoniojanach.local>

test.

--A3CB918F2FC7.1621804295@mail.antoniojanach.local--
[student@linux1 new]$
00:01
24/05/2021

```

Slika 48: prikaz uspješno poslanog maila prema korisniku student

```

2021-Praktikum10-KZOS-Linux1-2021B - VMware Remote Console
VMRC | May 24 00:03
student@linux1:~/Maildir/new

File Edit View Search Terminal Help

Reporting-MTA: dns; mail.antoniojanach.local
X-Postfix-Queue-ID: A3CB918F2FC7
X-Postfix-Sender: rfc822; student@antoniojanach.local
Arrival-Date: Sun, 23 May 2021 23:11:35 +0200 (CEST)

Final-Recipient: rfc822; root@linux1.antoniojanach.local
Original-Recipient: rfc822;root@linux1.antoniojanach.local
Action: failed
Status: 5.4.4
Diagnostic-Code: X-Postfix; Host or domain name not found. Name service error
for name=linux1.antoniojanach.local type=A: Host not found

--A3CB918F2FC7.1621804295@mail.antoniojanach.local
Content-Description: Undelivered Message
Content-Type: message/rfc822
Content-Transfer-Encoding: 8bit

Return-Path: <student@antoniojanach.local>
Received: by mail.antoniojanach.local (Postfix, from userid 1000)
        id A3CB918F2FC7; Sun, 23 May 2021 23:11:35 +0200 (CEST)
Date: Sun, 23 May 2021 23:11:35 +0200
To: root@linux1.antoniojanach.local
Subject: test
User-Agent: Heirloom mailx 12.5 7/5/10
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Message-Id: <20210523211135.A3CB918F2FC7@mail.antoniojanach.local>
From: student <student@antoniojanach.local>

test.

