Nadzor MPLS mreže alternativnog telekomunikacijskog operatora

Faletar, Dino

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

Dino Faletar

NADZOR MPLS MREŽE ALTERNATIVNOG TELEKOMUNIKACIJSKOG OPERATORA

DIPLOMSKI RAD

Zagreb, 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI
POVJERENSTVO ZA DIPLOMSKI ISPIT

Zagreb, 3. travnja 2020.

Zavod: **Zavod za informacijsko komunikacijski promet**
Predmet: **Planiranje telekomunikacijskih mreža**

DIPLOMSKI ZADATAK br. 5888

Pristupnik: **Dino Faletar (0246044549)**
Studij: Promet
Smjer: Informacijsko-komunikacijski promet

Zadatak: **Nadzor MPLS mreže alternativnog telekomunikacijskog operatora**

Opis zadatka:

Opisati značajke i funkcije MPLS mreža. Analizirati problematiku nadzora zatvorenih MPLS mreža. Opisati protokole (SNMP, ICMP, SSH, TELENT i dr.) i programske alate za nadzor MPLS mreža (Zabbix, Cacti, Zenoss, Nagios i dr.). Navesti i objasniti rješenja za nadzor zatvorenih MPLS mreža. Prikazati studiju slučaja pod nazivom: "Nadzor uređaja unutar zatvorenih MPLS mreža uz pomoć programskog alata Docker Engine".

Mentor:

doc. dr. sc. Ivan Grgurević

Predsjednik povjerenstva za diplomski ispit:

Sveučilište u Zagrebu
Fakultet prometnih znanosti

DIPLOMSKI RAD

NADZOR MPLS MREŽE ALTERNATIVNOG TELEKOMUNIKACIJSKOG OPERATORA

MPLS NETWORKS MONITORING OF ALTERNATIVE TELECOMMUNICATIONS OPERATORS

Mentor: doc. dr. sc. Ivan Grgurević

Student: Dino Faletar
JMBAG: 0246044549

Zagreb, rujan 2020.

SAŽETAK

Aktualno pitanje s kojim se danas susreće većina telekomunikacijskih operatora je kako obavljati nadzor mreže i mrežnih uređaja koji se nalaze unutar MPLS L3 VPN mreže. Kroz diplomski rad bit će razrađena aktualna problematika i opisano rješenje za nadzor MPLS L3 VPN mreže uz pomoć Zabbix *proxy* servisa pokretanog unutar alata *Docker Engine* koristeći kontejnere, tzv. *dockere*. Svrha diplomskog rada je istaknuti važnost nadzora komunikacijske mreže i mrežnih uređaja u realnom vremenu. U današnje vrijeme ispad pojedinog elementa mreže kod poslovnih korisnika može dovesti do velikih financijskih gubitaka. Upravo zato sustav Zabbix sa svojim funkcionalnostima omogućava telekomunikacijskim operatorima obavljanje proaktivnog nadzora kako bi se kvar mogao predvidjeti ili uočiti u trenutku nastanka.

Ključne riječi: nadzor mreže; telekomunikacijski operator; MPLS L3 VPN; Zabbix, proxy servis; docker.

SUMMARY

The current issue facing most telecommunications operators today is how to monitor the network and network devices located within the MPLS L3 VPN network. Through the thesis, current issues will be elaborated and a solution for monitoring the MPLS L3 VPN network will be described with the help of Zabbix proxy service running within the Docker Engine tool using containers, the so-called docker. The purpose of this thesis is to accentuate the importance of monitoring the communication network and network devices in real time. Nowadays, the failure of a single element of the network in business users can lead to large financial losses. That is why the Zabbix system, with its functionalities, enables telecommunications operators to perform proactive monitoring so that a system failure can be predicted or noticed at the time of occurrence.

Keywords: network monitoring; telecommunications operator; MPLS L3 VPN; Zabbix, proxy service; docker.

SADRŽAJ

1. UVOD

U današnje vrijeme gotovo je nezamisliv život bez pristupa Internetu. Većina uređaja koje koristimo u svakodnevnom poslovanju, obrazovanju i u obavljanju ostalih aktivnosti je povezana na komunikacijsku mrežu. Danas više ne postoji grana industrije koja nije obuhvaćena informatičkom tehnologijom. Budući da komunikacijske mreže i mrežni uređaji imaju sve veću ulogu u svim sustavima (poslovnim, obrazovnim, zdravstvenim, vojnim, itd.) trebalo je pronaći rješenje kako nadzirati sve elemente sustava, pogotovo ako su oni međusobno veoma udaljeni. Zbog sve većih potreba za nadzorom komunikacijskih mreža i sustava potkraj 20. stoljeća na tržištu su se počeli pojavljivati prvi specijalizirani programi za nadzor mrežnih komponenti i uređaja na udaljenim lokacijama preko mreže.

Danas, telekomunikacijski operatori nude poslovnim korisnicima usluge povezivanja svih poslovnica pomoću MPLS VPN[1] mreže kojom ostvaraju međusobnu povezanost i komunikaciju te imaju izlaz na Internet. Kako obavljati nadzor mrežnih uređaja koji se nalaze unutar MPLS VPN[2] mreže, aktualno je pitanje s kojim se danas susreće većina telekomunikacijskih operatora. Upravo ta problematika će detaljnije biti istražena kroz diplomski rad. Svrha diplomskog rada je istaknuti važnost nadzora komunikacijske mreže i mrežnih uređaja u realnom vremenu pošto ispad pojedinog elementa mreže može dovesti do velikih financijskih gubitaka, pogotovo ako je riječ o kvaru za čiju sanaciju je potrebno više vremena. Zbog toga je najvažnije uočiti kvar u trenutku nastanka ili ga pokušati predvidjeti kako bi se smanjili gubici u samom poslovanju. Cilj diplomskog rada je pomoću sustava Zabbix omogućiti nadzor mrežnih uređaja unutar MPLS VPN mreža alternativnog telekomunikacijskog operatora. Rad je podijeljen u osam povezanih cjelina:

1. Uvod
2. Značajke i funkcije MPLS mreža
3. Problematika nadzora zatvorenih MPLS mreža
4. Protokoli za nadzor MPLS mreža (SNMP, ICMP, SSH, TELENT i dr.)
5. Programski alati za nadzor MPLS mreža (Zabbix, Cacti, Zenoss, Nagios i dr.)
6. Rješenja za nadzor zatvorenih MPLS mreža

[1] VPN - tehnologija pomoću koje se promet iz privatne lokalne mreže zaštićen, tj. kriptiran prenosi preko nesigurne javne mreže kao što je Internet.
[2] MPLS VPN – tehnologija koja se koristi za komunikaciju i prosljeđivanje informacija između privatnih mreža (VPN-ova) putem MPLS mreže davatelja usluga.

7. Studija slučaja: Nadzor uređaja unutar zatvorenih MPLS mreža uz pomoć programskog alata *Docker Engine*
8. Zaključak

U Uvodu se daju osnovne smjernice rada, svrha i cilj te kratki opis po poglavljima/tezama diplomskog rada.

Drugo poglavlje obuhvaća značajke i funkcije MPLS mreža. Navedeni su i opisani glavni elementi, te je prikazan postupak usmjeravanja prefiksa kroz MPLS L3 VPN mrežu.

U trećem poglavlju opisana je problematika nadzora zatvorenih MPLS mreža. Navedeno je i opisano nekoliko elemenata na koje treba obratiti pažnju kako bi nadzor bio uspješan, a samim time osigurano zadovoljstvo krajnjeg korisnika.

U četvrtom su poglavlju izdvojeni i opisani samo glavni protokoli koju su potrebni za implementaciju i obavljanje nadzora unutar MPLS VPN mreža.

Peto poglavlje obuhvaća programske alate za nadzor. Izdvojeno je i ukratko opisano nekoliko programskih alata za nadzor te su izdvojene mogućnosti koje pružaju.

Šesto poglavlje obuhvaća rješenja za nadzor MPLS VPN mreža. Navedena su tri moguća rješenja koja su ukratko opisana i shematski prikazana. Opisani su sigurnosni aspekti te navedene prednosti i nedostatci svakog od rješenja.

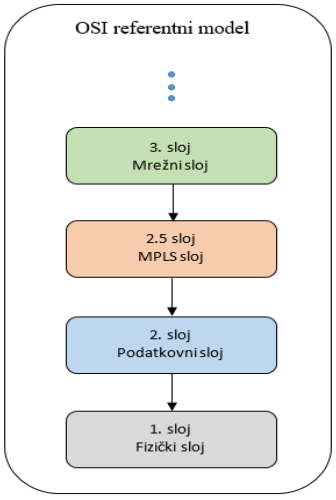
U sedmom je poglavlju prikazano praktično rješenje za nadzor MPLS VPN mreža. Istaknute su i opisane konfiguracijske naredbe korištene za uspostavu MPLS VPN mreža na PE i CE usmjerivačima, prikazana je konfiguracija za kreiranje kontejnera (engl. *dockera*) na serveru (poslužitelju), te je prikazan postupak kreiranja *proxy* servisa unutar programskog alata Zabbixa kao i namještanje akcija i radnji za otkrivanje uređaja unutar MPLS VPN mreže.

U Zaključku su rezimirani svi dobiveni rezultati, dana zaključna razmatranja, planovi i smjernice za buduća istraživanja na temu provedbe nadzora MPLS mreže alternativnog telekomunikacijskog operatora.

2. ZNAČAJKE I FUNKCIJE MPLS MREŽA

MPLS (engl. *Multi-Protocol Label Switching*)[3] ili višeprotokolno komutiranje oznaka je tehnologija koja osigurava tradicionalni model prosljeđivanja paketa kroz mrežu, na mnogo elegantniji, efikasniji i brži način nego što su to uspijevale prijašnje tehnike poput ATM-a (engl. *Asynchronous Transfer Mode*)[4] ili Frame Relay-a. Uvođenjem MPLS-a se ne želi zamijeniti dosadašnje IP usmjeravanje već se nastoji nadopunit nedostatke koji se očituju kod tradicionalnog usmjeravanja. U tradicionalnom modelu IP usmjeravanja zaglavlje svakog paketa koji prolazi mrežom se analizira pri svakom koraku na njegovom putu od usmjerivača *(*engl. *router)* do usmjerivača. Za razliku od tog načina MPLS tehnologija prilikom transporta paketa kroz mrežu koristi postupak zamjene oznaka *(eng. label swapping)*. Bitna prednost tog postupka je da se informacije iz zaglavlja paketa analiziraju samo jednom, a dalje se postupak usmjeravanja paketa zasniva samo na provjeravanju oznaka koje zapravo predstavljaju identifikacijske oznake paketa i fiksne su duljine. Samim time smanjuje se vrijeme potrebno za procesiranje informaciju unutar usmjerivača, [2].

MPLS kombinira značajke i prednosti pojednostavljenog konekcijskog prosljeđivanja na 2. sloju (podatkovnom) s fleksibilnošću i skalabilnošću usmjeravanja na 3. sloju (mrežnom). Također podržava protokole podatkovnog sloja (Ethernet, Token Ring, ATM, Frame Relay, PPP) i mrežnog sloja (IPv4, IPv6, IPX, AppleTalk). Pošto kombinira i integrira značajke i prednosti podatkovnog i mrežnog sloja, postavlja se pitanje kojem sloju unutar OSI referentnog modela zapravo MPLS tehnologija pripada.



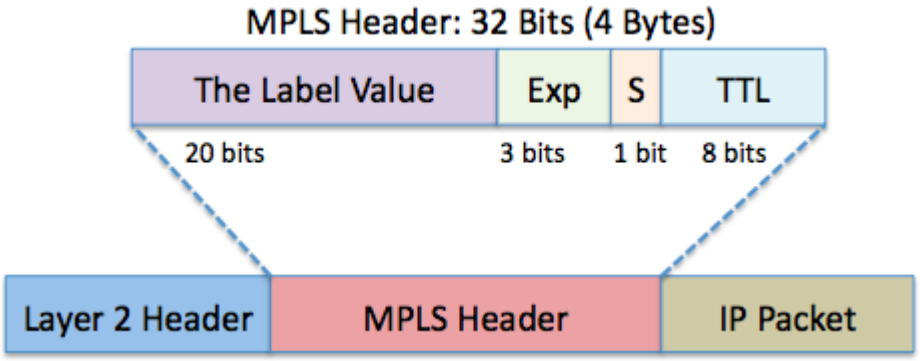
Slika 1 Zamišljeni prikaz MPLS-a unutar OSI referentnog modela, [8]

[3] MPLS - višeprotokolno komutiranje oznaka, protokol koji omogućava prosljeđivanje i prospajanje na temelju oznaka (engl. *label*).
[4] ATM - je tehnika prijenosa u telekomunikacijama koja se zasniva na asinkronom vremenskom multipleksiranju odsječaka prometa (ćelija) veličine 53 bajta.

Najtočniji odgovor bi glasio da pripada i podatkovnom i mrežnom sloju, tj. nalazi se između njih i predstavlja njihovu nadogradnju. U pojedinim literaturama se slikovito uvrštava u „2.5 sloj“, tzv. MPLS sloj (slika 1), [1], [2].

2.1. MPLS zaglavlje

MPLS zaglavlje sastoji se od četiri dijela ukupne duljine 32 bita. Naziva se još umetnutim zaglavljem *(*engl. *shim header)* upravo zbog pozicije gdje se enkapsulira. MPLS zaglavlje se ubacuje iza zaglavlja podatkovnog sloja, a ispred IP paketa mrežnog sloja.



Slika 2 Prikaz MPLS zaglavlja, [8]

Na slici 2 prikazan je format MPLS zaglavlja koje se sastoji od četiri dijela:

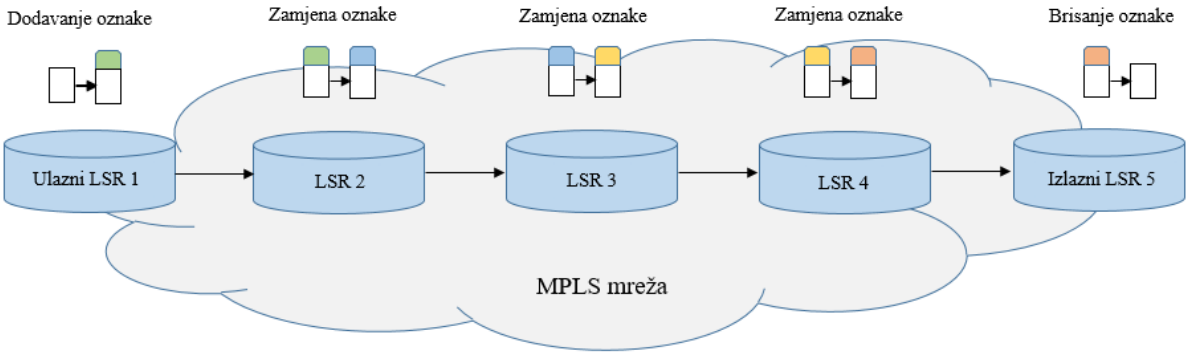
- Oznaka *(*engl. *Label)* - polje veličine 20 bita, predstavlja oznaku paketa, opisuje putanju za prosljeđivanje paketa do odredišta;
- Exp polje – polje veličine 3 bita, poznato kao *Class of Service (CoS)* bitovi, predstavlja eksperimentalne bitove korištene za određivanje tretmana, tj. prioriteta paketa, važna uloga prilikom upravljanja redovima i rasporedu posluživanja; Paketi mogu dobiti različit prioritet ovisno o CoS vrijednosti.
- S bit – polje veličine 1 bita, poznato kao *Bottom of Stack (BoS)*, podržava hijerarhijsko grupiranje oznaka na stog; Oznake se mogu slagati jedna na drugu, a S bit polje predstavlja posljednju oznaku na stogu. Kad je u navedenom polju vrijednost 1, znači da se radi o posljednjoj oznaci na stogu.
- TTL *(*engl. *Time to live)* - polje veličine 8 bitova, predstavlja životnih vijek MPLS paketa kako bi se izbjeglo stvaranje petlji. Pri svakom prolasku paketa kroz usmjerivač (LSR) TTL vrijednost se smanjuje za 1, a ukoliko TTL vrijednost iznosi 0 prije nego paket dođe do odredišnog čvora, paket se odbacuje.

2.2. Label Switch router (LSR)

LSR[5] je jedan od glavnih elemenata MPLS infrastrukture. LSR može biti usmjerivač ili preklopnik koji podržavaju i razumiju MPLS tehnologiju. LSR ima sposobnost razumijevanja MPLS oznaka što podrazumijeva slanje i primanje označenih paketa na podatkovnom sloju. U MPLS mreži (domeni) postoji tri vrste LSR-ova od kojih svaki ima svoju funkciju i zadaću:

- Ulazni LSR *(*engl. *Ingress LSR)* – nalazi se na ulazu u MPLS domenu, prima neoznačeni dolazni IP paket i stavlja oznaku, nakon toga prosljeđuje označen paket u MPLS mrežu.
- Posredni LSR *(*engl. *Intermediate LSR)* - kao što samo ime kaže, ima ulogu posrednika i nalazi se između dva LSR-a. Prima označen paket od ulaznog LSR-a, izvršava čitanje i zamjenu stare oznake novom te prosljeđuje paket s novom oznakom.
- Izlazni LSR *(*engl. *Egress LSR)* – nalazi se na izlazu iz MPLS domene, prima označeni paket, izvršava brisanje oznake i prosljeđuje paket izvan MPLS domene.

Radi jednostavnosti ulazi i izlazni LSR-ovi često se nazivaju rubnim LSR-ovima *(*engl. *Edge LSR)* ili LER.



Slika 3 Prikaz LSR usmjerivača i operacija koje obavljaju, [8]

Ovisno o kojoj vrsti LSR-a se radi, postoji mogućnost izvršavanja tri radnje: stavljanje *(*engl. *push)* jedne ili više oznaka na stog, zamjena *(*engl. *swap)* oznake i brisanje *(*engl. *pop)* oznake sa stoga. LSR ima mogućnost stavljanja jedne ili više oznaka na primljeni paket. Ukoliko primljeni paket već ima oznaku, LSR stavlja jednu ili više oznaka na postojeći stog oznaka. Ukoliko primljeni paket još nema oznaku, LSR kreira oznaku i stavlja je na prazan stog oznaka i šalje paket dalje. Također LSR ima mogućnost brisanja jedne ili više oznake sa vrha stoga oznaka prije nego što pošalje paket dalje. Osim dodavanja i brisanja oznaka, LSR također

[5] LSR - usmjerivač ili preklopnik koji podržava prosljeđivanje temeljeno na MPLS tehnologiji.

ima mogućnost zamjene oznaka. Kad LSR primi paket s oznakom, najviša oznaka sa stoga oznaka se mijenja novom oznakom i paket se šalje dalje. Na slici 3 se nalazi prikaz LSR usmjerivača i operacija koje izvršavaju.

LSR koji stavlja oznaku na pakete koji još nisu označeni zove se *„imposing LSR"* zato što je to općenito prvi LSR koji stavlja oznaku, a poznat je kao ulazni LSR. Zadnji LSR koji briše sve oznake sa označenog paketa naziva se *„disposing LSR*, poznat kao izlazni LSR.

2.3. Forwarding Equivalence Class (FEC)

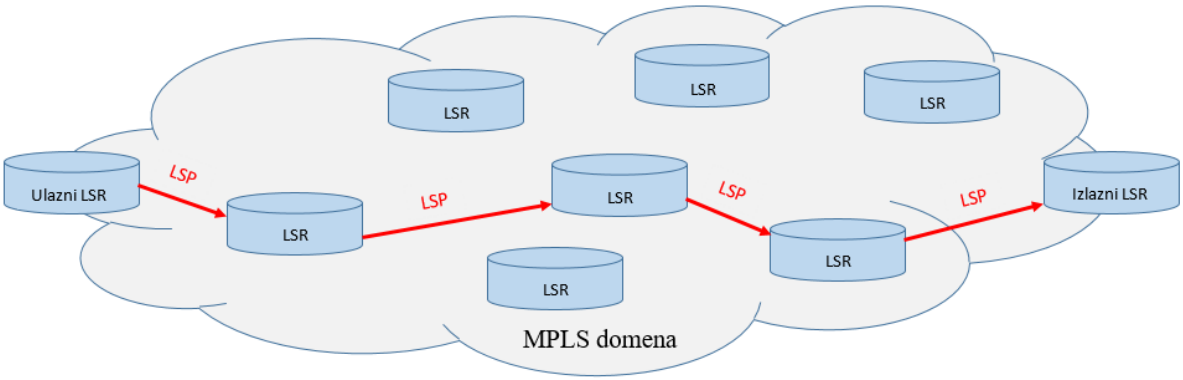
FEC je grupa IP paketa koji se prosljeđuju na isti način, istim putem i zahtijevaju jednak tretman pri prosljeđivanju, a kodira se kao oznaka *(*engl. *label).* Svi paketi koji pripadaju istom FEC-u imaju i istu oznaku. Međutim, ne pripadaju svi paketi s istom oznakom istom FEC-u jer se njihove EXP vrijednosti mogu razlikovati, a samim time i tretman prosljeđivanja može biti drugačiji te mogu pripadati drugom FEC-u. Prilikom ulaska IP paketa u MPLS domenu ulazni LSR razvrstava i označava pakete, a ujedno i odlučuje koji paket pripada kojem FEC-u. Paketi se mogu grupirati na temelju:

- Prefiksa adrese *(*engl. *address prefix)*
- Adresi hosta *(*engl. *Host address)* i
- Kvaliteti usluge - QoS *(*engl. *Quality of Service)*[6].