--A3CB918F2FC7.1621804295@mail.antoniojanach.local--
[student@linux1 new]$ mail -s "test" janach.antonio@gmail.com
Test.
EOT
[student@linux1 new]$
00:03
24/05/2021

```

Slika 49: prikaz uspješno poslanog maila van domene

Ovime završava konfiguracija mail servera na Linux1 poslužitelju.

6.16. Konfiguracija CentOS poslužitelja - lokacija B

6.16.1. Wordpress instalacija na Linux2 poslužitelju

Instalaciju WordPress-a je na Linux2 poslužitelj potrebno provesti kao i na Linux1 poslužitelju. Jedina razlika u konfiguraciji su promjene naziva u bazi podataka, korisničkome računu i nazivu direktorija.

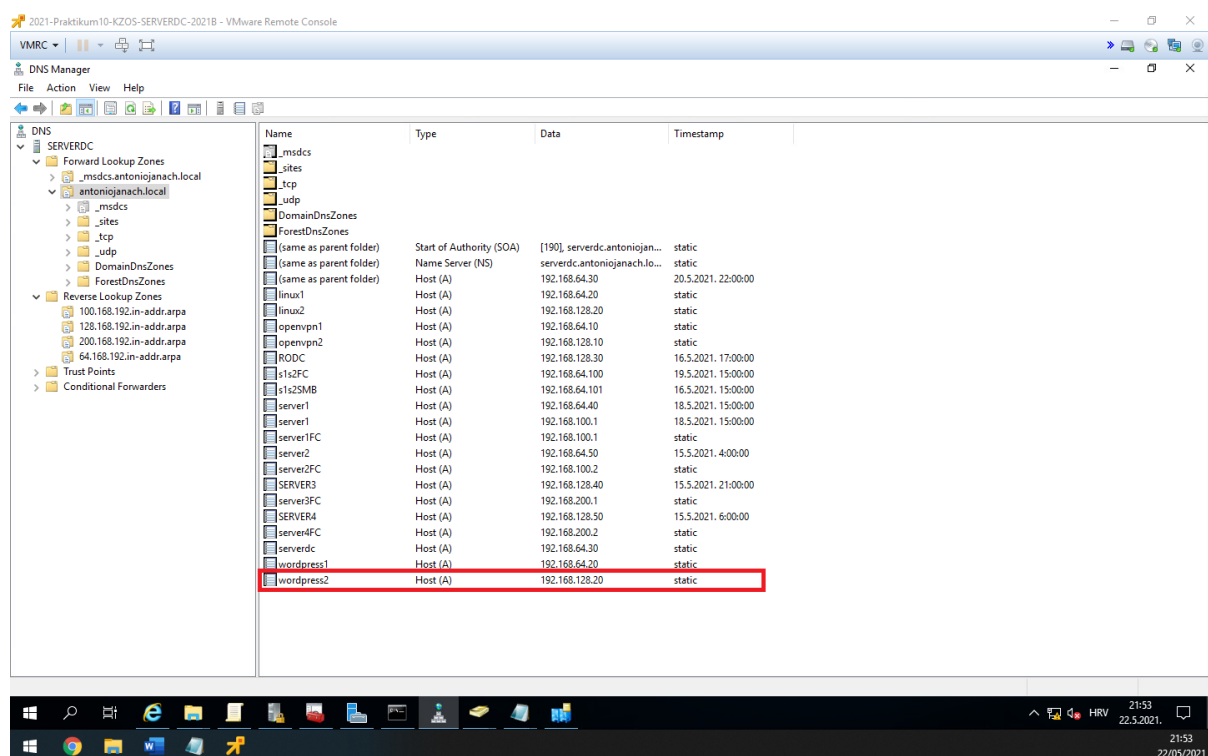
```
#instalacija potrebnih servisa i postavljene postavke da se koristi php verzije 7.4:
yum install mariadb mariadb-server httpd httpd-tools php php-cli php-json php-gd php-
mbstring php-pdo php-xml php-mysqlnd php-pecl-zip wget -y
yum module reset php
yum module enable php:7.4
yum install php php-cli php-json php-gd php-mbstring php-pdo php-xml php-mysqlnd php-pecl-
zip -y
#pokretanje servisa i omogućavanje servisa da se pokreću zajedno sa sustavom:
systemctl start httpd
systemctl enable httpd
systemctl start mariadb
systemctl enable mariadb
#propuštanje http i https prometa kroz firewall:
firewall-cmd --zone-public --add-service={http,https} --permanent
firewall-cmd --reload
#kreiranje direktorija i dodijela prava nad direktorij:
mkdir /wordpress2
chmod -R 775 /wordpress2
chown -R apache:apache /wordpress2
#konfiguracija mysql baze podataka:
mysql_secure_installation
mysql -u root -p
MariaDB [(none)]> CREATE DATABASE wordpress2;
MariaDB [(none)]> CREATE USER `wordpress2`@`localhost` IDENTIFIED BY 'centos';
MariaDB [(none)]> GRANT ALL ON wordpress2.* TO `wordpress2`@`localhost`;
MariaDB [(none)]> FLUSH PRIVILEGES;
MariaDB [(none)]> exit
#preuzimanje najnovije verzije WordPress-a:
Cd /wordpress2
Wget https://wordpress.org/latest.tar.gz
Tar xvzf latest.tar.gz
#Selinux konfiguracija:
Chmod -Rf 775 /wordpress2
Chown -Rf apache:apache /wordpress2
Semanage fcontext -a -t httpd_sys_rw_content_t
„/wordpress2(/.*)?“
Restorecon -Rv „/wordpress2“
```

Kako bi WordPress bio dostupan samo kroz HTTPS/TLS, generiran je request za potpisivanje certifikata od strane subordinate CA.

```
openssl req -new -newkey rsa:2048 -nodes -keyout wp_cert_ajanach1.key -out wp_req_ajanach1.cer
```

Novo izgenerirani request potrebno je potpisati od strane Server2 poslužitelja koji ima instaliranu „Subordinate CA“ ulogu. Novo generirani request s certifikatom koji je potrebno potpisati je prebačen na Server2 poslužitelj pomoću Putty cmdlet-a. Kad je datoteka certifikata prebačena na Server2 poslužitelj potrebno je taj certifikat potpisati tako da se otvori Web preglednik na sljedećoj Web lokaciji: <https://server2.antoniojanach.local/certsrv>. Potpisani certifikati potrebno je preuzeti u „base-64-encoded“ formatu kako bi ga Linux1 poslužitelj znao pročitati.

Potrebno je dodati DNS zapis na ServerDC poslužitelju za Web lokacija na kojem će se učítavati i administrirati WordPress stranica, a web lokacija glasi: <https://wordpress2.antoniojanach.local:10443>.



Slika 50: DNS zapis za Web lokaciju

Sad je potrebno stvoriti novu apache konfiguraciju na putanji /etc/httpd/conf.d/wordpress.conf na sljedeći način:

```
Vim /etc/httpd/conf.d/wordpress.conf
    Listen 10443 https
<VirtualHost *:80>
    ServerAdmin root@localhost
    ServerName wordpress2.antoniojanach.local
    Redirect permanent / https://wordpress2.antoniojanach.local:10443/
</VirtualHost>
<VirtualHost 192.168.128.20:10443>
    ServerAdmin root@localhost
    DocumentRoot /wordpress2/wordpress
    ServerName wordpress2.antoniojanach.local
    #ServerAlias linux2.antoniojanach.local
<Directory /wordpress2/wordpress>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>
    ErrorLog /var/log/httpd/wordpress_error.log
    CustomLog /var/log/httpd/wordpress_access.log common
    SSLEngine on
    SSLCertificateFile /certifikati/certnew.cer
    SSLCertificateKeyFile /certifikati/wp_cert_ajanach1.key
    SSLCertificateChainFile /etc/openldap/certs/ca.cer
</VirtualHost>
#kad je konfiguracija spremljena potrebno je ponovno pokrenuti httpd servis:
Systemctl restart httpd
```

Nakon ponovnog pokretanja httpd servisa pojavit će se razni errori i zato je potrebno instalirati SEALert da bi se navedeni „Aleart“ riješio. To se događa jer je SELinux u „Enforcing“ stanju.

```
Yum install setroubleshoot setools -y
#pokrenuti komandu sealert kako bi se riješili svi alearti tako što sealert alat predloži
što treba učiniti:
Sealert
```

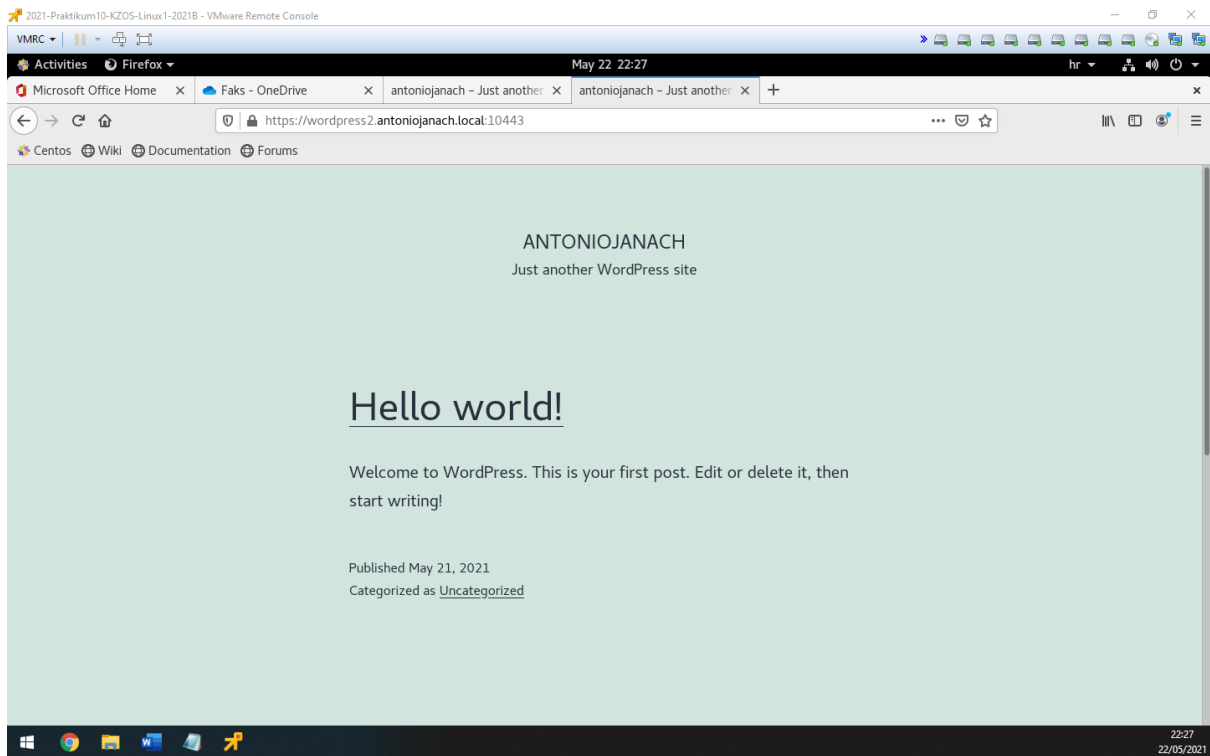
Zatim propustiti port 10443 kroz firewall, ovaj port je korišten zbog konfiguracije „reverse proxying/load balancer“ https prometa na L1 i L2 mašinama.