2.4. Label switched path (LSP)

LSP[7] je niz LSR-ova koji prolaze označeni paketi kroz MPLS mrežu. U osnovi, LSP je definirani put kroz MPLS mrežu ili dio MPLS mreže koji označeni paketi prolaze. Ulazni LSR je prvi LSR u LSP-u, dok je izlazni LSR zadnji LSR u LSP-u, a između u LSP-u su posredni LSR-ovi. Važno je naglasiti da su LSP-ovi jednosmjerni, što znači da se povratni promet šalje preko drugog LSP-a koji se treba uspostaviti. Također, bitno je znati da LSP-ovi nisu uvijek najkraći putevi. Na slici 4 prikazan je uspostavljen LSP između ulaznog i izlaznog LSR-a.

[6] QoS - Termin koji se koristi u računalnim mrežama kako bi se prikazale performanse i kvalitete nekog sustava ili usluge koje se pružaju krajnjem korisniku.
[7] LSP je definirani put kojim označeni paketi prolaze kroz MPLS mrežu.



Slika 4 Prikaz LSP-a između ulaznog i izlaznog LSR-a
Izvor: [1]

Za razliku od tradicionalnog IP prosljeđivanja gdje ne postoji fiksni put koji paket mora proći, u MPLS prosljeđivanju se unaprijed definira put, tj. LSP koji se mora slijediti tijekom prijenosa paketa, [1], [2].

2.5. Distribucija oznaka

Prva oznaka na paket stavlja se na ulaznom LSR-u i ta oznaka pripada jednom LSP-u. Put kojim paket treba ići kroz MPLS mrežu je vezan za taj LSP. Kada paket putuje kroz MPLS mrežu jedino što mu se mijenja je gornja oznaka na stogu oznaka pri svakom skoku. Ulazni LSR stavlja jednu ili više oznaka na paket. Posredni LSR radi zamjenu gornje oznake (dolazne oznake) označenog paketa s drugom oznakom (odlaznom oznakom) i šalje paket dalje. Izlazni LSR na postojećem LSP-u briše oznake vezane za taj LSP i prosljeđuje paket van MPLS mreže.

Razmjena oznaka bit će objašnjena na primjeru jednostavne MPLS mreže „IPv4 preko MPLS-a *(*engl. *IPv4-over-MPLS)*. Običan IPv4 preko MPLS-a je mreža koja se sastoji od LSR-ova koji za usmjeravanje koriste *Interiror Gateway Protocol (IGP)*[8] kao npr. *Open Shortest Path First (OSPF)*[9], *Intermediate System-toIntermediate System (IS-IS)*[10], *and Enhanced Interior Gateway Routing Protocol (EIGRP)*[11]. Ulazni LSR gleda IPv4 adresu odredišta, stavlja oznaku i prosljeđuje paket. Svaki sljedeći LSR prima paket sa oznakom, radi zamjenu dolazne oznake sa odlaznom oznakom i prosljeđuje paket. Izlazni LSR uklanja oznaku te prosljeđuje IPv4 paket bez oznake na odlazni link. Da bi ovo radilo, susjedni LSR-ovi moraju se složiti

[8] IGP - protokol provodi usmjeravanje unutar autonomnog sustava.
[9] OSPF - je link state hijerarhijski IGP protokol za usmjeravanje, koristi Dijkstra algoritam za izračunavanje najkraćeg puta do sljedećeg čvora.
[10] IS-IS - link state protokol, pripada skupini IGP protokola namijenjenih usmjeravanju unutar autonomnog sustava.
[11] IGRP - je distance vector IGP protokol tvrtke Cisco, omogućuje korištenje višestrukih metrika kao što su: propusnost, opterećenje, kašnjenje, MTU i pouzdanost.

koja će se oznaka koristiti za svaki IGP prefiks. Upravo zato, svaki posredni LSR mora imat mogućnost shvatiti s kojom odlaznom oznakom bi trebala biti zamijenjena ulazna oznaka. To znači da je potreban mehanizam kojeg će usmjerivači koristiti za prosljeđivanje paketa. Oznake su lokalne za svaki par susjednih usmjerivača, tj. nemaju globalno značenje u mreži. Da bi se susjedni usmjerivači složili koju oznaku će koristiti za koji prefiks, potreban im je neki oblik komunikacije. U protivnom, usmjerivači ne znaju koja odlazna oznaka odgovara kojoj dolaznoj oznaci. Upravo zato je potreban *„Label Distribution Protocol (LDP)“*, [1].

2.6. Label Distribution Protocol (LDP)

LDP[12] je protokol definiran za distribuciju oznaka. To je skup procedura i poruka pomoću kojih LSR-ovi uspostavljaju LSP kroz MPLS mrežu preslikavanjem informacija o usmjeravanju mrežnog sloja izravno u putanje podatkovnog sloja. LDP povezuje FEC sa svakim uspostavljenim LSP-om. FEC je skup paketa koji su preslikani na određeni LSP i šalju se preko tog LSP-a preko MPLS mreže.

Da bi se paketi prenijeli preko LSP-a kroz MPLS mrežu, svi LSR-ovi moraju pokrenuti LDP i povezivanje oznaka. Kada svi LSR-ovi imaju oznake za određeni FEC, paketi mogu biti proslijeđeni na LSP pomoću korištenja zamjene oznaka paketa na svakom LSR-u. Radnje koje će obavljati s oznakama (dodavanja, zamjena i brisanje) poznate su svakom LSR-u tako što su zapisane u tablicu LFIB-u *(*engl. *Label forwarding information base).* LFIB je tablica na temelju koje se radi prosljeđivanje paketa s oznakom, a popunjava se podacima o povezivanju oznaka iz posebne tablice LIB *(*engl. *Label information base)* koja sadrži sve podatke o susjednim usmjerivačima, tj. sva lokalna i udaljena povezivanja oznaka.

Svi izravno povezani LSR-ovi koji koriste LDP za međusobnu razmjenu informacija o poveznici oznaka/FEC nazivaju se *"LDP Peers"*. Prilikom razmjene informacija uspostavljaju LDP sesiju *(*engl. *LDP Session).* Dva povezana LSR-a (*LDP Peers*) izmjenjuju poruke i podatke o upravljanju oznaka putem LDP sesije. LDP je dvosmjeran protokol. Postoji četiri kategorije LDP poruka:

- *„Discovery“* poruke; Otkrivanje LSR-ova koji pokreću LDP.
- *„Session“* poruke; Uspostava i održavanje sesija između LDP peer-ova.
- *„Advertisement“* poruke; Oglašavanje preslikavanja oznaka, koristi se za stvaranje, promjenu i brisanje mapiranja oznaka za FEC.

[12] LDP - protokol za distribuciju oznaka između LSR usmjerivača.

- *„Notification“* poruke; Poruke obavijesti, koje se koriste za pružanje savjetodavnih informacija i za signaliziranje informacija o pogrešci.

Kad dva LSR-a pokrenu LDP i dijele jednu ili više veza između njih, trebali bi otkrivati jedni druge pomoću *„Hello“* poruka. Drugi je korak je uspostava sesije preko TCP veze. Preko ove TCP veze LDP oglašava preslikavanje oznaka poruka između dva LDP peer-ova. Te poruke o preslikavanju oznaka koriste se za oglašavanje, promjenu ili prekidanje povezivanja oznaka. LDP pruža mogućnost obavještavanja LDP susjeda slanjem obavijesti o pogreškama ili dodatnim informacijama, [9].

2.7. Label Forwarding Instance Base (LFIB) i Label Information Base (LIB)

LFIB je tablica koja se koristi za prosljeđivanje paketa s oznakom. U tablici se nalaze dolazne i odlazne oznake za LSP. Dolazna oznaka je oznaka od lokalnog povezivanje *(*engl. *local binding)* na određeni LSR. Odlazna oznaka je oznaka od udaljenog povezivanja *(*engl. *remote binding)* odabrana od strane LSR-a od svih naučenih udaljenih povezivanja. Sva naučena udaljena povezivanja sa svim oznaka su zapisana su unutar LIB. LIB je zapravo tablica koja sadrži sva naučena povezivanja i sve oznake na jednom mjestu. LFIB odabire samo jednu od ponuđenih odlaznih oznaka od svih naučenih udaljenih povezivanja koje se nalaze u LIB-u i zapisuje ih. Odabrana odlazna oznaka bira se na temelju najboljeg puta *(*engl. *best path)* iz tablice usmjeravanja, [1], [2].

2.8. MPLS L3 VPN

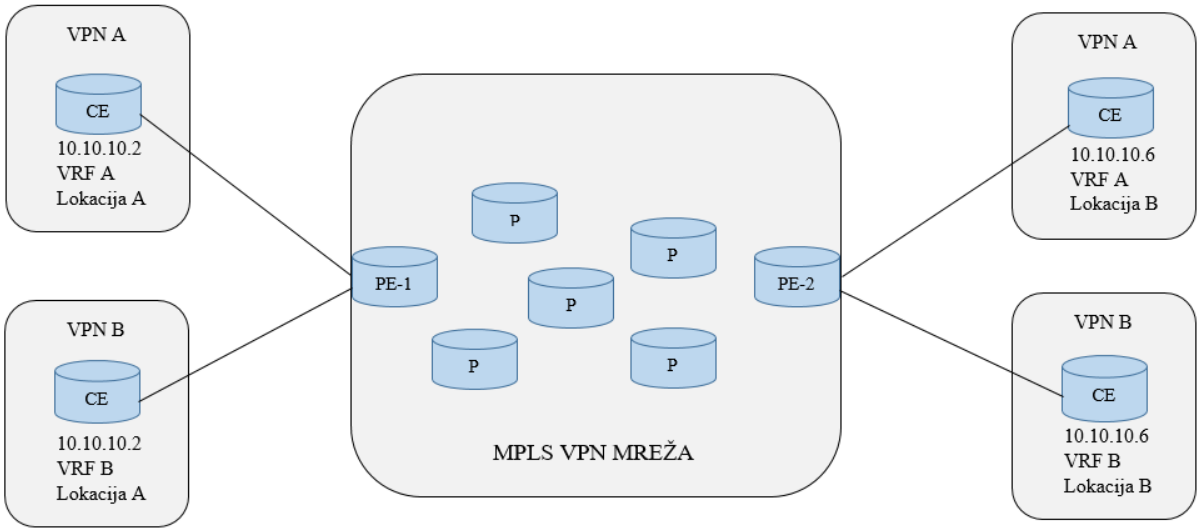
MPLS tehnologija danas je jako prihvaćena i raširena jer pruža razna aplikacijska proširenja i usluge. Neka od aplikacijskih rješenja i usluga koje nudi su: MPLS VPN *(*engl. *MPLS Virtual Private Networks)*, MPLS QoS *(*engl. *MPLS Quality of Service)*, MPLS T*E (*engl. *MPLS Traffic Engineering)*. Pošto se diplomski rad bavi problematikom nadzorom MPLS L3 VPN mreža, navedeno će biti detaljnije opisano. U slučaju MPLS VPN-a, ulazni i izlazni LSR se nazivaju *„provider edge“* (PE)[13] usmjerivači, dok se posredni LSR-ovi se nazivaju

[13] PE usmjerivač - naziv za ulazni i izlazni LSR u mreži operatora.

„provider“ (P)[14] usmjerivači. Usmjerivač na korisničkoj strani naziva se *„customer edge“* (CE)[15]. Radi lakše razumijevanja, pojmovi CE, PE i P usmjerivači će biti korišteni u nastavku.

MPLS L3 VPN često koriste ISP-ovi *(*engl. *Internet Service Provider)*[16] za pružanje usluga poslovnim korisnicima koji imaju više udaljenih poslovnica. Udaljene poslovnice se spajaju koristeći virtualne privatne mreže koje su realizirane preko MPLS mreže. ISP preuzima kompletno upravljanje nad uslugama mrežnog sloja, što znači da preuzima i kompletno usmjeravanje i prosljeđivanje korisničkih paketa. Unutar MPLS L3 VPN-a, informacije o usmjeravanju od jednog korisnika su u potpunosti odvojene od ostalih korisnika i proslijeđene preko MPLS mreže davatelja usluga.

Na slici 5 prikazan je shematski prikaz MPLS VPN mreže. PE i P usmjerivači su usmjerivači u nadležnosti davatelja usluge (ISP-a). Na korisničkoj strani se nalaze CE usmjerivači. PE usmjerivači imaju direktnu vezu sa CE usmjerivačem i međusobno ostvaruju komunikaciju baziranu na 3.sloju, tj. mrežnom sloju. P usmjerivač je usmjerivač bez izravne veze s CE usmjerivačima na korisničkoj strani. U MPLS VPN implementaciji, PE i P usmjerivači uspostavljaju MPLS što znači da moraju biti u mogućnosti međusobno razmjenjivati oznake i prosljeđivati pakete s oznakama.



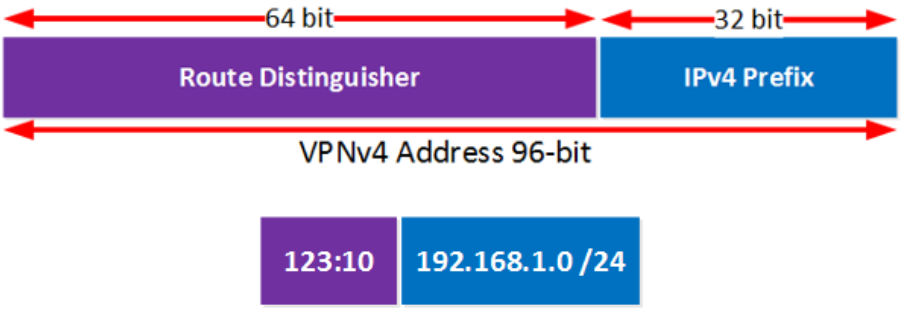
Slika 5 Shematski prikaz MPLS VPN mreže
Izvor: [1]

U MPLS L3 VPN mrežama, poslovnim korisnicima na udaljenim lokacijama omogućeno je da imaju vlastitu IP shemu. To znači da mogu koristiti privatne IP adrese i pri

[14] P usmjerivač – naziv za posredne LSR-ove unutar mreže operatora.
[15] CE usmjerivač - naziv za usmjerivače koji se nalaze na korisničkoj strani mreže.
[16] ISP - pružatelj internetskih usluga, tvrtka koja svojim korisnicima pruža uslugu pristupa Internetu.

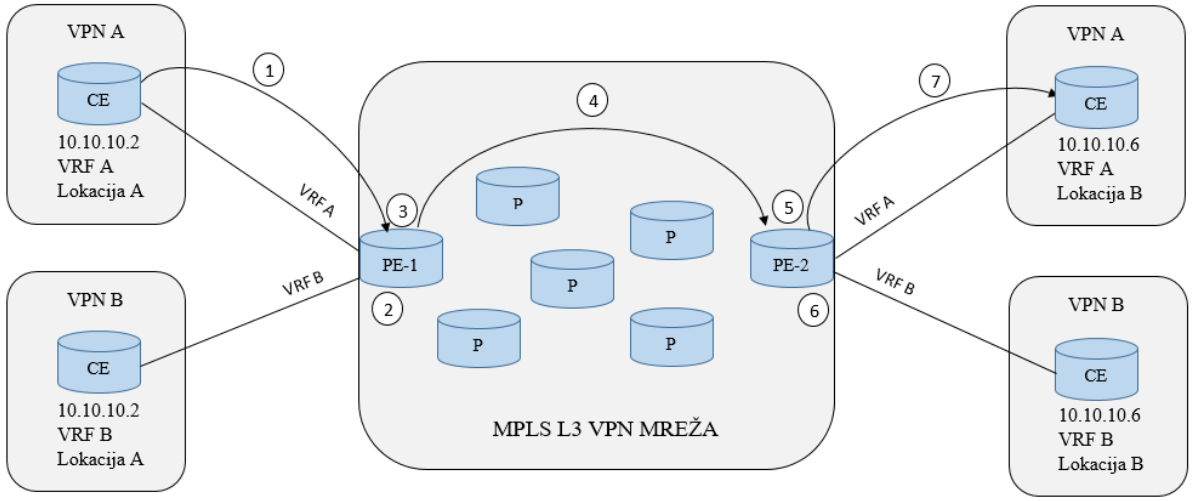
tome ne moraju brinuti da li tu adresu već ima drugi korisnik. Svaki korisnik, a samim time i IP adrese su međusobno logički odvojene te promet jednog korisnika neće ulaziti u promet i mrežu drugog korisnika. Budući da bi usmjeravanje trebalo biti zasebno i privatno za svakog kupca (VPN) na PE usmjerivaču, svaki VPN treba imati vlastitu tablicu usmjeravanja. Ova privatna tablica usmjeravanja naziva se VRF *(*engl. *Virtual Routing and Forwarding)*[17] tablica usmjeravanja. Sučelje PE usmjerivača prema CE usmjerivaču može pripadati samo jednom VRF-u. Svaki korisnik unutar ISP koristit će zaseban VRF. PE usmjerivač osim što sadrži globalni tablicu usmjeravanja, također sadrži i VRF tablicu usmjeravanja svakog korisnika. VRF tablica usmjeravanja sadrži prefikse koji su prepuni dinamičkih protokola usmjeravanja i statičkog usmjeravanja, baš kao i tablica globalnog usmjeravanja. Koncept metrike, udaljenosti, sljedećeg skoka ne mijenja se. Budući da je instanca VRF povezana sa sučeljima, samo IP paketi koji ulaze u PE usmjerivač preko tih VRF sučelja prosljeđuju se i upisuju u VRF tablicu usmjeravanja. Razmjena informacija o VRF tablicama između PE usmjerivača obavlja se pomoću *„Multiprotocol BGP-a“* (MP-BGP[18]). Pomoću MP-BGP-a uspostavlja se veza između PE usmjerivača, tzv. *„BGP neighbour“*. Prilikom razmjena informacija između PE usmjerivača bitno je da se IPv4 prefiks svakog korisnika jedinstveno odredi pomoću RD *(*engl. *Route Distinguisher)*[19] kako ne bi došlo do preklapanje adresa i problema prilikom usmjeravanja. RD je 64-bitno polje, a omogućuje korisnicima korištenje bilo koje privatne IP adrese na svojim CE usmjerivačima bez obzira na IP adrese drugih korisnika. Važno je napomenuti da RD ne označava kojem VRF-u pripada adresa/prefiks. RD se konfigurira u formatu ASN:NN, gdje ASN označava broj autonomnog sustava ISP-a, a NN proizvoljan broj (često ID korisnika, datum i dr.). RD u kombinaciji sa IPv4 prefiksom čini VPNv4 adresu ili prefiks koja se razmjenjuje između PE usmjerivača pomoću MP-BGP-a. Na slici 6 prikazana je VPNv4 adresa koja se sastoji od 96 bita.

[17] VRF - predstavlja virtualno usmjeravanje i prosljeđivanje, tehnologija koja omogućuje istodobno postojanje više instanci tablice usmjeravanja unutar istog usmjerivača.
[18] MP-BGP - proširenje BGP protokola koje omogućuje oglašavanje različitih vrsta adresa (IPv4 i IPv6 unicast, IPv4 i IPv6 multicast, VPNv4).
[19] RD - atribut unutar BGP protokola, osigurava jedinstvenost svih prefiksa/ruta unutar MPLS VPN mreže.



Slika 6 Prikaz VPNv4 adrese, [10]

Kako bi PE usmjerivači znali kojem točno VRF-u pripada koja adresa/prefiks koristi se oznaka RT *(*engl. *Route Target)*[20]. RT predstavlja atribut BGP proširene zajednice *(*engl. *BGP extended community)*, iste je veličine kao RD (64 bita) i konfigurira se u istom formatu ASN:NN.