```
Firewall-cmd --add-service:10443/tcp --permanent
Firewall-cmd --reload
```

Ovime završava inicijalna konfiguracija te preostaje instalacija WordPress-a preko GUI-a pomoću Web preglednika na Web lokaciji: <https://linux2.antoniojanach.local/wordpress>. Certifikat za https je potpisan od strane subordinate CA te je moguće pristupiti https protokolom WordPress stanici.

Zadnja stvar koju je potrebno napraviti, a to je kopirati ca.cer na lokaciju /etc/pki/ca-trust/source/anchors/:

```
Cp ca.cer /etc/pki/ca-trust/source/anchors
update-ca-trust extract
systemctl restart httpd
```



Slika 51: prikaz uspješne instalacije WordPress-a na Linux1 poslužitelj

Nakon uspješne instalacije WordPress-a na Linux2 moguće je pristupiti Web lokaciji koristeći https protokol.

6.17. Slanje log zapisa s Linux2 na Linux1 poslužitelj

Koristeći bash skriptni jezik kreirana je skripta koja pomoću TCP protokola šalje svoje log zapise na lokalnog poslužitelj (Linux2) na Linux1 poslužitelj te ih Linux1 poslužitelj treba rotirati svaka 2 dana koristeći kompresiju. Pomoću crontab-a postavljeno je da se scp naredbom šalju logovi na Linux1 poslužitelj u svakoj minuti nakon svakog 2. sata.

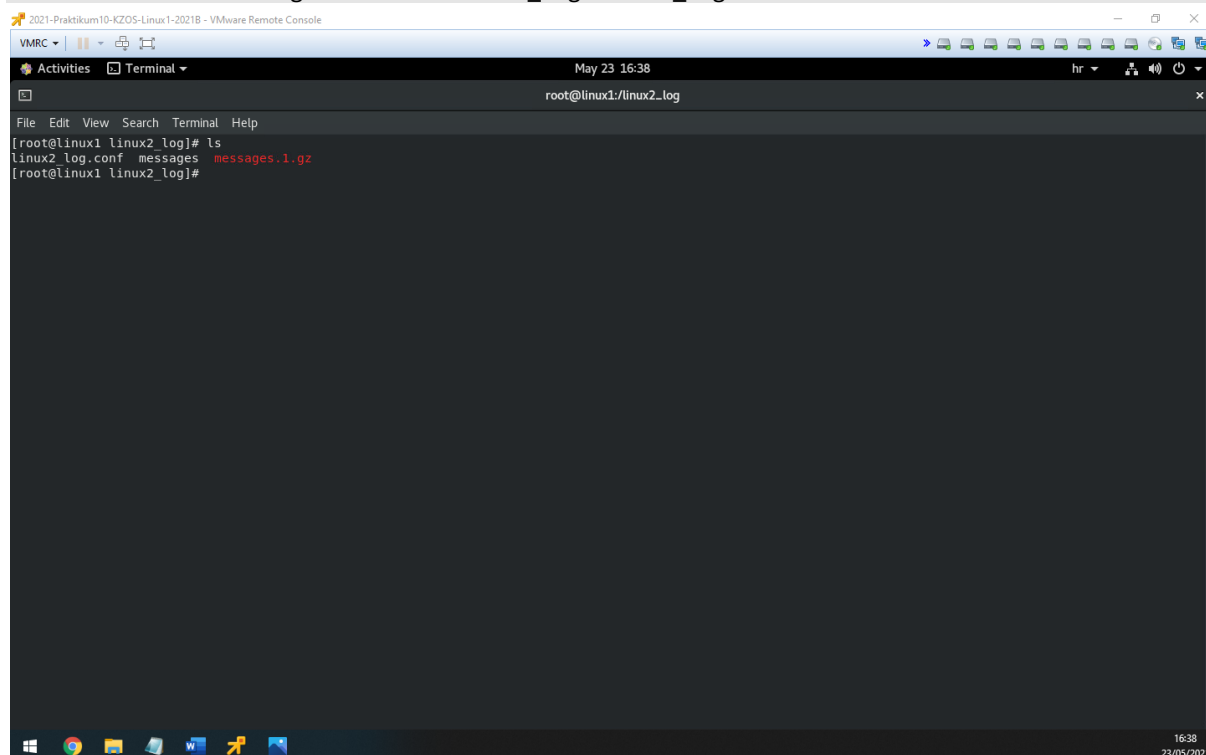
```
Crontab -e
* */2 * * * scp /var/log/messages root@linux1.antoniojanach.local:/linux2_log
```

Kad je log poslan na Linux1 poslužitelj potrebno je konfigurirati rotaciju logova svaka 2 dana koristeći kompresiju. To je napravljeno tako što je kreiran .conf datoteka na putanji: /linux2_log/linux2_log.conf.

```
Vim /linux2_log/linux2_log.conf
/linux2_log/messages {
  Rotate 4
  Daily
  Missingok
  Copytruncate
  Compress
}
```

Kad je konfiguracijska datoteka postavljena pomoću crontab-a je potrebno ju je pokretati svaka 2 dana.

```
Crontab -e
* * */2 * * logrotate -f /linux2_log/linux2_log.conf
```



Slika 52:Slanje log zapisa s rotacijom uspješno je postavljeno

7. Zaključak

U ovom projektu implementiran je informacijski sustav za tvrtku „Antonio Janach d.o.o“ koja se sastoji od dvije udaljene lokacije. Kroz implementaciju informacijskog sustava potrebno se je suočiti s nekim od vrlo važnih faza u razvoju, a to su analiza i konstruiranje. Analiza se odnosi na fazu prvog koraka u razvoju, a to je planiranje. Na planiranje se nadovezuje i analiziranje zahtjeva za informacijskim sustava. Kako bi se mogli analizirati konkretni zahtjevi za informacijskim sustavom, prvo je potrebno znati koji su poslovni subjekti izgradnje sustava. Nakon toga uzimaju se u obzir potrebe poslovanja konkretnog poslovnog subjekta i utjecaj razvoja na izgradnju i korištenje informacijskom sustava. Konstruiranje se odnosi na fazu koja daje odgovor na to kakav sustav treba biti, koje ciljeve treba realizirati i tko će biti korisnici. Kasnije slijedi utvrđivanje i otkrivanje zahtijeva.

Utvrđivanje se odnosi na analiziranje, pronalaženje te dokumentiranje zahtijeva za budući informacijski sustav, koji govore kakav bi sustav trebao biti, što bi trebao raditi i na kraju koje ciljeve i smisao trebao realizirati. Dakle, tu se nalazi poveznica s planiranjem gdje slijedi otkrivanje zahtjeva koje se provodi različitim tehnikama utvrđivanja, a podrazumijeva pregled dokumentacija, istraživanja relevantne literature, istraživanja i razumijevanje poslovnih praksi.

Nakon analiziranja korisničkih i sistemskih zahtijeva, dolazi se do koraka modeliranja podataka. Modeliranje podataka nadovezuje se na analizu, za cilj ima identifikaciju različitih kategorija podataka te utvrditi odnose među njima. Na modeliranje podataka nastavlja se modeliranje procesa koji se obavljaju i modeliranje na temelju prikupljenih podataka. Prolaskom kroz modeliranja procesa, potrebno je proći kroz proces modeliranja informacijskog sustava. Modeliranje informacijskog sustava daje rješenje koje je potrebno dokumentirati i napraviti tehnološki opis informacijskog sustava. Prije uvođenja informacijskog sustava u produkciju potrebno je izvršiti provjeru da li informacijski sustav radi na željeni način i na dani input daje željeni output. Jednom kad se informacijski sustav pusti u rad potrebno je kontinuirano održavati sustav te po potrebi vršiti promjene ili poboljšanja.

Kroz praktični dio ovog rada instalirana je kompletna domena i oba VPN poslužitelja, zajedno sa funkcionalnom mrežnom. Pod funkcionalnu mrežu odnosi se da se svako računalo u međusobno vidi na mreži i može komunicirati s bilo koji drugim računalom, uključujući lokaciju A i udaljenu lokaciju B koja je povezana VPN tunelom. Instalirani su svi Windows servisi i poslužiteljske komponente koje su navedene u zahtjevu zajedno sa instaliranim funkcionalnostima na Linux sustavima.

Popis slika

Slika 1: prikaz opisa infrastrukture kroz umnu mapu	6
Slika 2: prikaz topologije infrastrukture	7
Slika 3: prikaz instalacije AD DS uloge kao i prikaz kreirane domene imena „antoniojanach.local“	9
Slika 4: prikaz dodanih poslužitelja Server1 i Server2 u domenu "antoniojanach.local"	10
Slika 5: prikaz DNS zapisa i reverznih primarnih zona	10
Slika 6: prikaz problema	11
Slika 7: prikaz instalacije AD CS uloge kao root CA na Server1	12
Slika 8: prikaz instalacije AD CS uloga kao subordinate CA na Server2	13
Slika 9: prikaz kreiranog certifikata za Web poslužitelj i autentifikaciju korisnika	14
Slika 10: prikaz postavka "Certificate Services Client - Auto Enrollment Properties"	15
Slika 11: prikaz kreiranog domenskog certifikata na SERVER3 i SERVER4 poslužitelju	16
Slika 12: prikaz SSL postavki i promjene sadržaja zadane IIS Web stranice na oba poslužitelja	16
Slika 13: prikaz prozora autentifikacije korisnika pomoću certifikata	17