Slika 7 Prikaz usmjeravanja adrese kroz MPLS L3 VPN mrežu
Izvor: [1]

Na slici 7 se nalazi prikaz usmjeravanja adrese kroz MPLS L3 VPN mrežu s označenim brojevima koji definiraju redoslijed kako se stvari odvijaju. Ispod je po brojevima ukratko opisano što se točno odvija:

1. Pomoću IGP ili eBGP protokola oglašava se IPv4 adresa između CE i PE-1 usmjerivača;
2. Na PE-1 usmjerivaču IPv4 prefiks je dodan u VRF tablicu usmjeravanja;
3. IPv4 prefiks se redistribuira unutar MP-BGP. RD se dodaje IPv4 prefiksu te zajedno čine VPNv4 adresu. Također se dodaje i RT kako bi se znalo kojem VRF-u pripada;

[20] RT - atribut unutar BGP protokola, specificira kojem točno VRF-u pripada koji prefiks/ruta.

4. MP-BGP oglašava i šalje VPNv4 adresu sa MPLS oznakom i RT-om PE-2 usmjerivaču;
5. PE-2 usmjerivač na temelju RT oznake zna kojem VRF-u adresa pripada te gdje će biti zapisana. Briše se RD sa VPNv4 adrese;
6. Na PE-2 usmjerivaču IPv4 adresa je dodana u odgovarajuću VRF tablicu usmjeravanja;
7. Pomoću IGP ili eBGP protokola oglašava se IPv4 adresa između PE-2 i CE usmjerivača, [1], [3].

3. PROBLEMATIKA NADZORA ZATVORENIH MPLS MREŽA

Prije kreiranja nadzora zatvorenih MPLS VPN mreža unutar alternativnog telekomunikacijskog operatora potrebno je unaprijed dogovoriti s korisnikom koji dio mreže i koje parametre želi imati pod nadzorom. Kako bi nadzor bio uspješan, a samim time osigurano zadovoljstvo krajnjeg korisnika, operator prilikom kreiranja nadzora zatvorenih MPLS VPN mreža mora obratiti pažnju na nekoliko elemenata koji će biti opisani u nastavku:

- Dostupnost MPLS VPN mreža za nadzor
- Sigurnost
- Funkcionalnost i jednostavnost
- Rast nadzornog rješenja i
- Prezentacija prema korisniku – *„multi tenant“*.

3.1. Dostupnost MPLS VPN mreža za nadzor

MPLS VPN predstavlja zatvorenu mrežu koja koristi privatne IP adrese. Svaka MPLS VPN mreža ima zasebnu VRF tablicu usmjeravanja koja se distribuira unutar mreže alternativnog operatora pomoću MP-BGP protokola. Svaka MPLS mreža nema dodirnih točaka sa globalnom tablicom usmjeravanja od operatora niti sa drugim mrežama. Pošto su to zatvorene/privatne mreže, postavlja se pitanje kako pristupiti iz nadzorne mreže operatora u privatne korisničke mreže. Ako se ne osigura pristup zatvorenim korisničkim mrežama i parametrima onda nije moguće obavljati nadzor. Kako bi se osigurao pristup, a ujedno i nadzor, bit će implementirano rješenje za „ulaz“ u zatvorenu MPLS mrežu uz pomoć *proxy*[21] servisa nadzornog alata Zabbix koji se pokreće unutar alata *Docker Engine*[22] (koristeći „kontejnere“). Kontejner će imati pristup VRF tablici usmjeravanja korisnika za kojeg alternativni operator obavlja nadzor. U nastavku diplomskog rada, detaljnije će biti opisano navedeno rješenje.

[21] PROXY - posrednik između klijenta i glavnog poslužitelja.
[22] Docker Engine - alat dizajniran tako da olakšava razvoj, implementaciju i pokretanje aplikacija koristeći kontejnere.

3.2. Sigurnost

Prilikom implementacije nadzor MPLS VPN rješenja jedna od najbitnijih stvar koju je potrebno imati na umu je sigurnost. IP promet unutar MPLS VPN mreže mora ostati unutar korisničke privatne mreže. Ne smije se dopustiti da se dogodi takozvano „curenje“ prometa *(*engl. *MPLS leaking)*. To znači da ne smije se doći do „curenja“ prometa jedne privatne korisničke mreže u privatnu mrežu drugog korisnika, ili da se promet iz jedne MPLS VPN mreže korisnika miješa sa globalni IP prometom unutar mreže operatora. Također potrebno je zaštiti privatne mreže unutar MPLS VPN-a od napada izvana. Zbog funkcionalnost i jednostavnosti koristi se centralni sustav za nadzor, ali kod takvog pristupa treba obratiti pozornost na sigurnosne aspekte. Kako bi se osigurala sigurnost koristit će se: *proxy* servisi u izoliranim okolinama (kontejnerima), vatrozid[23] *(*engl. *Firewall)* i drugi sigurnosni mehanizmi metro mreže operatora kao npr. VLAN-ovi[24]. Sigurnosne mehanizme nije dovoljno postaviti na jednoj točci u mreži te se zato koristi vatrozid da brani promet iz MPLS VPN mreže prema *proxy* servisima za nadzor. Na razini vatrozida striktno je propušten samo odgovarajući promet kao: SNMP, ICMP, telnet, SSH i dr. Također koriste se VLAN-ovi unutar metro mreže operatora kako ne bi došlo do miješanja prometa na metro razini, kao što se informacije o nadzoru mreža i uređaja unutar mreža provode zasebnim VLAN-om kroz mrežu operatora. Moguće je i korištenje izolirane „kontejner okoline“ za razdvajanje pristupa na serverskoj razini MPLS VPN mrežama. Sve navedene postupke i mehanizme potrebno je primjenjivati kako bi se osigurala sigurnost.

3.3. Funkcionalnost i jednostavnost

Prilikom implementacije rješenja za nadzora MPLS VPN mreže, osim sigurnosti sustava, također treba brinuti o jednostavnosti implementacije samog rješenja, održavanju funkcionalnosti sustava kao i o cijeni. Prilikom planiranja implementacije sustava, ukoliko postoje ograničenja s resursima, važno je da je svaka komponenta otvorenog koda *(*engl. *open source),* tj. da je besplatna za korištenje i razvoj. Naravno, u većini slučajeva osim besplatnih verzija, postoje i *„premium“* usluge koje se naplaćuju, a obuhvaćaju aktivan razvoj *(*engl.

[23] Vatrozid - je mrežni uređaj čija je namjena filtriranje mrežnog prometa tako da se stvori sigurnosna zona.
[24] VLAN - Virtualna lokalna mreža, predstavlja način logičke segmentacije mreže koja se može dinamički mijenjati i nije ovisna o fizičkoj topologiji mreže.

development), široko dostupnu aktivnu zajednicu, sigurnosna rješenja i nadogradnje. *Premium* usluge obuhvaćaju mogućnost plaćanja implementacije ili prilagođavanja sustava za nadzor od strane pojedinih firmi koja to nude kao uslugu. Također, trebala bi postojati mogućnost jednostavne zamjene svake od komponenti sustava za nadzor sa drugim alternativnim rješenjem. Primjerice, Zabbix sustav za nadzor zamijeniti s nekim drugim sustavom za nadzor, *dockere* koji imaju pristup VRF tablici usmjeravanja s nekim drugim alternativnim rješenjima i dr. Prilikom svakog koraka u razvoju, implementaciji i korištenju usluge potrebno je napraviti dobru dokumentaciju pošto više timova sudjeluje od samog razvoja, implementacije do korištenja nadzora. Tim koji razvija uslugu predat će rješenje timu koji će odraditi implementaciju usluge (npr. sistemski administratori), a uslugu će koristiti mrežni administratori za nadzor. Dobro je razgraničiti takvo rješenje na više komponenti tako da svaka komponenta ima i mogućnost jednostavnog *"debugiranja"*, tj. istraživanja i utvrđivanja poteškoće ili problema. Vođenim time u ovome radu krenuli smo u implementaciju besplatnog rješenja kao što su Zabbix sustav za nadzor i *dockeri*. Oba rješenja imaju i poslovnu *(*engl. *enterprise)* podršku kao i bogati odaziv i podršku iz zajednice korisnika preko foruma itd. Kod ovakvog rješenja moguće je cijelu implementaciju napraviti na jednom *(*engl. *stand alone)* serveru ili distribuirati na više servera (virtualni strojevi kao serveri). Svaku komponentu sustava možemo pokretati na više servera, ali treba imati na umu da zajedno s time rastu i troškovi. Druga stvar koju je potrebno uzeti u obzir je jednostavnost održavanja sustava i nadogradnja. Također ukoliko pojedini korisnici imaju posebne zahtjeve vezane za nadzor koji odstupaju od standardnog rješenja koji se pruža korisnicima, treba postojati mogućnost realizacije traženoga na što jednostavniji način, npr. uz pomoć gotovih predložaka *(*engl. *template)* ili pozivanjem vanjskih skripti koje su podržane unutar Zabbix sustava za nadzor. Također, rješenje za nadzor mora pružati: visoki stupanj automatizacije što podrazumijeva *"discovery"* procese, tj. automatsko otkrivanje mrežnih uređaja i parametara u mreži, optimizaciju nadzora uređaja *(*engl. *low-level discovery)* - da se ne prikupljanju informacije s sučelja uređaja koji nisu aktivni, *„alerting"* sustav, tj. sustav za slanje obavijesti i upozorenja, prezentaciju prema korisnicima.

3.4. Rast nadzornog rješenja

Osim već navedenih elemenata, prilikom planiranja implementacije rješenja za nadzor MPLS VPN usluge potrebno je uzeti u obzir rast usluge, tj. mogućnost proširenja. U kontekstu rasta i mogućnosti proširenja treba razmišljati o horizontalnom i vertikalnom rastu. Horizontalan rast se odnosi na sve usluge koje se nude unutar rješenja nadzora. Od osnovnog nadzora mrežnih uređaja, automatskog otkrivanja uređaja u mreži, mogućnosti slanja obavijesti i upozorenja prilikom pogrešaka i neželjenih događaja u mreži, interaktivnih grafova gdje se klikom miša može odabrati proizvoljno razdoblje unutar kojeg se želi provjeriti stanje mrežnih parametara (prometa, temperature, memorije i dr.), gotovih predložaka za FTP[25], HTTP[26], HTTPS[27], LDAP[28], MySQL[29], SMTP[30], SSH[31], Telnet[32], ICMP[33] i dr., do dodatnih mogućnosti koje se ostvaraju učitavanjem vanjskih skripti i ostalo. Kada se govori o vertikalnom rastu tada se misli na sve komponente sustava. Moguće je umjesto jednog centralnog servera za nadzor ugraditi više servera, proširiti memoriju diska, povećati brzinu procesora za obradu podataka, dodati više baza za pohranu i dr. Prilikom planiranja horizontalnog i vertikalnog rasta, treba voditi brigu o već navedenim elementima, prije svega o sigurnosti, jednostavnosti implementacije i razvoja, a opet da sve bude u skladu s resursima koji su dostupni te da budu zadovoljeni korisnički zahtjevi.

3.5. Prezentacija prema korisniku

Nakon što su zadovoljeni svi navedeni elementi prilikom razvijanja i implementacije rješenja za nadzor MPLS VPN mreža vrlo je važno postojanje mogućnosti prezentacije prema korisniku. Važno je da korisnik ima jednostavan uvid u stanje svoje mreže i parametre na mrežnim uređajima. Postoji mogućnost slanja grafova korisniku na dnevnoj, tjednoj i mjesečnoj razini, ovisno o zahtjevima korisnika. Također moguća je implementacija dodatnog besplatnog

[25] FTP - je standardni mrežni protokol koji se koristi za premještanje datoteka s jednog računala na drugo putem mreže.
[26] HTTP - protokol aplikacijske razine OSI modela, osnovna namjena ovog protokola je omogućavanje objavljivanja i prezentacije HTML dokumenata, tj. web stranica.
[27] HTTPS - je protokol nastao kombinacijom protokola HTTP s protokolom SSL/TLS.
[28] LDAP - aplikacijski protokol za čitanje i pisanje imenika preko IP mreže.
[29] MySQL - sustav za manipuliranje relacijskim bazama podataka.
[30] SMTP - protokol za prijenos elektroničke pošte na Internetu.
[31] SSH - mrežni protokol koji omogućuje uspostavu sigurnog komunikacijskog kanala između dva računala putem računalne mreže.
[32] Telnet - mrežni protokol koji omogućava uspostavu dvosmjernog 8-bitnog komunikacijskog kanala između dva umrežena računala.
[33] ICMP - je mrežni protokol koji mrežni uređaji koriste za dijagnosticiranje problema mrežne komunikacije.

rješenja integriranog s rješenjem za nadzor, koje je moguće zamisliti kao posebno korisničko sučelje gdje se prezentiraju podaci za pojedinog korisnika o stanju mreže i mrežnih uređaja prikupljenih pomoću alata za nadzor. Ukoliko je potrebno moguće je da sustav za prezentaciju bude *„Multi-tenant“*, što znači da jedan sustav može koristiti više firmi, tj. korisnika. Unutar takvog sustava moguće je kreirati više organizacija za svaku firmu odvojeno kako bi se odvojili podaci svake firme od drugih, postigla veća preglednost, a samim time i osigurala zaštita podataka. Postoji mogućnost kreiranje više korisnika, tj. korisničkih računa unutar jedne organizacije pomoću RBAC *(*engl. *Role-based access control)*[34] funkcije za ograničavanje pristupa sustavu. Pojedini korisnici imaju mogućnost uređivanja sučelja, dok određeni imaju samo mogućnost pregleda. Također, određeni korisnici mogu biti ograničeni samo na određene prikaze. Prezentacijsko rješenje mora biti skalabilno tako da postoji mogućnost implementacije na više načina. Ovisno o zahtjevima korisnika moguća je implementacija rješenja na jednom serveru za više različitih korisnika ili jedan server samo za jednog korisnika. Također, prilikom implementacije rješenja za prezentaciju potrebno je voditi brigu o sigurnosti, [5], [7].

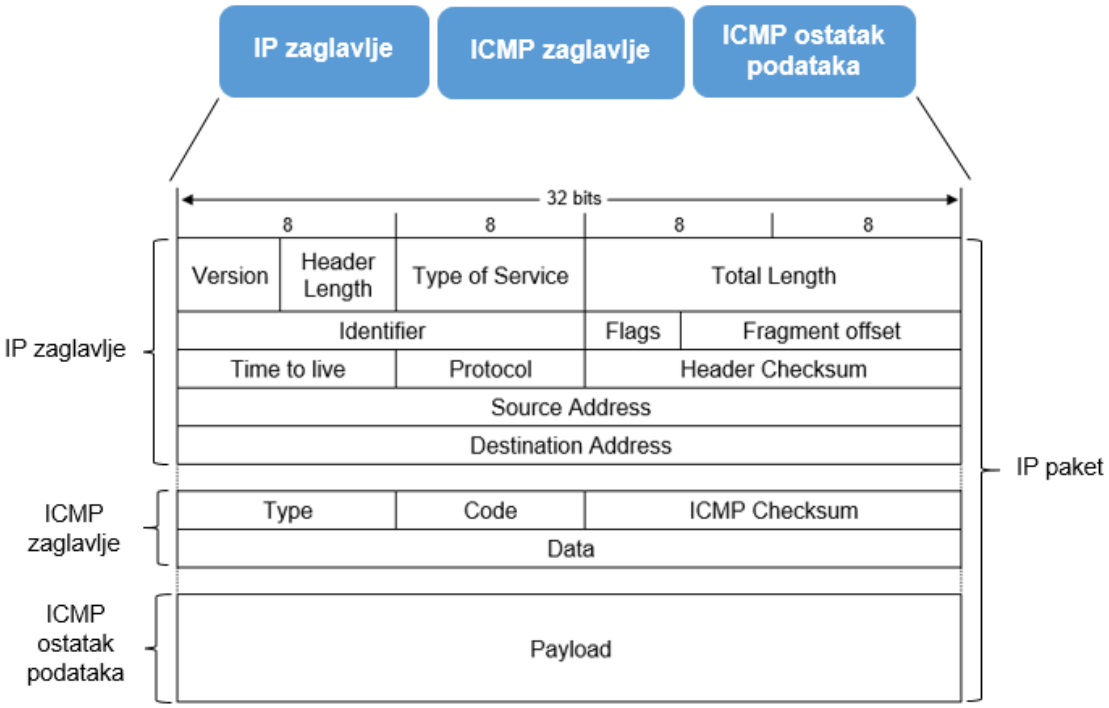
[34] RBAC - sigurnosna kontrola pomoću koje se korisnicima daje pravo pristup ovisno o njihovoj ulozi u organizaciji.

4. PROTOKOLI ZA NADZOR MPLS MREŽA

Za omogućavanje komunikacije između mrežnih uređaja kao i za obavljanje nadzora potrebno je koristiti pojedine protokole. U navedenom poglavlju bit će opisani samo određeni protokoli, a to su: ICMP, TELNET, SSH i SNMP. ICMP se koristi za provjeru dostupnosti uređaja, TELNET i SSH za spajanje na usmjerivače i postavljanje potrebne konfiguracije, te SNMP za prikupljanje podataka sa mrežnih uređaja na temelju kojih će se crtati grafovi prometa i paketa.

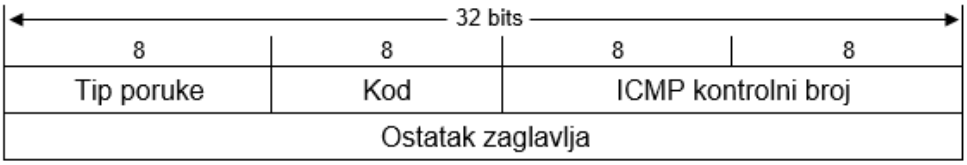
4.1. Internet Control Message Protocol

ICMP (engl. *Internet Control Message Protocol*) je protokol mrežne razine i sastavni dio IP protokola, iako se ponaša kao protokol više razine šaljući svoje poruke preko IP protokola. Definiran je RFC-om 792. Osnovna namjena ICMP protokola je osigurati nadzor i kontrolu prijenosa podataka do odredišta, s obzirom da to IP protokol ne osigurava. ICMP protokol šalje poruke koje osiguravaju: kontrolu toka, prijavu pogreške, pojavu alternativnog puta do odredišta i druge informacije namijenjene samoj TCP/IP programskoj podršci. Time nije osiguran pouzdani prijenos podataka, to treba osigurati protokol više razine. Poruke se šalju samo kao odgovor na poslane IP pakete, na poslane ICMP pakete odgovor se ne šalje. U slučaju gubitka ICMP poruke, ne generira se nova ICMP poruka o nastaloj pogrešci. Mrežni uređaji, uključujući usmjerivače *(*engl. *router)*, koriste ICMP za slanje poruka o pogrešci i operativnih informacija koje upućuju na to da tražena usluga, računalo ili usmjerivač nisu dostupni.



Slika 8 Struktura ICMP paketa
Izvor: [12]

ICMP poruka se šalje unutar IP paketa. Na slici 8 je prikazana struktura IP paketa unutar kojega je enkapsulirano ICMP zaglavlje. IP paket se sastoji od IP zaglavlja, ICMP zaglavlja i ICMP ostatka podataka. Prvi oktet polja podataka IP paketa definira tip ICMP poruke, čime je određen format ostatka paketa. Vrijednost polja *"Protocol"* za ICMP poruku je 1. Svaka ICMP poruka sadrži i IP zaglavlje poruke o čijem gubitku izvješćuje te prvih 64 bita podataka originalnog paketa.



Slika 9 Prikaz formata ICMP zaglavlja

Na slici 9 je prikazan format ICMP zaglavlja. ICMP zaglavlje se sastoji od sljedećih polja:

- Tip ICMP poruke *(*engl. *Type)* - poruke i njihova značenja navedena su u nastavku
- Kod *(*engl. *Code)* - svaki tip poruke je definiran određenim kodom
- ICMP kontrolni zbroj paketa *(*engl. *Checksum)*
- Podaci i ostatak zaglavlja *(*engl. *Data)*

ICMP generira osam različitih tipova poruka, od kojih tri zahtijevaju odgovor. U tablici 1 su prikazani različiti tipovi poruka i objašnjeno je njihovo značenje.

Tipovi poruka	Objašnjenje poruka
Odredište nedostupno *(*engl. *Destination Unreachable)*	Šalje se kad nije moguće uspostaviti vezu ili pronaći put do odredišnog računala, kao i u slučaju kad odredišno računalo ne može prepoznati koja se usluga od njega potražuje, tj. ne prepoznaje priključnu točku usluge *(*engl. *port).* Ako je nedostupna mreža ili računalo, poruku šalje usmjerivač, a ako nije prepoznata priključna točka onda ju šalje odredišno računalo.
Istek vremena *(*engl. *Time Exceeded)*	Šalje se kad je paket odbačen jer je polje "TTL" postalo jednako nuli. Koristi se za određivanje puta kroz mrežu.
Problem s parametrima *(*engl. *Parametar Problem)*	Poruku generiraju usmjerivač ili odredišno računalo kad paket treba biti odbačen jer zbog problema s parametrima u zaglavlju ne mogu završiti obradu paketa.
Blokiranje izvorišta *(*engl. *Source Quench)*	Generira se kad paketi stižu brže nego što ih odredište može obraditi pa usmjerivač ili odredišno računalo šalju izvorištu ICMP poruku za privremeni prekid slanja paketa.
Preusmjeravanje *(*engl. *Redirection)*	ICMP poruka koju šalje usmjerivač kad u svojoj tablici usmjeravanja nađe bolji put do odredišta. Drugi usmjerivač mora se nalaziti u istoj mreži.
Echo zahtjev / echo odgovor *(*engl. *Echo Request / Echo Reply)*	Par poruka kojima se saznaje je li odredište aktivno. Adrese izvorišta i odredišta zahtjeva zamjene mjesta u odgovoru. Ove poruke koristi naredba *ping*.
Vrijeme / odgovor o vremenu *(*engl. *Timestamp / Timestamp Reply)*	Šalju se kad je potrebno saznati za koje vrijeme se poruka preko odredišta vrati do izvorišta *(RTT - Round Trip Time).*
Zahtjev za informacijom / odgovor na zahtjev za informacijom *(*engl. *Information Request / Information Reply)*	Koriste se za poznavanje adrese vlastite mreže.

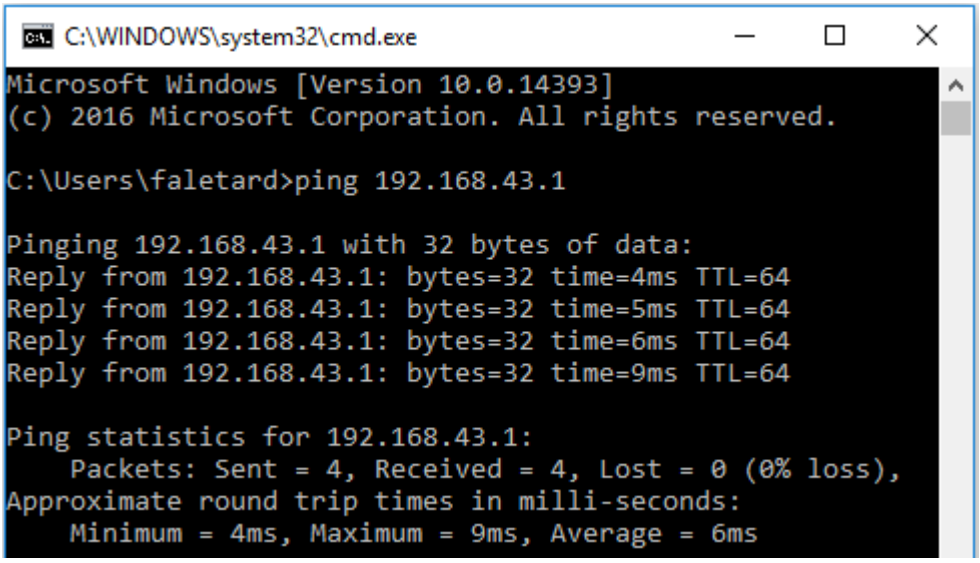
Tablica 1 Prikaz različitih tipova ICMP poruka

Izvor: [11]

Mrežni administratori uglavnom koriste ICMP protokol za istraživanje i rješavanja mrežnih problema. Dvije glavne naredbe za rješavanje mrežnih problema koje obuhvaća ICMP protokol su: *„ping“* i *„traceroute“*, [4], [11].

4.1.1. Ping

Jedan od najpoznatijih primjera korištenja ICMP paketa je naredba *ping*. Njenom uporabom, uz ispravno postavljanje potrebnih parametara, moguće je ocijeniti povezanost dvaju računala na Internetu. Također, *ping* se koristi i za prikupljanje statističkih podataka, tzv. *„round trip“* vremena potrebnog da paket ode do određenog računala na Internetu i natrag, broja neuspješnih odgovora itd. Pokretanjem naredbe uz određivanje odredišnog računala zadavanjem njegove IP adrese, odašilje se ICMP *„echo request“* poruka. Ukoliko je odredišno računalu aktivno, od njega dolazi odgovor u obliku poruke ICMP *„echo reply“*. U protivnom, nakon isteka određenog vremena, odnosno nedobivanja odgovora, računalo se smatra nedostupnim.



Slika 10 Prikaz izvođenja ping naredbe

Svaki usmjerivač prilikom prosljeđivanja svakog paketa umanji vrijednost parametra TTL (engl. *Time To Live*) za jedan. Ukoliko vrijednost tog parametra dosegne nulu, stvara se poruka tzv. ICMP *„Time to live exceeded in transit“* i odašilje prema izvorištu izvorne poruke. Primjer izvođenja *ping* naredbe za aktivno računalo dano je na slici 10 (upit se izvodi s računala pokretanog Windows operacijskim sustavom).

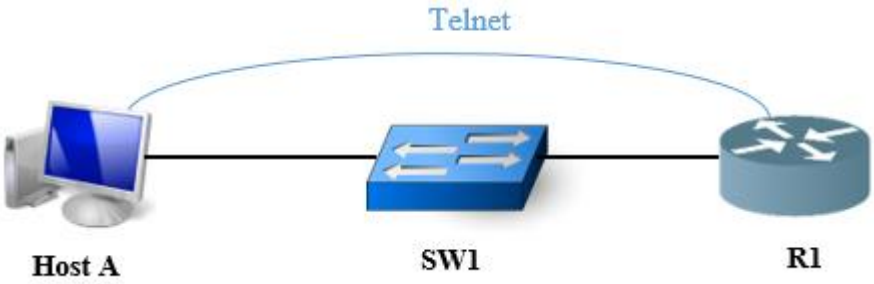
4.1.2. Traceroute

Druga korisna primjena ICMP protokola je u izvedbi naredbe *traceroute*, korištene za određivanje povezanosti dvaju računala na mreži, ali s tim da daje informacije i o svim računalima koja se nalaze na putu od izvorišta do odredišta. Ona prvo šalje paket ICMP *„echo request“* s postavljenom vrijednošću TTL parametra na 1. Prvi čvor na kojeg naiđe ta poruka umanjuje vrijednost TTL parametra na nulu, zbog čega dolazi do odbacivanja paketa i slanja odgovarajućeg ICMP *„Time to live exceeded in transit“* odgovora. Sljedeći koraci uključuju slanje *„request“* paketa s vrijednošću TTL parametra postavljenom na 1, kako bi se ustanovilo koliko je različitih računala u neposrednoj blizini izvorišta, odnosno slanjem paketa uz postupno uvećavanje parametra za 1. Postupak se ponavlja sve do nailaska na odredišno računalo. [4]

4.2. Telnet

Telnet je aplikacijski protokol koji omogućava korisniku komunikaciju s udaljenim uređajem. Najčešće se koristi da osigura korisniku jednog računala sesiju za korištenje CLI-a[35] tzv. sučelja komandne linije *(*engl. *command-line interface)* na drugom računalu ili uređaju. Telnet često koriste mrežni administratori za pristup i upravljanje udaljenim uređajima. Za pristup udaljenom uređaju, mrežni administratori koriste naredbu *telnet* te IP adresu ili naziv uređaja *(*engl. *hostname)*. Nakon uspješnog spajanja i pristupa udaljenom uređaju, mrežni administrator pomoću virtualnog terminala ima mogućnost upravljanja i komuniciranja s udaljenim uređajem. Telnet koristi model telnet klijenta (aplikacija terminala) i telnet poslužitelja (usmjerivač u ovom slučaju). Telnet klijent, tj. uređaj/računalo koji koje korisnik koristi, prihvaća unos tipkovnice i šalje naredbe na telnet poslužitelj. Telnet poslužitelj osluškuje zahtjeve klijenata na TCP[36] priključku 23.

[35] CLI - sučelje naredbene linije pomoću kojeg korisnik zadaje tekstualne naredbe i instrukcije koje operativni sustav treba izvršiti.
[36] TCP - protokol transportnog sloja, uspostavlja vezu prije slanja podataka i garantira pouzdanu isporuku podataka od izvorišta do odredišta u kontroliranom redoslijedu.



Slika 11 Shematski prikaz povezivanja pomoću telnet protokola, [4]

Na slici 11 je prikazano računalo (Host A) kojemu je pomoću *telnet* protokola omogućen pristupit i upravljanje usmjerivačem (R1). Na računalu (Host A) se pokreće *telnet* klijent program i unosi se IP adresa udaljenog usmjerivača (R1).

```
PC_A#telnet 192.168.10.2
Trying 192.168.10.2 ... Open

User Access Verification

Username:
```

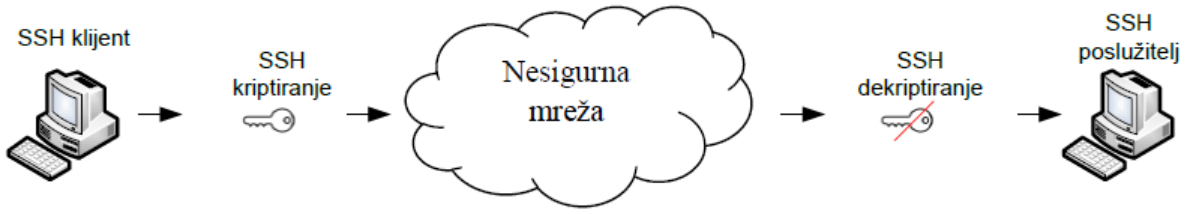
Slika 12 Prikaz pristupa udaljenom usmjerivaču pomoću telnet-a

Na slici 12 je prikazan postupak pristupa udaljenom usmjerivaču pomoću *telnet-a*. Nakon što se ostvari pristup usmjerivaču, potrebno je unijet lozinku za pristup. Naredbe i podaci se šalju u obliku čistog teksta kojeg *telnet* poslužitelj prihvaća i uzvraća odgovorom. Telnet protokol je jednostavan za korištenje, iako se više ne koristi tako često u produkcijskom okruženju. Ne koristi se iz razloga što je nije siguran protokol, tj. upravo zbog toga što se svi podaci i naredbe šalju u obliku čistog (ne kriptiranog) teksta, čak i lozinke što narušava sigurnosne aspekte, [4].

4.3. SSH

SSH *(*engl. *Secure Shell)* protokol je nastao 90-ih godina prošlog stoljeća kao zamjena za druge, nesigurne protokole, poput *rlogin, rsh, telnet* i FTP koji putem računalne mreže razmjenjuju podatke. SSH za razliku od postojećih protokola uvodi zaštitu tajnosti podataka. Naime, kod drugih sličnih protokola podaci se kroz mrežu šalju u otvorenom (ne kriptiranom) obliku i bilo koji korisnik može ih presresti, pročitati ili čak mijenjati. SSH podatke kriptira

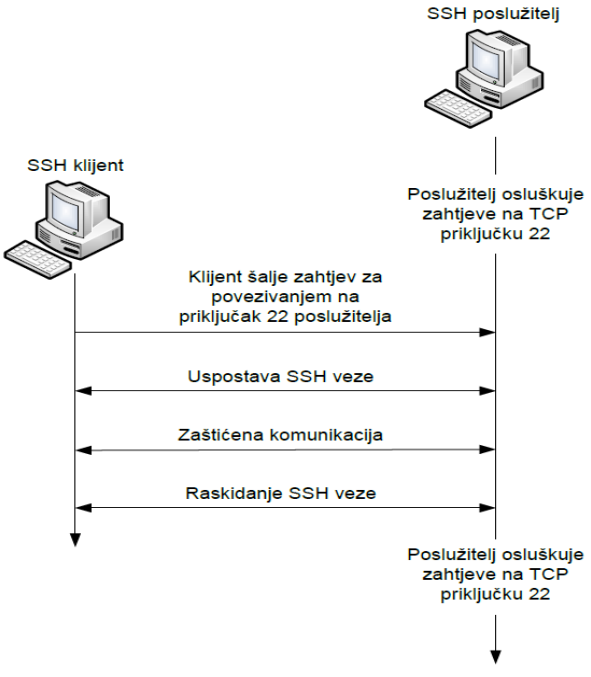
prije slanja i dekriptira nakon primitka čime se onemogućuje njihovo otkrivanje dok se kreću mrežom. Na slici 13 je prikazan tijek SSH komunikacije.



Slika 13 Prikaz tijeka SSH komunikacije, [13]

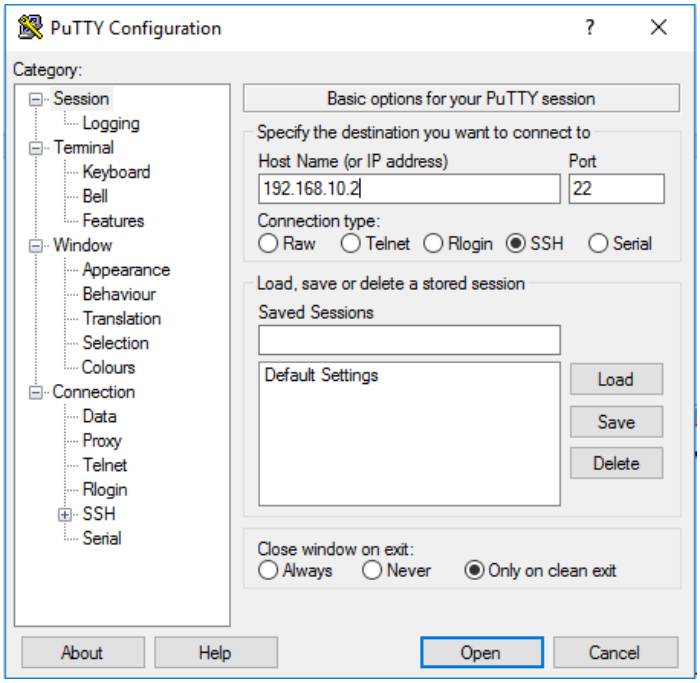
SSH se temelji na modelu klijent/poslužitelj što znači da se komunikacija odvija između dvije različite strane. Na slici 14 je prikazana shema modela klijent poslužitelj. Poslužitelj s jedne strane osluškuje zahtjeve na unaprijed zadanom mrežnom priključku *(*engl. *port),* a klijent ih po potrebi šalje poslužitelju. SSH poslužitelj osluškuje zahtjeve klijenata na TCP priključku 22. Uspostava komunikacije i sama komunikacija u SSH protokolu može se opisati troslojnom arhitekturom:

1. Transportni sloj *(*engl. *Transport Layer Protocol)* - dogovaranje kriptografskih algoritama i parametara, razmjena ključa, autentifikacija poslužitelja;
2. Autentifikacijski sloj *(*engl. *Authentication Protocol)* - autentifikacija klijenta i
3. Spojni sloj *(*engl. *Connection Protocol)* - udaljeno pokretanje naredbi i udaljena prijava na sustav, prosljeđivanje TCP prometa.



Slika 14 Shema SSH modela klijent/poslužitelj, [13]

Na slici 15 je prikazano računalo (Host A) koje će pomoću SSH protokola pristupiti i upravljati usmjerivačem (R1). Na računalu (Host A) se pokreće SSH klijent program (u ovome slučaju, klijent program je *Putty*) i unosi se IP adresa udaljenog usmjerivača (R1).



Slika 15 Prikaz SSH povezivanja na udaljeni usmjerivač pomoću Putty-a

Prilikom uspostave komunikacije i udaljenog pristupanja pomoću SSH protokola, računalo (klijent) i usmjerivač (server) će proći kroz proces opisan troslojnom arhitekturom. Nakon što dogovore i usklade sve parametre, računalo će imati pristup udaljenom usmjerivaču.

Osim spajanja na udaljeni poslužitelj i udaljenog izvođenja naredbi, SSH protokol se može koristiti i za sigurnosno poboljšane usluge mrežne komunikacije. Među rješenjima su najčešće sigurnosno poboljšane inačice pojedinih mrežnih protokola i usluga kao na primjer:

• scp – SSH inačica *rcp* naredbe koja kopira datoteke s lokalnog računala na udaljeno. Pritom se podaci šalju kriptirani i zaštićeni.

• sftp – SSH inačica FTP *(*engl. *File Transfer Protocol)* protokola kojim se datoteke prenose između računala i

• sshfs *(*engl. *SSH Filesystem)* – protokol za siguran rad s datotečnim sustavom udaljenog računala, [13].

4.4. Simple Network Management Protocol

SNMP je protokol aplikacijskog sloja koji olakšava razmjenu informacija o upravljanju između mrežnih uređaja, kao što su: čvorovi, usmjerivači, preklopnici i dr. Kao dio skupa TCP/IP protokola, SNMP omogućava administratorima udaljeni nadzor i upravljanje mrežnim performansama, pronalaženje i rješavanje mrežnih problema te planiranje potreba za proširivanjem mreže. SNMP je mrežni upravljački protokol dizajniran tako da olakša upravljanje i nadzor kompletne mreže te svih njenih entiteta. Funkcionalnost i implementacija SNMP protokola je relativno jednostavna no ipak dovoljno fleksibilna da pruži mogućnost kvalitetnog upravljanja velikim brojem različitih tipova uređaja u današnjoj distribuiranoj mrežnoj okolini. Do sada su se pojavile tri verzije SNMP protokola: SNMPv1, SNMPv2 i SNMPv3.

4.4.1. SNMPv1

SNMPv1 se koristi od 1988. godine i prihvaćen je kao jedan od standarda TCP/IP modela. Sigurnost se kod SNMPv1 temelji na korištenju takozvanih zajedničkih znakovnih nizova *(*engl. *community string),* a to je zapravo niz ASCII znakova. Koristi se za autentifikaciju SNMP poruka između upravljačke jedinice i upravljanog uređaja. Najveći sigurnosni problem je u tome što neovlašteni korisnici mogu snimanjem IP paketa koji se prenose mrežom pročitati sadržaj SNMP poruka i saznati zajednički znakovni niz pošto se prenosi u čitljivom (ne kriptiranom) obliku. Ukoliko se otkrije zajednički znakovni niz, napadači mogu pristupiti upravljačkim informacijama nekog mrežnog uređaja te čak promijeniti njegovu konfiguraciju.

4.4.2. SNMPv2

SNMPv2 protokol je uveden u TCP/IP mrežama 1993. godine. Ponudio je neka proširenja kao što su dodatne operacije i korištenje *„party-based“* sigurnosti koja se pokazala veoma složenom u svakodnevnoj upotrebi. Zbog toga SNMPv2 nije nikada zaživio u upravljanim mrežama. Umjesto njega je 1995. godine prihvaćen kao standard u TCP/IP mrežama *„Community-Based Simple Network Management Protocol version 2“* - SNMPv2c. Kao što mu i samo ime kaže, SNMPv2c protokol svoju sigurnost zasniva također na

zajedničkim znakovnim nizovima tako da su sigurnosni problemi ostali isti kao i kod SNMPv1 protokola. SNMPv2 podržava tri načina pristupa upravljačkoj informaciji, dok SNMPv1 podržava prvi i treći, a to su sljedeći:

1. Upravljač-agent: SNMPv2 upravljač šalje zahtjev agentu, a agent odgovara slanjem traženih upravljačkih informacija korištenjem dohvaćanja i modificiranja upravljačkih informacija,
2. Upravljač-upravljač: jedan SNMPv2 upravljač šalje zahtjev drugom upravljaču, a drugi odgovara slanjem traženih upravljačkih informacija i
3. Agent-upravljač: SNMPv2 agent šalje poruku "trap" upravljaču.

4.4.3. SNMPv3

SNMPv3 protokol je uveden u TCP/IP mrežama 1998. godine i posjeduje bitno poboljšane sigurnosne mehanizme. SNMPv3 posjeduje mehanizme za sigurnu kontrolu pristupa, autentikaciju, odnosno provjeru vjerodostojnosti korisnika, te enkripciju, odnosno zaštitno kodiranje SNMP poruka. SNMPv3 može koristiti takozvanu korisničku *(*engl. *user-based)* autentikaciju na temelju korisničkog imena i lozinke ili se provjera vjerodostojnosti korisnika može obaviti bez slanja lozinki u čitljivom obliku, a temelji se na upotrebi algoritama HMAC-MD5[37] *(*engl. *Hash-based Message Authentication Code- Message-Digest algorithm 5)* ili HMAC-SHA[38] *(*engl. *Hash-based Message Authentication Code -Secure Hash Algorithm)*. Zaštitna enkripcija koristi 56-bitni CBC-DES[39] *(*engl. *Cipher Block Chaining-Data Encryption Standard)* algoritam za kodiranje i dekodiranje SNMP poruka.