Slika 14: pristupanje stranici koristeći SSL certifikat	17
Slika 15: prikaz izlance naredbe koja prikazuje da je ServerDC NTP server	19
Slika 16: ilustracijski prikaz site-to-site VPN-a	20
Slika 17: openvpn servis je uspješno pokrenut	23
Slika 18: klijentski openvpn2 se je uspješno spojilo na serverski poslužitelj openvpn2	25
Slika 19: prikaz uspješne komunikacije između ServerDC i RODC poslužitelja koristeći VPN vezu	25
Slika 20: prikaz konfiguracije u "Active directory sites and services" na SERVERDC poslužitelju	27
Slika 21: DNS konfiguracija na RODC poslužitelju	27
Slika 22: prikaz svih dodanih Windowd poslužitelja u domenu	28
Slika 23: prikaz DNS konfiguracije na ServerDC poslužitelju	28
Slika 24: prikaz dodanog raspona adresa i autorizacije DHCP servera	29
Slika 25: prikaz instalirane DHCP uloge i postavke DHCP poslužitelja	30
Slika 26: Prikaz kreiranog „Storage pool-a“, virtualnog diska, volumena i uključene deduplikacije nad volumenom	31
Slika 27: prikaz kreiranog volumena na SERVER3 i SERVER4 poslužitelju imena "DFS"	32
Slika 28: prikaz instalirane DFS uloge i kreirane mape imena „Poslovanje“ na novokreiranom volumenu	32
Slika 29: prikaz kreiranog DFS namespace-a i dodanog Server4 poslužitelja	33
Slika 30: prikaz uspješne konfiguracije DFS + R na Server3 i Server4 poslužitelju	33
Slika 31: prikaz završne iSCSI konfiguracije na Linux1 poslužitelju	35
Slika 32: prikaz uspješnog spajanja na iSCSI target koristeći iSCSI inicijator	36
Slika 33: prikaz konfiguracije diskova	37
Slika 34: prikaz uspješno kreiranog klastera na SERVER1 i SERVER2 poslužitelju	37
Slika 35: prikaz uspješno konfigurirane uloge "File Server"	38
Slika 36: prikaz uspješno kreiranog dijeljenog SMB diska	39
Slika 37: prikaz uspješne instalacije FreeIPA servisa	40
Slika 38: prikaz kreiranog korisnika s dodanim grupama u AD-u	41
Slika 39 prikaz preuzetog i instaliranog FreeIPA certifikata na ServerDC poslužitelj	41
Slika 40: Preuzimanje CA certifikate od strane AD-a i prijenos istog na Linux1 poslužitelj	42
Slika 41: Prikaz uspješno postignute replikacije između Windows AD-a i FreeIPA servisa	43
Slika 42: konfiguracija "Group Policy" na ServerDC poslužitelju	45
Slika 43: prikaz uspješne konfiguracije SMB share-a	45
Slika 44: prikaz uspješnog postavljenog inkrementalnog backup-a na SMB share folder	49

Slika 45: DNS zapis za Web lokaciju	51
Slika 46: prikaz uspješne instalacije WordPress-a na Linux1 poslužitelj	53
Slika 47: MX zapis je dodan u DNS server	54
Slika 48: prikaz uspješno poslanog maila prema korisniku student	56
Slika 49: prikaz uspješno poslanog maila van domene	56
Slika 50: DNS zapis za Web lokaciju	58
Slika 51: prikaz uspješne instalacije WordPress-a na Linux1 poslužitelj	60
Slika 52: Slanje log zapisa s rotacijom uspješno je postavljeno	61

Literatura

OpenVPN

- [1] <https://openvpn.net/community-resources/how-to/>
- [2] <https://www.digitalocean.com/community/tutorials/how-to-set-up-an-openvpn-server-on-ubuntu-16-04>
- [3] <https://openvpn.net/vpn-server-resources/how-to-configure-the-openvpn-access-server/>
- [4] <https://openvpn.net/quick-start-guide/>
- [5] <https://www.digitalocean.com/community/tutorials/how-to-set-up-an-openvpn-server-on-ubuntu-18-04>
- [6] <https://www.digitalocean.com/community/tutorials/how-to-set-up-and-configure-an-openvpn-server-on-ubuntu-20-04>
- [7] <https://www.tecmint.com/install-openvpn-in-ubuntu/>

iSCSI

- [1] <https://computingforgeeks.com/how-to-configure-iscsi-initiator-on-centos-rhel/>
- [2] https://linuxhint.com/iscsi_storage_server_centos/
- [3] <https://computingforgeeks.com/configure-iscsi-target-and-initiator-on-centos-rhel/>
- [4] https://www.server-world.info/en/note?os=CentOS_8&p=iscsi&f=1

FreeIPA

- [1] https://docs.fedoraproject.org/en-US/Fedora/17/html/FreeIPA_Guide/active-directory.html
- [2] https://docs.fedoraproject.org/en-US/Fedora/17/html/FreeIPA_Guide/managing-sync-agmt.html
- [3] https://www.freeipa.org/page/Active_Directory_trust_setup
- [4] <https://community.cloudera.com/t5/Support-Questions/Sync-AD-users-using-FreeIPA-LDAP-with-a-trust-on-AD/td-p/238686>
- [5] <https://www.admin-magazine.com/Archive/2016/31/Integrating-FreeIPA-with-Active-Directory>

SMB

- [1] <https://www.linuxtechi.com/install-configure-samba-centos-8/>
- [2] <https://vitux.com/centos-samba-server/>
- [3] <https://linuxconfig.org/install-samba-on-redhat-8>
- [4] <https://www.howtoforge.com/how-to-install-samba-server-on-centos-8/>
- [5] https://www.server-world.info/en/note?os=CentOS_8&p=samba&f=1

Web server

- [1] <https://linuxconfig.org/install-wordpress-on-redhat-8>
- [2] https://linuxhint.com/install_wordpress_centos8/
- [3] <https://www.itzgeek.com/how-tos/linux/centos-how-tos/how-to-install-wordpress-with-nginx-on-centos-8-rhel-8.html>
- [4] <https://upcloud.com/community/tutorials/install-wordpress-lemp-centos-8/>
- [5] <https://www.informaticar.net/how-to-install-wordpress-on-centos-red-hat/>

Mail server

- [1] <https://www.linuxtechi.com/install-configure-postfix-mailserver-centos-8/>
- [2] <https://www.linuxbabe.com/mail-server/postfix-send-only-smtp-server-centos-8>
- [3] <https://www.linuxbabe.com/redhat/run-your-own-email-server-centos-postfix-smtp-server>
- [4] https://www.server-world.info/en/note?os=CentOS_8&p=mail&f=1
- [5] <https://linuxconfig.org/how-to-install-postfix-on-redhat-8>

ADDS

- [1] <https://social.technet.microsoft.com/wiki/contents/articles/52765.windows-server-2019-step-by-step-setup-active-directory-environment-using-powershell.aspx>
- [2] <https://computingforgeeks.com/how-to-install-active-directory-domain-services-in-windows-server/>

DNS

- [1] <https://computingforgeeks.com/install-and-configure-dns-server-in-windows-server/>
- [2] <https://www.wintelpro.com/install-and-configure-dns-on-windows-server-2019/>

DHCP

- [1] <https://computingforgeeks.com/how-to-install-and-configure-dhcp-server-on-windows-server/>
- [2] https://www.server-world.info/en/note?os=Windows_Server_2019&p=dhcp&f=2

NTP

- [1] <https://computingforgeeks.com/how-to-configure-ntp-server-in-windows-server/>
- [2] <https://linuxconfig.org/redhat-8-configure-ntp-server>

RODC

- [1] <https://www.windowstechno.com/step-by-step-guide-to-install-read-only-domain-controller-rodcc/>
- [2] <https://dailysysadmin.com/KB/Article/3947/how-to-create-a-windows-server-2019-rodcc-or-read-only-domain-controller/>

Storage spaces

- [1] <https://docs.microsoft.com/en-us/windows-server/storage/storage-spaces/deploy-standalone-storage-spaces>
- [2] <https://docs.microsoft.com/en-us/windows-server/storage/storage-spaces/deploy-storage-spaces-direct>
- [3] <https://abouconde.com/2019/03/20/creating-a-storage-pool-on-windows-server-2019/>

Root CA i subordinate CA

- [1] <https://www.prajwaldesai.com/install-enterprise-root-certificate-authority/#:~:text=On%20your%20Windows%20Server%202019,you%20begin%20window%2C%20click%20Next.>
- [2] <https://vmlabblog.com/2019/09/setup-server-2019-enterprise-ca-2-5-offline-root-ca/>
- [3] <https://vmlabblog.com/2019/09/setup-server-2019-enterprise-ca-3-5-subordinate-ca/>
- [4] <https://mjcb.io/blog/2020/03/09/certificate-authority-windows-server-2019-part-2/>

Failover cluster

- [1] <https://docs.microsoft.com/en-us/windows-server/failover-clustering/create-failover-cluster>
- [2] <https://www.veeam.com/blog/windows-server-2019-failover-cluster.html>
- [3] <https://www.xpertstec.com/how-to-create-a-failover-cluster-in-windows-server-2019-step-by-step/>

DFS

- [1] <https://www.windowscrush.com/how-to-set-up-dfs-namespaces-in-windows-server-2019.html>
- [2] <https://www.vembu.com/blog/distributed-file-system-dfs-windows-server-2016-brief-overview/>
- [3] <https://www.infotechram.com/wp-content/uploads/2019/03/How-to-setup-DFS-Windows-Server-2019.pdf>
- [4] <https://docs.microsoft.com/en-us/windows-server/storage/dfs-namespaces/dfs-overview>

IIS

- [1] <https://computingforgeeks.com/install-and-configure-iis-web-server-on-windows-server/>
- [2] <https://www.rootusers.com/how-to-install-iis-in-windows-server-2019/>
- [3] <https://enterprise.arcgis.com/en/web-adaptor/latest/install/iis/enable-iis-2019-components-server.htm>
- [4] <https://help.cadcorp.com/en/9.0/webmapAdmin/IIS-WinServer2019.htm>