[37] HMAC-MD5 - algoritam za kriptiranje i provjeru autentičnosti poruke koji koristi kriptografsku hash funkciju MD5.
[38] HMAC-SHA - algoritam za kriptiranje i provjeru autentičnosti poruke koji koristi kriptografsku hash funkciju SHA.
[39] CBC-DES - algoritam za kodiranje i dekodiranje SNMP poruka u SNMPv3.

4.4.4. Arhitektura SNMP protokola

Funkcija SNMP protokola je prikupljanje i organiziranje primljenih informacija o stanju računalne mreže. SNMP protokol mrežnom administratoru omogućuje nadgledanje performansi te pronalaženje i rješavanje mrežnih problema. Mreža koja se upravlja SNMP-om sastoji se od nekoliko ključnih elemenata:

- Sustava upravljanja mrežom - NMS *(*engl. *network management system NMS),*
- upravljanih mrežnih uređaja,
- agenata,
- baze upravljačkih informacija - MIB *(*engl. *Management Iniformation Base)*[40]

Sustav upravljanja mrežom predstavlja računalo ili poslužitelj na kojem se pokreće aplikacija za nadzor, tzv. SNMP upravitelj *(*engl. *manager)* koja nadzire ili upravlja elementima upravljanih mrežnih uređaja. Također, sustav upravljanja mrežom, osigurava većinu resursa i memorije potrebne za upravljanje mrežom. NMS pruža jedinstveno sučelje pomoću kojega mrežni administratori mogu postaviti pojedine uvjete i izdvojiti dio mrežnih performansi koje žele nadzirati te primati automatska upozorenja i obavijesti ukoliko dođe do nepravilnosti u radu. Sustav upravljanja mrežom često se naziva i upravljačka stanica.

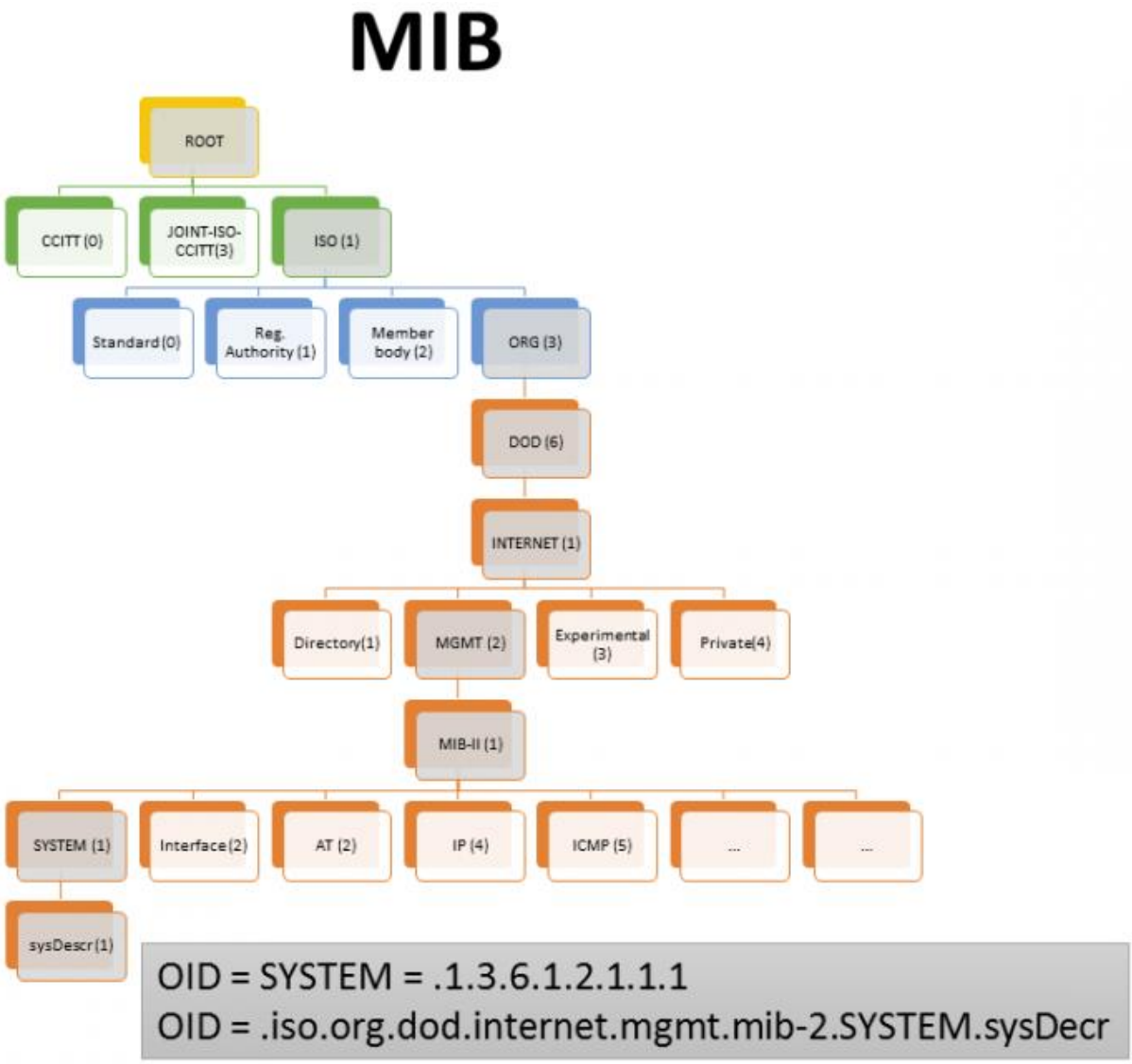
Upravljani mrežni uređaji predstavljaju sve mrežne uređaje koje se nalaze u mreži koja se nadzire. To mogu biti: preklopnici, usmjerivači, poslužitelji, računala i svi IoT *(*engl. *Internet of Things)* uređaji koji poznaju SNMP protokol.

SNMP agent predstavlja programski modul koji je instaliran na upravljanim mrežnim uređajima, a služi za prikupljanje različitih mjernih podataka poput: prostora na disku, opterećenosti procesora, propusnosti linka, dostupnosti uređaja i dr. Prikupljene podatke koji su strukturirani kao upravljani skup objekata, agenti spremaju u baze upravljačkih informacija, tzv. MIB (koristit će se taj naziv u nastavku). SNMP agent omogućava komunikaciju između nadziranih uređaja i SNMP upravitelja. Osim što odgovara na upite upravitelja i šalje tražene podatke sustavu za upravljanje mrežom, SNMP agent može proaktivno reagirati i obavijestiti sustav za upravljanje mrežom ukoliko dođe do pogreške.

[40] MIB - upravljačka informacijska baza koja sadrži definirane podatke o svojstvima objekata.

MIB je kratica za upravljačku informacijsku bazu i predstavlja skup definicija koje definiraju svojstva upravljanog objekta unutar uređaja kojim se upravlja. MIB-ovi sadrže informacije o konfiguraciji mrežnih komponenti, poput verzije softvera koji se pokreće na komponenti, IP adrese ili broja priključka, te o količini dostupnog prostora na disku za pohranu. MIB-ovi funkcioniraju kao vrsta direktorija koji sadrži logička imena mrežnih resursa i njihovih konfiguracijskih parametara kojima upravlja SNMP. Upravljački parametri definirani su kao upravljani objekti, a predstavljaju upite koje SNMP upravitelj postavlja SNMP agentu. Manji sastavni dijelovi koji čine MIB-a nazivaju se OID-ovi. OID *(*engl. *Object Identifier)*[41] ili identifikatori objekata jedinstveno identificiraju upravljane objekte unutar MIB-a. OID točno identificira vrijednost koja se treba prikazati *(*engl. *get)* ili koju treba postaviti *(*engl. *set)*. MIB je organiziran hijerarhijski i može se prikazati kao stablo *(*engl. *tree)* s različitim razinama od korijena *(*engl. *root)* do pojedinih čvorova. Općenito, OID je dugi niz brojeva koji označavaju čvorove i razdvojeni su točkicama. Svaki OID ima zapis koji se čita s lijeva na desno i prati se struktura stabla od korijena do njegovih čvorova, [14], [15]. Na slici 16 prikazano je MIB stablo i uzorak strukture OID-a.

[41] OID - identifikatori objekata, jedinstveno identificiraju vrijednosti upravljanih objekata koje treba prikazati ili koju treba postaviti.



Slika 16 Prikaz MIB stabla i uzorka strukture OID-a, [15]

Prikazani OID 1.3.6.1.2.1.1.1 služi za dohvat tekstualnog opis entiteta, tj. uređaja *(*engl. *sysDescription)* na kojemu se obavlja nadzor. Opis uključuje puni naziv i identifikaciju verzije i vrste hardvera sustava, operacijski sustav softvera, i mrežni softver. Povratne informacije koje vraća navedeni OID prikazane su u nastavku:

SNMPv2-MIB::sysDescr.0 = STRING: Cisco IOS Software, ME340x Software (ME340x-METROIPACCESSK9-M), Version 12.2(60)EZ14, RELEASE SOFTW

4.4.5. Odvijanje komunikacije i razmjena SNMP poruka

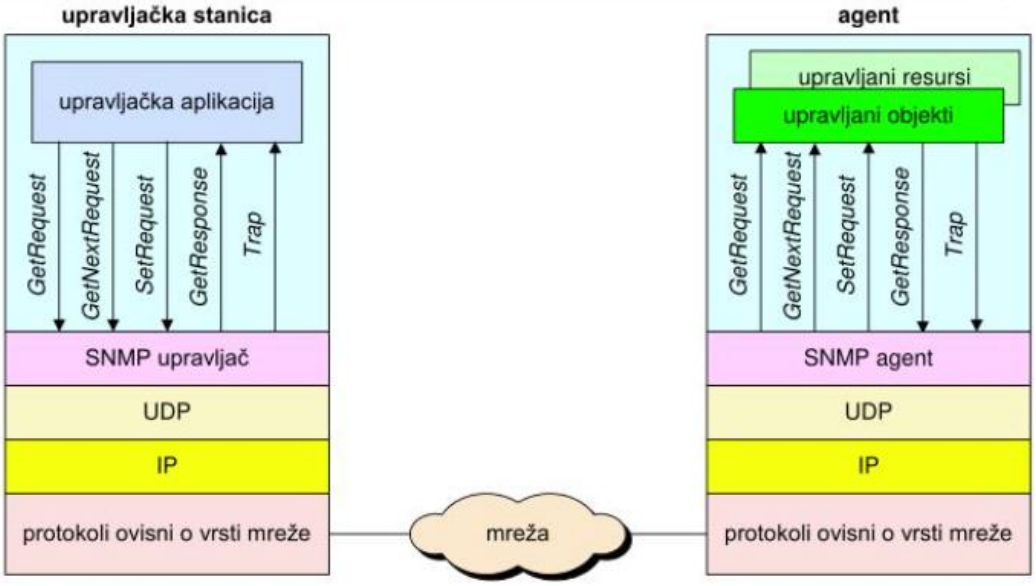
Komunikacija pomoću SNMP protokola odvija se po modelu klijent-poslužitelj. Ulogu klijenta posjeduje upravljačka stanica koja šalje zahtjeve, dok ulogu poslužitelja obnašaju upravljani mrežni uređaji koji odgovaraju na postavljene zahtjeve. Komunikacija između klijenta i poslužitelja odvija se slanjem pojedinih poruka: *get, set, response, trap*.

Poruku *get (dohvati)* upravljačka stanica šalje agentu u određenim vremenski intervalima koju su definirani unutar upravljačke aplikacije. Slanjem poruke *get* upravljačka stanica dohvaća vrijednosti upravljanih objekata sadržanih u MIB-ovima agenata.

Poruka *set (postavi)* omogućava upravljačkoj stanici postavljanje i izmjenu vrijednosti upravljanih objekata u MIB-ovima agenata. Za postavljanje vrijednosti upravljačka stanica se javlja agentu s OID oznakom i pripadajućim inačicama te novom vrijednosti. Agent prosljeđuje zahtjev i dodjeljuje nove vrijednosti unutar MIB-a.

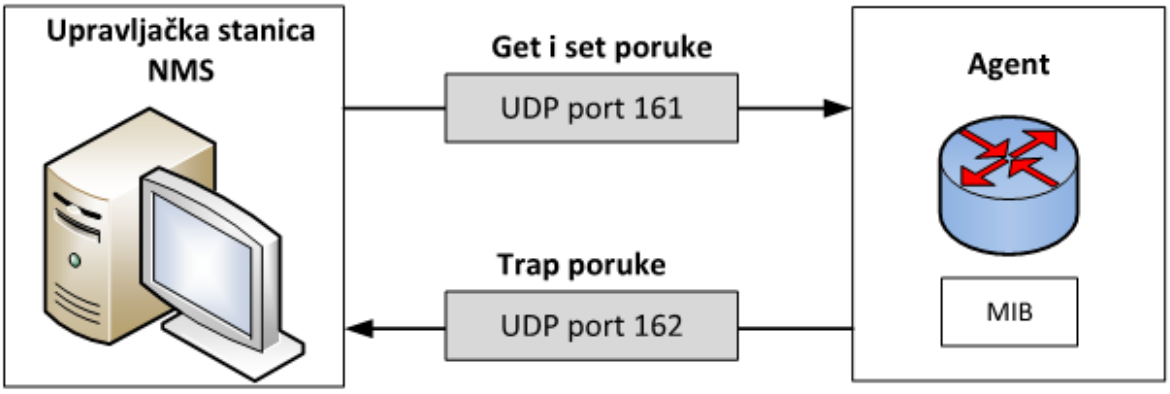
Poruku *response (odgovor)* šalje agent prema upravljačkoj stanici kao odgovor na *get* poruke. Poruka *response* sadrži vrijednosti traženih varijabli koje se dostavljaju upravljačkoj stanici.

Poruku *trap (filter za praćenje)* šalje agent kao upozorenje ili obavijest upravljačkoj stanici bez ranijeg zahtjeva. Poruke *trap* omogućavaju proaktivan nadzor, a šalju se pod određenim uvjetima, kao što je u slučaju nastanka pogreške na mreži ili prelaska unaprijed postavljenog praga (npr. temperature, memorije i dr.). Potrebno je ranije kreirati *trap* pomoću SNMP upravitelja na upravljačkoj stanici.



Slika 17 Prikaz razmjene SNMP poruka između upravljačke stanice i agenta, [14]

Na slici 17 prikazana je razmjena SNMP poruka između upravljačke stanice i agenta. Upravljačka aplikacija šalje agentima poruke „*GetRequest*“, „*GetNextRequest*“ i „*SetRequest*“. Primitak bilo koje od tih poruka agent potvrđuje slanjem poruke „*GetResponse*“ upravljačkoj stanici. Poruku „*Trap*“ agent šalje upravljačkoj stanici kao reakciju na događaj koji utječe na sadržaj MIB-a agenta ili na njegove upravljane resurse u podlozi MIB-a.



Slika 18 Prikaz slanja osnovnih poruka (get i set) između upravljačke stanice i agenta

Izvor: [14]

Za prijenos navedenih SNMP poruka, SNMP koristi transportni protokol UDP *(*engl. *User Datagram Protocol)*[42]. UDP je beskonekcijski protokol što znači da se prilikom komunikacije između upravljačke stanice i agenata ne uspostavlja konekcija. Također, uređaj koji primi UDP datagram ne potvrđuje prijem pošiljatelju čime se ubrzava prijenos. Prema zadanim postavkama SNMP protokol primarno koristi dva UDP porta. Na slici 18 je vidljivo da se prilikom slanja osnovnih poruka (*get* i *set*) koristi se UDP port 161, a za slanje trap poruka UDP port 162, [14].

[42] UDP - protokol transportnog sloja, ne uspostavlja vezu prije slanja, ne garantira isporuku podataka u cijelosti.

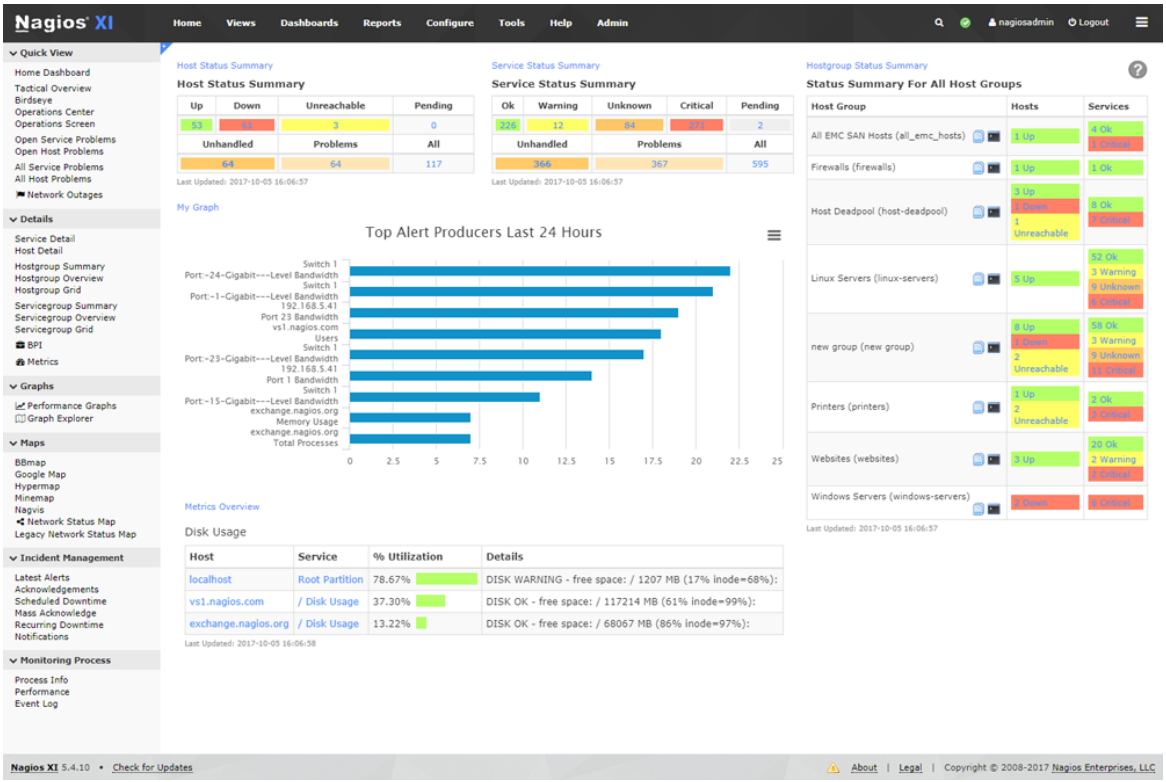
5. PROGRAMSKI ALATI ZA NADZOR MPLS MREŽA

Razvojem postojećih tehnologija i pojavom novih tehnologija kao što su računalni oblak, bežični i udaljeni pristup, virtualne privatne mreže, mobilnih uređaja itd., računalne mreže su napredovale i postale sveprisutne u današnjim sustavima. Danas više ne postoji grana industrije koja nije obuhvaćena informatičkom tehnologijom. Budući da komunikacijske mreže i mrežni uređaji imaju sve veću ulogu u svim sustavima (poslovnim, obrazovnim, zdravstvenim, vojnim, itd.) trebalo je pronaći rješenje kako nadzirati sve elemente sustava, pogotovo ako su oni međusobno veoma udaljeni. Zbog toga su se potkraj 20. stoljeća na tržištu počeli pojavljivati prvi specijalizirani programi za nadzor mrežnih komponenti i uređaja na udaljenim lokacijama preko mreže. Postoji mnoštvo programskih alata za nadzor mreže i mrežnih uređaja, a ukratko će bit opisani sljedeći alati: Nagios, Cacti, Zenos, Zabbix. Navedeni programski alati spadaju među vodeće alate za nadzor mreža prema istraživanja i objavama na Internetu. Prije svega navedeni alati su otvorenog koda i besplatni, nude pregledna grafička sučelja, jednostavni su za implementaciju i korištenje, posjeduju detaljnu dokumentaciju i široko dostupnu aktivnu zajednicu što su samo neki od važnih kriterija koji ih svrstavaju na vodeća mjesta. Ostale mogućnosti i prednosti navedene su u nastavku prilikom opisivanja svakog od alata za nadzor.

5.1. Nagios XI

Nagios je snažan sustav za nadzor koji omogućava firmama i organizacijama identifikaciju i rješavanje problema unutar IT infrastrukture prije nego što dođe do kritičnog utjecaja na poslovne procese, kao i same krajnje korisnike. Nagios je osmislio i razvio Ethan Galstad s grupom programera. Nagios dolazi u dvije verzije: Nagios Core (otvorenog koda i besplatan) i Nagios XI - napredna poslovna *(*engl. *enterprise)* verzija koja se plaća.

Nagios XI je prošireno sučelje i aplikacija poslovne klase koja nadgleda poslovne sustave, mreže i mrežnu infrastrukturu. Služi za nadzor mrežne infrastrukture, uključujući aplikacije, specifične usluge, operacijske sustave, mrežne protokole, mjerne podatke sustava i dr. Nudi proširivo korisničko sučelje, uređivač konfiguracija, napredno izvještavanje, čarobnjake za praćenje, različite dodatke *(*engl. *add-ons)* "trećih strana" koji dodatno proširuju funkcionalnosti sustava. Instalacija i samo postavljanje Nagoisa XI je vrlo jednostavno i gotovo u svega par minuta. Na slici 19 je prikazana nadzora ploča koja predstavlja centralizirano mjesto za pregled važnih i istaknutih informacija o stanju mreže, mrežnih usluga, i ostalih podataka.



Slika 19 Prikaz sučelja Nagios XI alata za nadzor, [16]

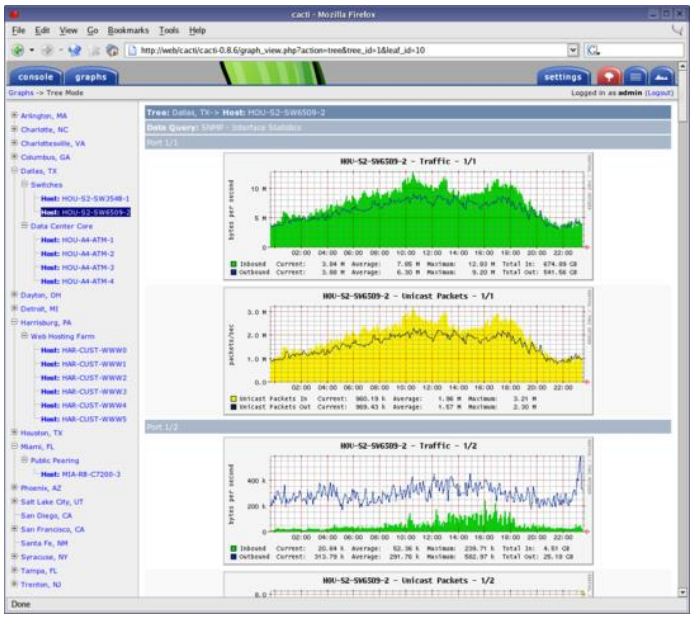
Neke od mogućnosti i prednosti koje Nagios XI pruža su:

- Snažni mehanizam za nadgledanje: Nagios XI koristi snažan Nagios Core 4 mehanizam za praćenje koji korisnicima omogućava učinkovitiji nadzor.
- Ažurirano web sučelje: Vaša nova nadzorna ploča *(*engl. *dashboard)* nudi prilagodljiv pregled stanja mreže, mrežnih uređaja i ostalih usluga.
- Pregledniji grafovi: Olakšava administratorima pregledavanje i otkrivanje mrežnih incidenata i pogrešaka te ubrzavaju rješavanje istih prije nego što dođe do većih kvara i dužih prekida usluga.
- Čarobnjaci za postavljanje konfiguracije: Unosom potrebnih podataka i u svega par klikova omogućava postavljanje i pokretanja nadzora
- Upravljanje infrastrukturom: poboljšani skupni uvoz nadziranih subjekata, automatsko otkrivanje *(*engl. *autodiscovery)*, automatsko uklanjanje
- Automatsko slanje alarma i obavještavanje: ukoliko dođe do kvara ili neželjenih događaja automatski se šalje obavijest mrežnom administratoru putem e-maila, SMS poruke i dr.
- Spremanje konfiguracije: Omogućava arhiviranje konfiguracija i vraćanje istih u željenom trenutku, [16], [17]

5.2. Cacti

Cacti je mrežni nadzorni alat, otvorenog koda *(*engl. *Open source)* koji koristi funkcije RRD alata *(*engl. *Round-robin Database Tool)*. RRD alat je nastao na temelju prijašnjeg programa za nadzor MRTG *(*engl. *Multi Router Traffic Grapher)*, a predstavlja skup uslužnih alata za rad s RRD *(*engl. *Round-robin bazom podataka)*. RRD alat omogućuje spremanje, obradu i grafički prikaz dinamičkih podatka promatranih u određenom vremenskom razdoblju kao što su mrežna propusnost, opterećenje procesora (CPU), temperatura i dr. Projekt Cacti prvi je pokrenuo Ian Berry u rujnu 2001. godine. Cacti je moguće besplatno preuzeti, a uključuje LAMP[43] (Linux, Apache, MySQL, PHP) paket, koji pruža standardiziranu programsku platformu za izradu grafova na temelju prikupljenih podataka koji se spremaju u MySQL bazu podataka.

Cacti sadrži gotove predloške za nadzor poslužiteljskih aplikacija od Linux i Windows poslužitelja do Cisco usmjerivača i preklopnika, u osnovi svih uređaja koji komuniciraju pomoću SNMP protokola. Osim standardne metode za prikupljanje podataka pomoću SNMP protokola, Cacti omogućava korištenje vanjskih skripti pisanih u Perlu ili PHP-u što dodatno proširuje mogućnosti i značajke alata za nadzor. Na slici 20 nalazi se sučelje Cacti alata gdje su grafovi prikazani pomoću prikaza stablo *(*engl. *tree view).*



Slika 20 Prikaz sučelja Cacti alata za nadzor, [19]

[43] LAMP - akronim za skup paketa softvera otvorenog koda: Linux operacijski sustav, Apache web-server, MySQL/MariaDB baza podataka, PHP/Pearl/Python programski jezik.

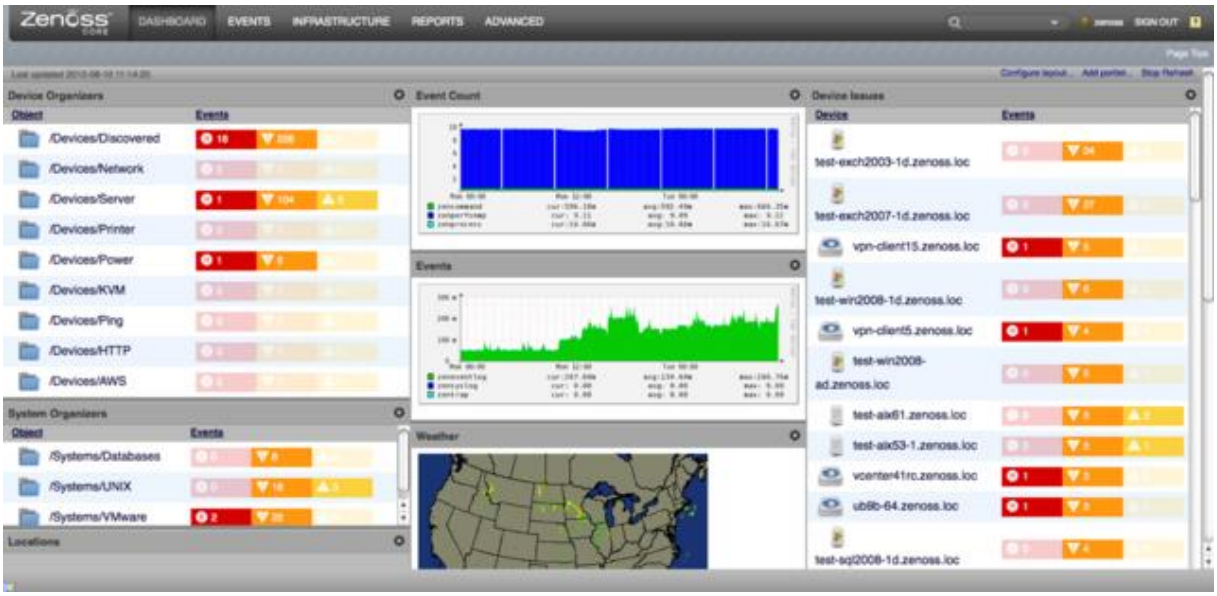
Cacti je potpuno besplatan alat, u usporedbi s ostalim novijim alatima izgleda pomalo zastarjelo i pruža manje mogućnosti, ali idealan je za nadzor mreže i mrežnih uređaja unutar manjih firmi koje nemaju mogućnost izdvajanja financija za nadzor. Neke od mogućnosti koje pruža Cacti:

- Mogućnosti grafova - nudi neograničen broj stavki i opcija koje se mogu definirati za svaki graf, automatsko grupiranje stavki grafa i poravnanje teksta i kazala na grafu, odabir boja, automatsko ubacivanje teksta, mogućnost manipulacije podacima koristeći CDEF matematičke funkcije ugrađene u RRD alatu.
- Grafički prikaz - nudi tri vrste prikaza: prikaz stabla *(*engl. *tree view),* prikaz popisa *(*engl. *list view)* i prikaz pregleda *(*engl. *preview view).* Prikaz stabla omogućava korisnicima stvaranje "hijerarhije grafova" i postavljanje grafova po važnosti. Ovo je jednostavan način organiziranja velikog broja grafova. Prikaz popisa sadrži naslov svakog grafa na jednom velikom popisu koji povezuje korisnika sa stvarnim grafom. Prikaz pregleda prikazuje sve grafove na korisničkom sučelju.
- Predlošci - nudi nekoliko vrsta predložaka: predložak grafa, predložak uređaja, predložak podataka.
- Prikupljanje podataka - osim standardnog prikupljanja podataka pomoću SNMP-a, moguće je pokrenuti vanjske skripte pisane u različitim programskim jezicima
- Upravljanje korisnicima - postoji mogućnost dodavanja korisnika s određenim pravima i razinama pristupa. Pojedini korisnici imaju mogućnost uređivanja grafova i mijenjanja parametara, dok pojedini imaju samo mogućnost pregledavanja grafova, [18], [19].

5.3. Zenoss

Zenoss je jedan od vodećih rješenja za upravljanje i nadzor računalne mreže. Zenoss prati cjelokupnu korisničku fizičku i virtualnu IT infrastrukturu pomoću standardnih protokola kao što su SNMP, WMI i SSH. Zenoss je alat za upravljanje aplikacijama, poslužiteljima i mrežom, a temelji se na poslužiteljskoj aplikacija Zope. Dostupne su dvije verzije Zenoss-a: Zenoss Core i Zenoss Enterprise (poslovna verzija). Zenoss Core je besplatna aplikacija otvorenog koda s web sučeljem koja se može koristiti za praćenje performansi, konfiguracije, mrežnih događaja i dostupnosti mrežnih uređaja. IT administratori mogu koristiti dodatnu arhitekturu zvanu ZenPacks, koju je kreirala organizacija Zenoss za proširenje funkcionalnosti.

Zenoss nudi i poslovnu verziju *(*engl. *enterprise)*, koja uključuje podršku i dodatne značajke kao što su: proširene biblioteke izvještaja, sintetičke web transakcije, globalne nadzorne ploče i certificirane aplikacije za nadzor. Na slici 21 nalazi se prikaz nadzorne ploče Zenoss sustava za nadzor.



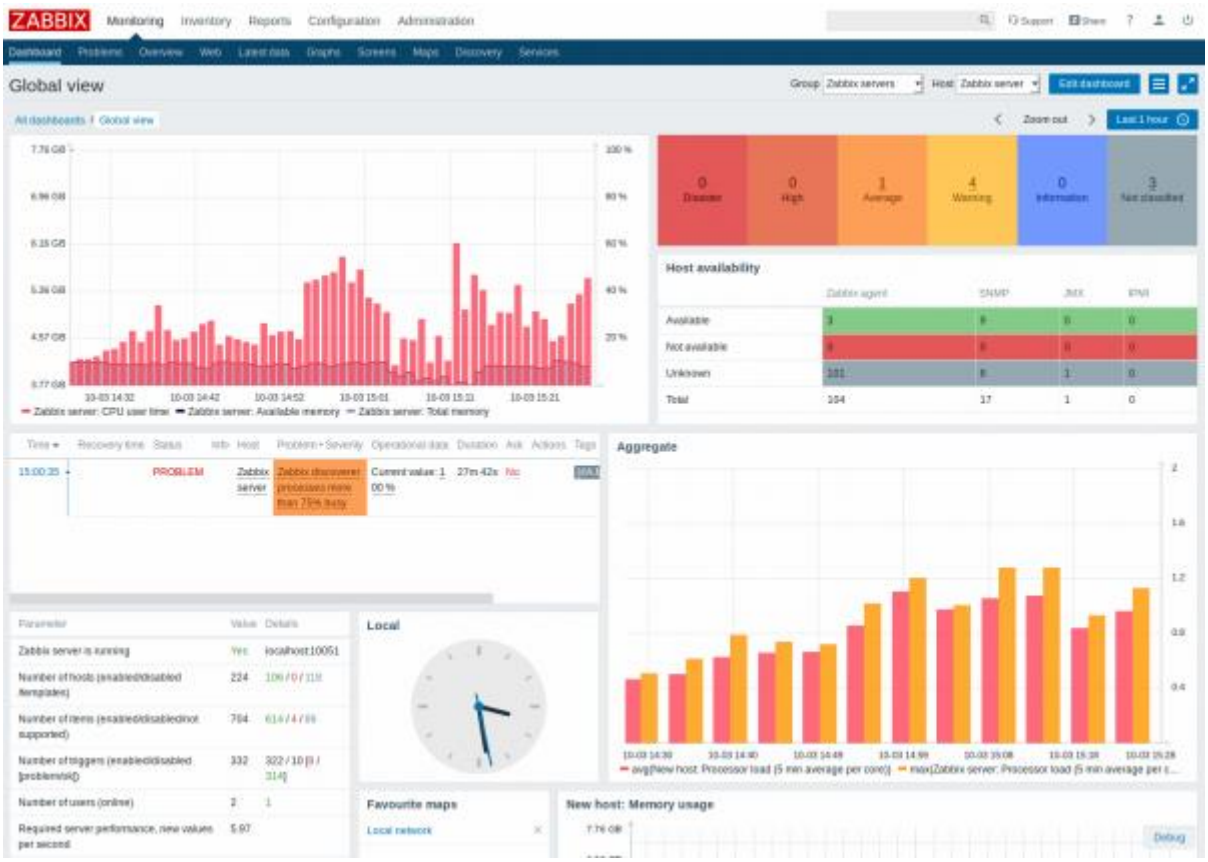
Slika 21 Prikaz sučelja Zenoss alata za nadzor, [21]

Neke od mogućnosti koje pruža Zenoss:

- Dinamične nadzorne ploče *(*engl. *dashboard)* - omogućavaju preglednost, mogućnost pregledavanja raznih vrsta grafova, mogućnost ručnog uređivanja i raspoređivanja elemenata na početnom sučelju
- Mogućnosti grafova - modeliranje grafova po želji korisnika sa crtanjem i izdvajanjem samo bitnih informacija na grafovima
- Automatsko otkrivanje mrežnih uređaja i promjena u mrežnoj konfiguraciji
- Automatsko slanje alarma i obavještavanje - omogućava postavljanje pravila i granica (mrežne iskoristivosti linka, iskorištenje procesora, memorije, granica temperature i dr.) na serverima i mrežnoj opreme. Ukoliko dođe do nekog incidenta ili prelaska postavljenih granica, administratorima i korisnicima se šalju upozorenja putem elektronske pošte, SMS-a i dr.
- Ugrađene procedure za izradu sigurnosnih kopija i vraćanje konfiguracije, [20], [21].

5.4. Zabbix

Zabbix je besplatno programsko rješenje za nadzor mreža, operacijskih sustava i aplikacija u realnom vremenu. Kreirao ga je Alexei Vladishev, a cijeli projekt započet je 1998. godine. Osim što je program otvorenog koda, iza sustava Zabbix stoji istoimena firma (Zabbix SIA) koja aktivno pruža podršku, razvoj i integraciju navedenog sustava. Postoji javno dostupna službena dokumentacija koja detaljno opisuje specifičnosti, mogućnosti i samu instalaciju postojeće verzije sustava. Instalacija i implementacija sustava Zabbix je vrlo jednostavna i brza za razliku od ostalih sustava. Na slici 22 prikazano je grafičko web sučelje sustava Zabbix.



Slika 22 Prikaz sučelja Zabbix alata za nadzor, [22]

Sustav Zabbix ima vrlo jednostavno i pregledno grafičko web sučelje koje omogućava korisnicima puno opcija kao: promjena konfiguracije i izgleda samog sučelja, jednostavnije dodavanje hostova u svega par koraka, pregledni grafovi prometa kroz vremensko razdoblje koje je moguće proizvoljno odabrati te izvesti i pohraniti, proizvoljno kreiranje upozorenja i obavijesti koje je moguće slati putem elektroničke pošte ili SMS-a, gotovi predlošci za FTP, HTTP, HTTPS, IMAP, LDAP, MySQL, SMTP, SSH, POP, Telnet i dr. Jedna od mogućnosti

sustava Zabbix koju pruža je integracija vlastitog *ticketing* sustava. Postoje tvrtke koje uz pojedine nadogradnje i modifikacije plasiraju sustav Zabbix kao vlastiti proizvod što je isto jedna od velikih prednosti, ali i mogućnosti. Sustav Zabbix je besplatan što omogućava da se uz minimalne troškove obavlja nadzor komunikacijske mreže, čime se smanjuje vrijeme nedostupnosti mrežnih uređaja, a povećava dostupnost i efikasnost usluga koju pruža alternativni telekomunikacijski operator. Zbog svih navedenih prednosti i mogućnosti koje pruža, sustav Zabbix zauzima mjesto među najboljim i najkompletnijim sustavima za nadzor mreže i uređaja.

Neke od mogućnosti koje pruže sustav Zabbix su:

- Pregledni grafovi - mogućnost kreiranja prilagođenih grafova kombinirajući više stavki u jednom prikazu; crtanje grafova u stvarnom vremenu; mogućnost pregleda grafova kroz vremensko razdoblje koje je moguće proizvoljno odabrati te izvesti i pohraniti
- Praktično korisničko web sučelje - mogućnost pristupa od bilo gdje; mogućnost kreiranja i uređivanja početnog zaslona kao nadzorne ploče *(*engl. *dashboard)* sa izdvajanjem samo bitnih resursa s većim prioritetom; mogućnost izrade mrežnih mapa
- Otkrivanje mreže i uređaja - automatsko otkrivanje mrežnih uređaja, mrežnih sučelja, SNMP OID-ova
- Kreiranje obavijesti i upozorenja - mogućnost slanja upozorenja ili eskalacija putem elektroničke pošte, SMS-a administratorima ili proizvoljnim korisnicima ukoliko dođe do određenih poteškoća ili uređaji postanu nedostupni
- Dozvole pristupa i pregleda - sigurna provjera autentičnosti korisnika, pojedini korisnici imaju mogućnost uređivanja i mijenjanja parametara, dok određeni imaju samo mogućnost pregleda. Također, određeni korisnici mogu biti ograničeni samo na određene prikaze
- Korištenje predložaka - dostupan velik broj gotovih predložaka
- Pohrana podataka – pohranjivanje podataka u bazu podataka, mogućnost kreiranje izvještaja
- Zabbix API - pruža programsko sučelje Zabbixu za masovne manipulacije, integraciju softvera treće strane i dr.
- Nadzora složenih okruženja - mogućnost udaljenog nadzora MPLS VPN mreža pomoću Zabbix *proxy-a,* [22].

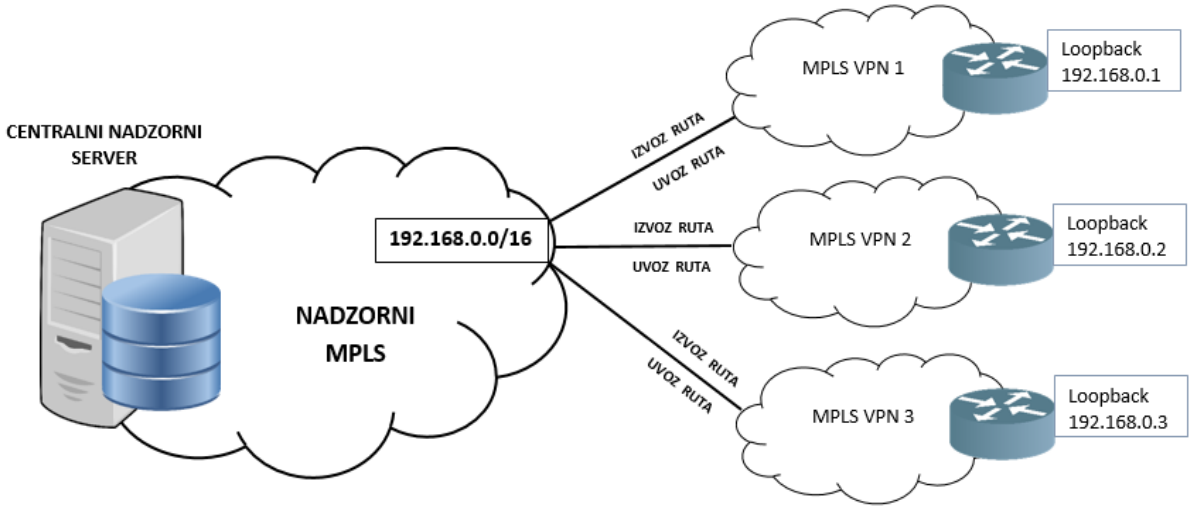
6. RJEŠENJA ZA NADZOR ZATVORENIH MPLS MREŽA

Aktualno pitanje s kojim se danas susreće većina telekomunikacijskih operatora je kako obavljati nadzor mreže i mrežnih uređaja koji se nalaze unutar zatvorene MPLS VPN mreže. U ovome poglavlju teorijski će biti navedeno i ukratko opisano par mogućih rješenja. Moguća rješenja za nadzor MPLS VPN mreže su:

- Rješenje pomoću centralnog nadzornog servera i nadzornog MPLS-a
- Rješenje unutar kojeg svaka MPLS VPN mreža ima vlastiti server za nadzor i
- Rješenje uz pomoć *proxy* servisa unutar kontejner okoline.

6.1. Rješenje pomoću centralnog nadzornog servera i nadzornog MPLS-a

Kao što je vidljivo na slici 23, u navedenom rješenju nalazi se jedan centralni nadzorni server koji se nalazi unutar nadzornog MPLS-a, a služi za nadzor korisničkih mreža i uređaja. Prilikom konfiguracije i implementacije ovoga rješenja potrebno je obratiti pozornost pri dodjeljivanju IP adresa.



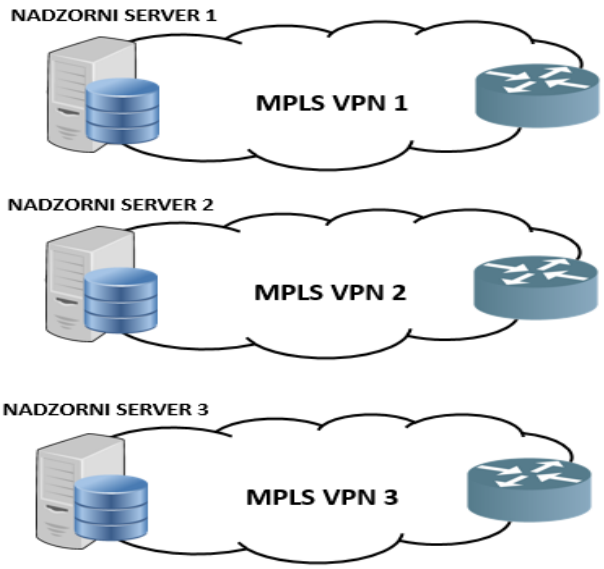
Slika 23 Prikaz rješenja pomoću centralnog nadzornog servera i nadzornog MPLS-a

Za navedeno rješenje potrebno je izdvojiti poseban raspon *(*engl. *range)* adresa koji će se postavljati u obliku *loopback* adresa na korisničke usmjerivače. Svaki od tih *loopback-a* potrebno je izvesti *(*engl. *export route)* iz nadzornog MPLS-a u korisničku MPLS VPN mrežu

i uvesti *(*engl. *import route)* u suprotnom smjeru od korisnika prema nadzornom MPLS-u. Unutar korisničkog MPLS također je potrebno odraditi konfiguraciju za usmjeravanje adresa kako bi se omogućila komunikacija između centralnog nadzornog servera i usmjerivača u korisničkoj MPLS mreži. Nedostatak navedenog rješenja je što se odabrani raspon adresa ne smije koristiti unutar više korisničkih MPLS mreža, nego samo unutar jedne pošto može doći do preklapanja adresa. Ukoliko dođe do preklapanja adresa, potrebno je zatražiti korisnika da odradi promjenu adresne sheme unutar privatne mreže, što baš nije praktično. Potrebno je vođenje detaljne evidencije adresa, tj. adresnog managementa kako bi se znalo koji raspon adresa postavljen kod kojeg korisnika i unutar koje MPLS VPN mreže. Također, navedeno rješenje nije jednostavno za održavanje pošto se prilikom dodavanja svaka novog uređaja u nadzor, mora dodijeliti nova *loopback* adresa na uređaju, odraditi uvoz i izvoz *loopback* adrese te konfigurirati usmjeravanje unutar korisničkog MPLS-a. Unutar navedenog rješenja nije moguće koristiti vatrozid za zabranu prometa iz MPLS VPN mreže prema centralnom serveru za nadzor i obrnuto. Bez vatrozida nije moguće napraviti niti ograničenje prometa po protokolu (SNMP, ICMP, telnet, SSH i dr.). Prilikom konfiguracije i implementacije ovoga rješenja treba biti na oprezu jer uslijed pogreške može doći do narušavanja sigurnosni aspekata, [23].

6.2. Rješenje unutar kojeg svaka MPLS VPN mreža ima vlastiti server za nadzor

Kao što je prikazano na slici 24, u navedenom rješenju svaka MPLS VPN mreža ima vlastiti (zaseban) server za nadzor svoje mreže. Kod navedenog rješenja se implementira i konfigurira više nadzornih servera koji se mogu nalaziti u izoliranim virtualnim platformama (virtualni strojevi) ili zasebno na više fizičkih servera. Samim time postiže se veća sigurnost. Kroz management adresu telekomunikacijskog operatora konfigurira se zaseban VLAN za svaki nadzorni sustav koji završava u korisničkom MPLS-u preko vatrozida kako ne bi došlo do miješanja prometa na metro razini.

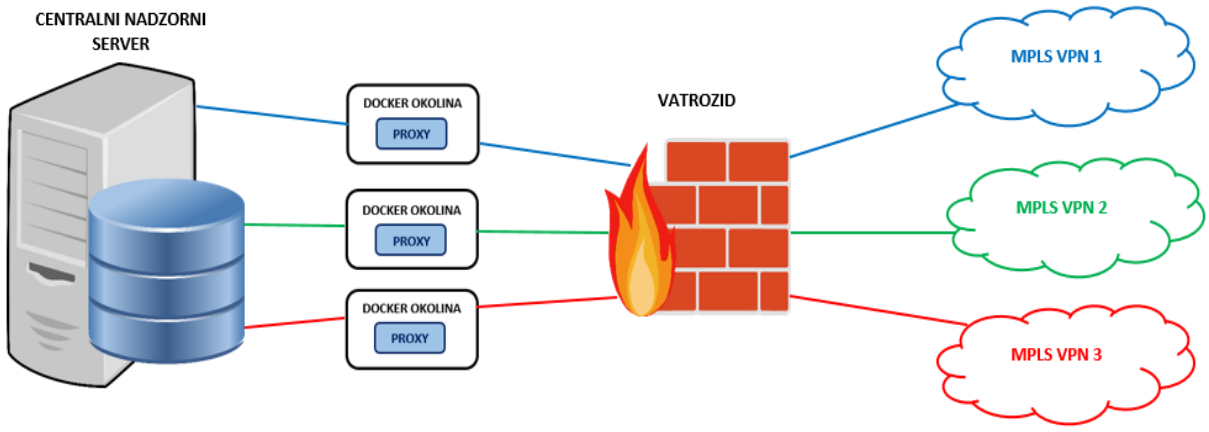


Slika 24 Prikaz rješenja unutar kojeg svaka MPLS VPN mreža ima vlastiti server za nadzor

Prednost navedenog rješenja je mogućnost implementacije i korištenja vatrozida kako bi ograničili pristup i promet od nadzorne mreže do korisničkih mreža. Pomoću vatrozida postoji mogućnost ograničavanja i propuštanja prometa po određenim portovima ili protokolima. Kod naprednijih vatrozida postoji mogućnost uvida u vrstu prometa i ponašanje prometa. Također, bitna prednost za razliku od prvog rješenja je da se ne mora koristiti točno određeni raspon adresa prilikom konfiguriranja *loopback* adrese na usmjerivačima kod korisnika, već se nudi fleksibilnost da se u dogovoru s korisnikom uskladi adresni prostor, odnosno međumreža. Nedostatak ovakvog rješenja je što se za svaki od nadzornih sustava mora implementirati i konfigurirati zaseban server. Takvo rješenje zahtjeva više resursa i obično je financijski puno skuplje. Osim što je financijski skuplje, samim time se povećava i kompleksnost održavanja jer se svaki server mora posebno konfigurirati, održavati, obnavljati pojedine komponente i servise. Najveći problem je što se nadzor ne odvija preko jedne konzole, tj. sučelja. Za nadzor svake pojedine MPLS VPN mreže mora se pristupiti sučelju servera za nadzor koji služi za nadzor te određene MPLS VPN mreže. Ovakvo rješenje je vrlo nepraktično, ali ako alat za nadzor nema mogućnost *proxy* servera onda je nadzor moguće implementirati na ovaj način, [6].

6.3. Rješenje uz pomoć proxy servisa unutar kontejner okoline

U navedenom rješenju nalazi se centralni nadzorni server koji ostvaraje komunikaciju s MPLS VPN mrežom pomoću *proxy* servisa koji se nalazi unutar kontejner okoline. Navedeno rješenje za nadzor sastoji se od alat za nadzor, baze podataka i *proxy* servisa unutar kontejner okoline. Ovisno o sigurnosnom kontekstu moguće je da se sve komponente pokreću na istom serveru ili svaka komponenta na različitom.



Slika 25 Prikaz rješenja uz pomoć proxy servisa unutar kontejner okoline

Konkretno, u ovome diplomskom radu upravo će bit korišteno i opisano navedeno rješenje nadzora MPLS VPN mreže uz pomoć *proxy* servisa unutar kontejner okoline. Alat za nadzor koji će biti korišten je Zabbix, a za „ulaz“ u MPLS VPN mrežu koristi će se *zabbix proxy* servis koji se pokreće unutar alata *Docker Engine* (koristeći „kontejnere“). *Docker Engine* je klijent-server aplikacija dizajnirana za kreiranje, implementaciju i pokretanje aplikacija koristeći kontejnere. Kontejnere možemo smatrati minimalnim verzijama virtualnih strojeva tj. oni sadrže samo ono što je apsolutno potrebno za pokretanje željene aplikacije. Kontejner omogućuje izoliranu okolinu za pokretanje zasebnih *proxy* servisa za svaku MPLS VPN mrežu posebno. Što znači da svaki korisnik, tj. firma unutar zasebne MPLS VPN mreže može koristiti bilo koju adresu neovisno da li se adresa preklapa s adresom iz mreže drugog korisnika. Za svaku MPLS VPN mrežu koristi će se zaseban *proxy* servis unutar izolirane okoline kontejnera. Kako bi se osigurala sigurnost u komunikaciji između MPLS VPN mreža i nadzornog alata koristi se vatrozid. Na razini vatrozida moguće je ograničiti promet po portu, protokolu i aplikaciji. U konkretnom primjeru Zabbix alatu za nadzor dozvoljeno je korištenje

prometa po portu 161, protokol SNMP i aplikacija *zabbix proxy*. Kako bi se povećala sigurnost unutar navedenog rješenja, moguće je svaku od navedenih komponenti implementirati i pokretati na zasebnom serveru i u zasebnoj sigurnosnoj zoni koju omogućava vatrozid. Ukoliko se svaka komponenta pokreće na zasebnom serveru, automatski je složenije održavanje i samo rješenje zahtjeva više resursa. Jedna od glavnih prednosti navedenog rješenja je ta što *zabbix proxy* servisi mogu raditi neovisno da li je server dostupan. Nakon što dobiju instrukcije od servera o otkrivanju *(*engl. *discovery)* uređaja u mreži i prikupljanju podataka, *zabbix proxy* servisi pohranjuju prikupljene podatke u privremeni spremnik *(*engl. *buffer)* i spremaju u vlastitu bazu. Kad server ponovo postane dostupan prosljeđuju mu sve prikupljene podatke. *Proxy* servisi mogu raditi određeno vrijeme neovisno o serveru, tj. sve dok se ne zapuni spremnik. Praktična primjena navedene prednosti rada *proxy* servisa neovisno o serveru očituje se prilikom nadogradnje ili ponovnog pokretanje servera. Implementacija *proxy* servisa unutar kontejner okoline daje određeni stupanj kompleksnosti navedenom rješenju, ali uzevši u obzir sve navede prednosti poput sigurnosti i dostupnosti navedeno rješenje je prihvatljivo, [6].

7. STUDIJA SLUČAJA: NADZOR UREĐAJA UNUTAR ZATVORENIH MPLS MREŽA UZ POMOĆ PROGRAMSKOG ALATA *DOCKER ENGINE*

Kako bi se omogućio nadzor uređaja unutar MPLS VPN mreža potrebno je odraditi pravilnu konfiguraciju svih komponenti koje sudjeluju u nadzoru. Potrebno je na PE usmjerivačima omogućiti usmjeravanje unutar MPLS VPN mreže, na zasebnom serveru kreirati kontejner *(*engl. *Docker)* okoline te kreirati *proxy* servis unutar alata Zabbix kao i namjestiti akcije i radnje za otkrivanje uređaja unutar MPLS VPN mreže. Navedene radnje bit će prikazane u navedenom poglavlju. Važna napomena, sva konfiguracija i sve radnje rađene su na opremi i serverima telekomunikacijskog operatora. Zbog zaštite i tajnosti podataka sve korištene IP adrese, AS unutar BGP-a i ostali podaci nemaju veze sa stvarnim podacima. Nadzor će se obavljati na MPLS L3 VPN mreži firme Faletar koja ima tri poslovnice.

7.1. Konfiguracija na PE usmjerivačima za omogućavanje usmjeravanja unutar MPLS VPN mreže

U nastavku će biti istaknuta samo najvažnija konfiguracija na PE usmjerivačima za uspostavu usmjeravanja prefiksa unutar MPLS VPN mreže. Bit će prikazano koje postavke trebaju biti omogućene na PE usmjerivaču, kako kreirati virtualnu tablicu usmjeravanja VRF, kako pokrenuti BGP protokol za usmjeravanje, te kako se uspostavlja veza sa susjednim PE usmjerivačima. Ostatak konfiguracije i usmjeravanja unutar mreže telekomunikacijskog operatora između PE i P usmjerivača neće biti prikazan u ovom diplomskom radu.

Na početku potrebno je na PE usmjerivačima omogućiti, tj. aktivirati korištenje MPLS protokola. Za aktivaciju protokola MPLS za usmjeravanje oznaka IPv4 paketa potrebno je koristiti sljedeću naredbu u globalnom načinu konfiguracije:

```
mpls ip
```

Za specifikaciju protokola distribucije oznaka LDP na platformi, tj. PE usmjerivačima potrebno je koristiti sljedeću naredbu:

```
mpls label protocol ldp
```

Na PE usmjerivaču potrebno je unaprijed konfigurirati virtualnu tablicu usmjeravanja i prosljeđivanja pomoću naredbe *ip vrf VFR_IME.* U navedenom primjeru bit će kreirana VRF tablica imena *faletar*. Navedenom naredbom ulazimo u način za konfiguriranje dodatnih postavki unutar VRF-a. Prilikom razmjena informacija između PE usmjerivača bitno je da se IPv4 prefiks svakog korisnika jedinstveno odredi pomoću RD *(*engl. *Route Distinguisher)* kako ne bi došlo do preklapanje adresa i problema prilikom usmjeravanja. RD omogućuje korisnicima korištenje bilo koje privatne IP adrese na svojim CE usmjerivačima bez obzira na IP adrese drugih korisnika, a definira se unutar VRF-a pomoću naredbe *rd AS_broj:NN*, gdje *AS_broj* označava broj autonomnog sustava ISP-a, a *NN* proizvoljan broj (često ID korisnika). Važno je napomenuti da RD ne označava kojem VRF-u pripada adresa/prefiks. Kako bi PE usmjerivači znali kojem točno VRF-u pripada koja adresa/prefiks koristi se oznaka RT (engl. *Route Target*) koja se definira također unutar VRF-a. Konfigurira se pomoću naredbe *route-target [export | import] AS_broj:NN* za izvoz ili uvoz prefiksa. U nastavku je primjer konfiguracije na PE usmjerivaču:

```
ip vrf faletar
  rd 64600:1234567
  route-target export 64600:1234567
  route-target import 64600:1234567
```

Razmjena informacija o VRF tablicama između PE usmjerivača obavlja se pomoću *„Multiprotocol BGP-a“* (MP-BGP). Za pokretanje i konfiguraciju BGP protokola koristi se naredba *router bgp AS_broj*. U navedenom primjeru korišten će biti AS broj 64600. Pomoću naredbe *bgp router-id IP_adresa* konfigurira se ID usmjerivača koji je u navedenom slučaju ujedno predstavlja i adresu PE usmjerivača. Pomoću protokola BGP uspostavlja se veza/sesija između PE usmjerivača, tzv. *„BGP neighbour“*. Za uspostavu BGP sesije koriste se naredba *neighbor IPadresa_SusjednogPE remote-as AS_broj*. U nastavku je primjer konfiguracije na PE usmjerivaču:

```
router bgp 64600
  bgp router-id 11.11.11.11
  neighbor 12.12.12.12 remote-as 64600
  neighbor 12.12.12.12 description PE-2
  neighbor 13.13.13.13 remote-as 64600
  neighbor 13.13.13.13 description PE-3
```

Unutar BGP protokola za usmjeravanje postoji koncept *address families,* što bi u doslovnom prijevodu značilo obitelj adresa. Koncept podržava četiri vrste adresa, a to su: IPv4, IPv6, VPNv4 (VPN-IPv4), i VPNv6 (VPN-IPv6). Prilikom razmjena informacija između PE usmjerivača bitno je da se IPv4 prefiks svakog korisnika jedinstveno odredi pomoću RD kako ne bi došlo do preklapanje adresa i problema prilikom usmjeravanja. RD u kombinaciji sa IPv4 prefiksom čini VPNv4 adresu koja se razmjenjuje između PE usmjerivača pomoću BGP protokola. Za omogućavanje razmjene i usmjeravanja VPNv4 adresa između PE usmjerivača potrebno je koristiti naredbu *address-family vpnv4.* Pomoću naredbe *neighbor IPadresa_SusjednogPE activate* aktivira se oglašavanje VPNv4 prefiksa prema IP adresama susjednih PE usmjerivača. Već spomenuti RD i RT predstavljaju atribute BGP proširene zajednice *(*engl. *BGP extended community),* a za njihovu „aktivaciju" unutar BGP protokola potrebno je koristiti naredbu *neighbor IPadresa_SusjednogPE send-community extended.* U nastavku je primjer konfiguracije na PE usmjerivaču:

```
address-family vpnv4
  neighbor 12.12.12.12 activate
  neighbor 12.12.12.12 send-community extended
  neighbor 13.13.13.13 activate
  neighbor 13.13.13.13 send-community extended
```

Za razmjenu IPv4 prefiksa između PE i CE usmjerivača unutar BGP protokola potrebno je definirati IPv4 *address-families* koje pripadaju određenoj VRF-a tablici. U navedenom slučaju oglašavat će se IPv4 prefiksi iz VRF tablice imena *„faletar"*. Naredba *redistribute connected* služi za aktiviranje oglašavanja ruta koje se automatski uspostavljaju kada je na sučelju omogućen IP, dok naredba *redistribute static* služi za oglašavanje statičkih ruta. U nastavku se nalazi primjer konfiguracije na PE usmjerivaču koja je opisana:

```
address-family ipv4 vrf faletar
  redistribute connected
  redistribute static
  exit-address-family
```

Za svaku od tri poslovnice potrebno je na PE usmjerivaču konfigurirati pojedinačno sučelje *(*engl. *Interface)* na koje se povezuje CE usmjerivač. Pomoću naredbe *interface GigabitEthernet0/0* ulazi se u konfiguraciju sučelja *GigabitEthernet0/0* te definira opis pomoću naredbe *descrption* iza koje se unosi tekst opisa. Kako bi povezali VRF s određenim sučeljem, unutar konfiguracije sučelja koristi se naredba *ip vrf forwarding VRF_IME*. U navedenom slučaju sučelje *GigabitEthernet0/0* povezujemo sa VRF-om *faletar*. Također, unutar konfiguracije sučelja dodjeljuje se IP adresa i mrežna maska. Isti je postupak i na preostala dva sučelja (*GigabitEthernet0/1 i GigabitEthernet0/2*). U nastavku se nalazi primjer konfiguracije na PE usmjerivaču koji je opisan u tekstu:

```
interface GigabitEthernet0/0
 description Faletar P1 Zagreb
 ip vrf forwarding faletar
 ip address 10.10.10.1 255.255.255.252

interface GigabitEthernet0/1
 description Faletar P2 Split
 ip vrf forwarding faletar
 ip address 10.10.10.5 255.255.255.252

interface GigabitEthernet0/2
 description Faletar P3 Bjelovar
 ip vrf forwarding faletar
 ip address 10.10.10.9 255.255.255.252
```

Na PE usmjerivaču potrebno je napraviti poveznicu između MPLS L3 VPN mreže i mreže kontejnera u kojoj se nalazi *proxy* servis za pristup MPLS L3 VPN mreži. Potrebno je konfigurirati međumrežu kojem će se omogućiti „ulaz“ *proxy* servisu u MPLS L3 VPN mrežu. Konfigurirat će se virtualno pod-sučelje *(*engl. *sub-interface) Bundle-Ether24.1800*. Bit će korišten trunk link između PE usmjerivača i centralnog servera za nadzor za prijenos i filtriranja prometa po specifičnom vlan-u, u ovome slučaju vlan 1800. Primjer korištene konfiguracije je u nastavku:

```
interface Bundle-Ether24.1800
 description Korisnik: Faletar-zabbix
 vrf faletar
 ipv4 address 10.100.10.5 255.255.255.252
 encapsulation dot1q 1800
```

7.2. Kreiranje baze, korisnika i dodjeljivanje prava

Kako bi svaki *proxy* servis mogao prikupljati podatke potrebna je baza podataka za spremanje prikupljenih podataka. Na istom serveru na kojem se nalazi i Zabbix server za nadzor potrebno je kreirati bazu podataka. U navedenom primjeru biti će kreirana baza podataka s nazivom *faletar* i postavljeno UTF-8 kodiranje za znakovne nizove.

```
create database faletar character set utf8 collate utf8_bin;
```

Također potrebno je kreirati korisnika *(*engl. *user)* kojim će se *proxy* servis spajati na kreiranu bazu. Za navedeni primjer kreirat će se korisnik *faletar.*

```
create user faletar;
```

Kako bi kreirani korisnik *faletar* imao mogućnost upravljanja nad bazom, potrebno mu je definirati koja će prava korisnik imati nad bazom podataka *faletar.* Nakon što se definiraju prava korisnika nad bazom, potrebno je i dodijeliti prava s naredbom *flush privileges.*

```
grant all privileges on faletar.* to faletar identified by 'faletar';

flush privileges;
```

7.3. Kreiranje i pokretanje kontejnera

Na istom serveru gdje se nalaze Zabbix server i baza podataka, pokreće se i *Docker Engine* klijent-server aplikacija dizajnirana za kreiranje, implementaciju i pokretanje servisa i aplikacija koristeći kontejnere. Kako bi se obavljao nadzor unutar MPLS VPN mreže, nadzorni alat Zabbix treba imati direktan pristup MPLS L3 VPN mreži. U takvim situacijama se kreira i pokreće docker mreža koja koristi mrežni *driver „macvlan“* za dodjeljivanje MAC adrese svakom virtualnom mrežnom sučelju kontejnera što ih čini fizičkim mrežnim sučeljem koje je izravno povezano s MPLS VPN mrežom. U tom slučaju potrebno je odrediti fizičko sučelje unutar docker mreža koje će koristiti *driver macvlan*, kao i adresu mreže i izlaz *(eng, gateway)*. Za stvaranje i pokretanje docker mreže koja koristi *driver macvlan* za povezivanje s određenim fizičkim mrežnim sučeljem, koristi se naredba *docker network create -d macvlan*. Također na kraju navedene naredbe potrebno je navesti unutar parametra *parent* sučelje preko kojeg će promet fizički prolaziti kroz docker mrežu. U navedenom slučaju unutar parametra *parent* navedeno je podsučelje (engl. *subinterface*) eth2.1800. Sučelje eth2.1800 označava da se koristi trunk link za prijenos i filtriranja prometa po specifičnom vlan-u, u ovome slučaju vlan 1800.

```
sudo docker network create -d macvlan  \
  --subnet 10.100.10.4/30  \
  --gateway 10.100.10.5  \
  -o parent=eth2.1800 faletar
```

Za na nadzor MPLS L3 VPN mreže koristi će se zaseban *proxy* servis unutar izolirane okoline kontejnera. Za pokretanje kontejnera unutar kojeg će se pokretati *proxy* servis koristi se naredba *docker run* iza koje slijede razne dodatne opcije. U navedenom primjeru prilikom pokretanja kontejnera definirat će se sljedeće opcije:

--name (ime kontejnera),

--ip (adresa kontejnera),

--network (povezivanje kontejnera navedenoj mreži),

--restart (ponovo primjenjuje definirane parametre i pravila prilikom novog pokretanja).

Također prilikom pokretanja će se definirati određene varijable okruženja:

-e DB_SERVER_HOST (predstavlja IP adresu MySQL servera),

-e MYSQL_USER (korisnik koji se povezuje na Zabbix bazu na MySQL serveru),

-e MYSQL_PASSWORD (lozinka za povezivanje na MySQL server),

-e MYSQL_DATABASE (naziv baze koje se nalazi na MySQL serveru),

-e ZBX_HOSTNAME (naziv *proxy* servisa unutar Zabbix alata),

-e ZBX_SERVER_HOST (predstavlja IP adresu Zabbix servera),

-d zabbix/zabbix-proxy-mysql:alpine-4.2-latest (predstavlja verziju Zabbix *proxy Docker images*). U nastavku je primjer korištenja opisane konfiguracije:

```
sudo docker run --name= faletar  \
        --ip=10.100.10.6  \
        --network=faletar  \
        --hostname= faletar  \
        --restart=always   \
        -e DB_SERVER_HOST=10.255.254.1   \
        -e MYSQL_USER= faletar   \
        -e MYSQL_PASSWORD= faletar123  \
        -e MYSQL_DATABASE= faletar   \
        -e ZBX_HOSTNAME= faletar   \
        -e ZBX_SERVER_HOST=10.255.254.1  \
        -d zabbix/zabbix-proxy-mysql:alpine-4.2-latest
```

Kako bi kreirani i pokrenuti kontejneri mogli komunicirati i koristiti zabbix servise i bazu podataka potrebno je povezati *bridge* mrežu sa kontejnerom. U navedenom slučaju koristiti se zabbix *bridge* mreža preko koje se svi kontejneri povezuju s bazom podataka i zabbix servisima. Pomoću naredbe *docker network connect* povezujemo kontejner „*faletar“* s *bridge* mrežom *„zabbix“*.

```
        docker network connect zabbix faletar
```

7.4. Konfiguracija na CE usmjerivačima za omogućavanje nadzora

Kako bi nadzorni Zabbix server imao mogućnost nadzora CE usmjerivača i prikupljanja potrebnih mrežnih parametara sa usmjerivača potrebno je omogućiti korištenje odgovarajućeg protokola na navedenim uređajima. Prikupljanje podataka i nadzor usmjerivača i mrežnih uređaja bit će omogućen korištenjem SNMP protokola. Da bi se omogućio nadzor i prikupljanje podataka pomoću SNMP protokola koristi se naredba *snmp-server* iza koje se navodi takozvani zajedničkih znakovni niz *(*engl. *community string),* a to je zapravo niz ASCII znakova. Znakovni niz moguće je zamisliti kao korisnički id ili lozinka pomoću koje se omogućava pristup podacima usmjerivača ili drugih mrežnih uređaja. U nastavku naredbe se stavlja RO - *„read-only“* dozvola, što znači da se pomoću znakovnog niza omogućava samo čitanje, tj. prikupljanje podataka sa usmjerivača. Također na kraju naredbe se specificira pristupna lista *(*engl. *access-list)* koja odobrava pristup samo s određene adrese. U navedenom slučaju se koristi standardna pristupna lista oznake 10 koja odobrava pristup samo s adrese 10.100.10.6, a to je zapravo adresa kontejnera unutar koje se nalazi *zabbix proxy* servis koji pristupa i prikuplja podatke sa CE usmjerivača unutar MPLS L3 VPN mreže.

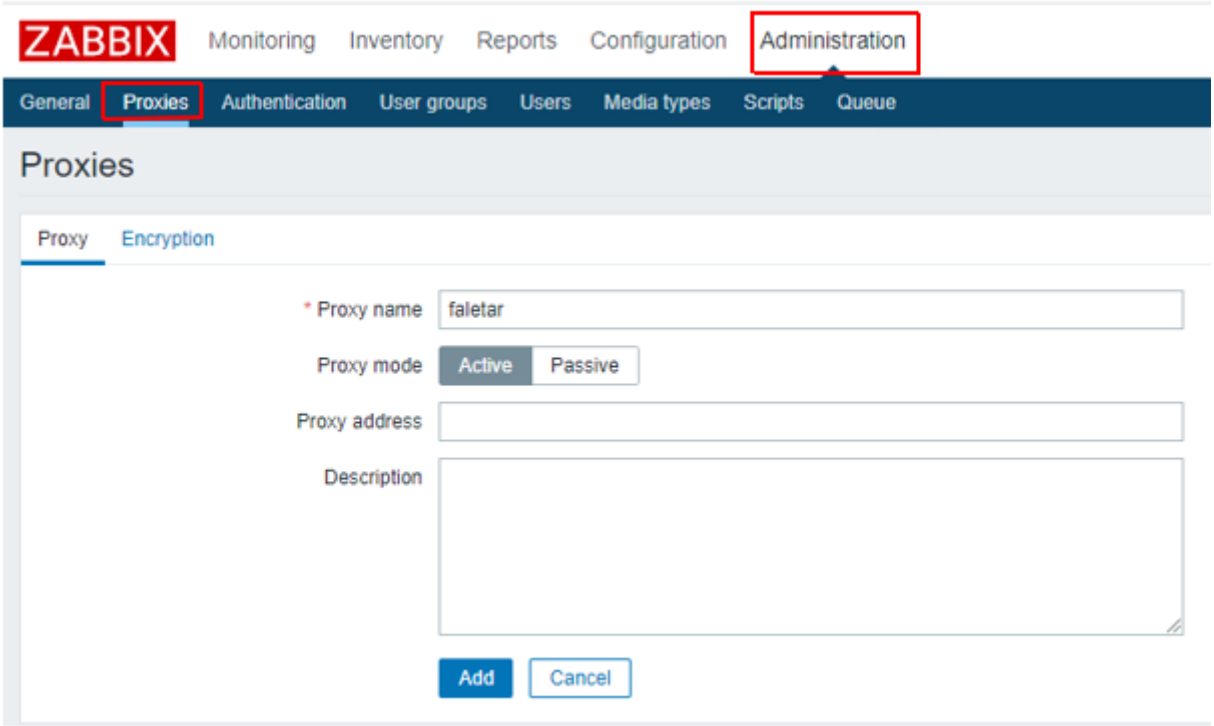
```
snmp-server community pr4istup RO 10

access-list 10 permit 10.100.10.6
```

7.5. Implementacija nadzora unutar Zabbix alata za nadzor

Nakon odrađene konfiguracije potrebnih parametara na PE i CE usmjerivačima, te implementacije i pokretanja baze podataka, docker mreže, kontejnera i *proxy* servisa na serveru, potrebno je unutar Zabbix alata za nadzor postaviti određene postavke za obavljanje nadzora. Kako bi se omogućio nadzor usmjerivača unutar MPLS L3 VPN mreže firme „Faletar“ potrebno je u Zabbix alatu za nadzor odraditi sljedeće: kreirati *proxy,* kreirati *host* grupu gdje će se svi otkriveni usmjerivači grupirati, kreirati predložak *(*engl. *template)* koji će biti korišten za otkrivanje usmjerivača u mreži i skupljanje podataka za crtanje grafova prometa, konfigurirati pravila za otkrivanje *(*engl. *discovery rule)* uređaja, kreirati i konfigurirati akcije *(*engl. *action)* koje će se poduzeti nakon što se otkrije uređaj u navedenoj mreži (npr. uređaj će

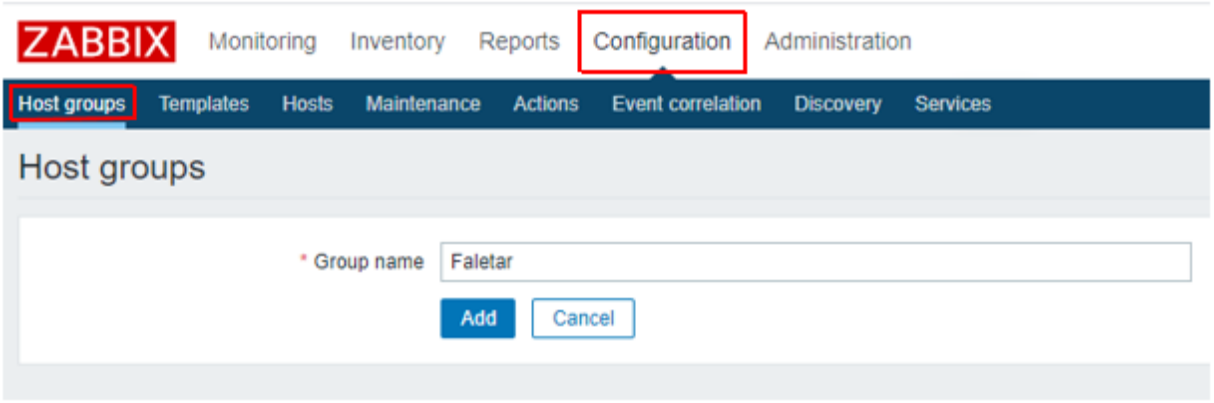
biti grupiran u *host* grupu „faletar“, povezan će biti sa predloškom za crtanje grafova prometa i provjeru dostupnosti i dr.).

Na početku je potrebno u Zabbix alatu za nadzor kreirati *proxy* servis koji služi kao „ulaz“ u određenu mrežu. U Zabbix alatu za nadzor nalaze se dvije trake s ponuđenim opcija. Za kreiranje *proxy* servisa potrebno je u prvoj traci odabrati opciju *„Administration“* nakon čega će u drugoj traci biti ponuđeno nekoliko opcija, a odabrat će se opcija *„Proxies“*. Potrebno je unijeti ime *proxy* servisa i kliknuti na gumb *„Add“* za dodavanje, tj. kreiranje. Na slici 26 nalazi se prikaz postupka kreiranja *proxy* servisa s nazivom „faletar“.



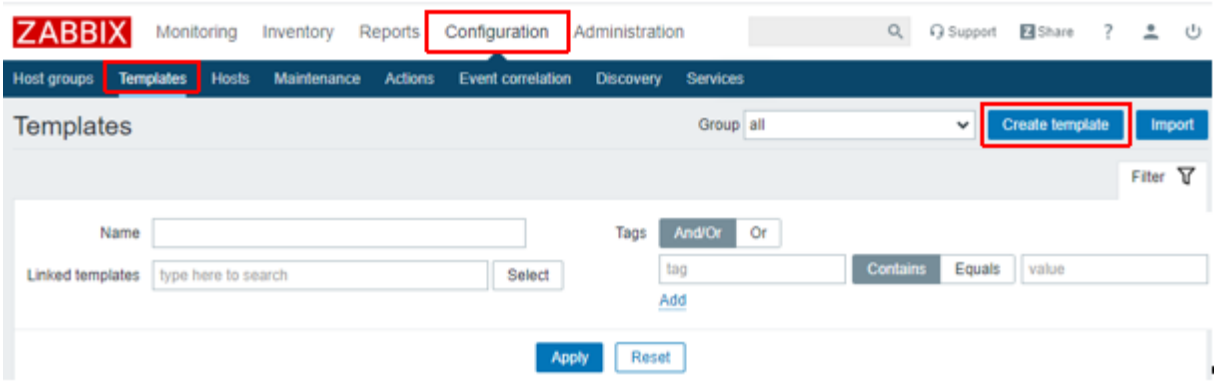
Slika 26 Prikaz postupka kreiranja proxy servisa

Nakon što je uspješno dodan *proxy* servis, potrebno je kreirati *„host“* grupu unutar koje će se grupirati svi otkriveni usmjerivači iz MPLS VPN mreže koja se nadzire. Za kreiranje *„host“* grupe potrebno je u prvoj traci odabrati opciju *„Configuration“* nakon čega će u drugoj traci biti ponuđeno nekoliko opcija, a odabrat će se opcija *„Host groups“*. Potrebno je unijeti ime *host* grupe i kliknuti na gumb *„Add“* za dodavanje, tj. kreiranje. Na slici ispod nalazi se prikaz postupka kreiranja *host* grupe s nazivom „Faletar“.



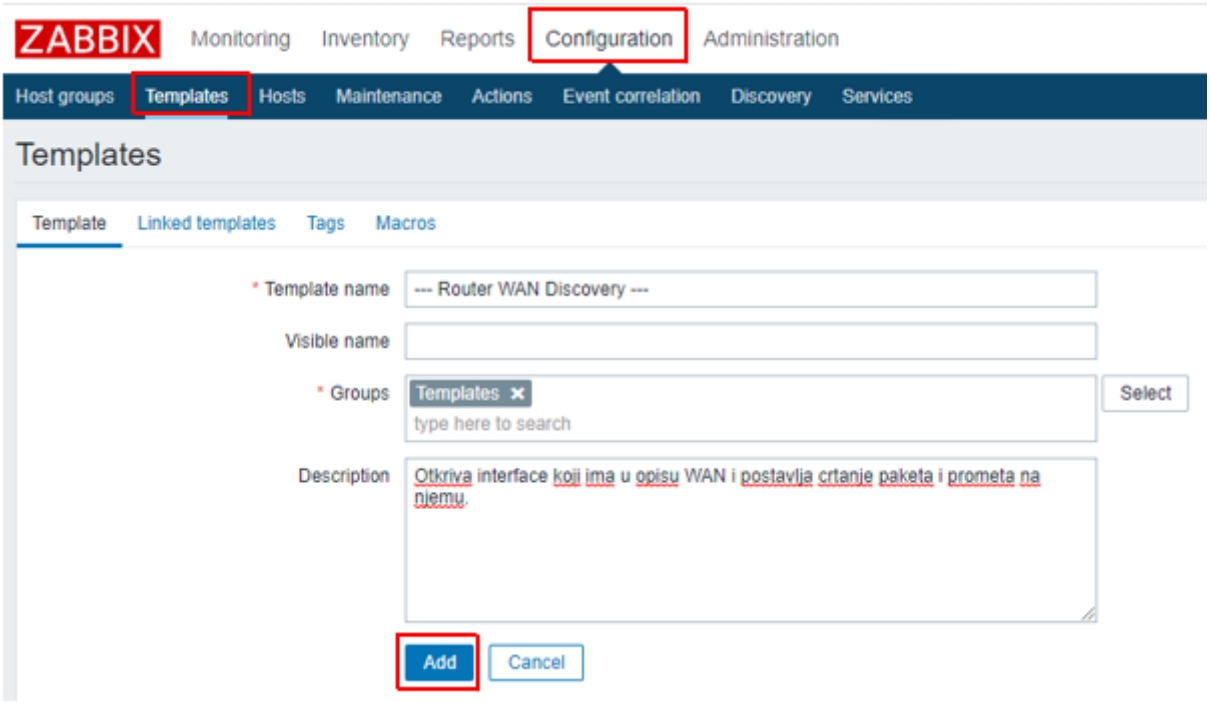
Slika 27 Prikaz postupka kreiranja host grupe

Kako bi se moglo prikupljati promet s sučelja usmjerivača potrebno je napraviti predložak koji će definirati navede radnje. Za kreiranje predloška potrebno je u prvoj traci odabrati opciju *„Configuration“* nakon čega će u drugoj traci biti ponuđeno nekoliko opcija, a odabrat će se opcija *„Templates“* te zatim u gornjem desnom uglu gumb *„Create template“* koji služi za ulaz u formu za kreiranje predloška. Na slici 28 je prikazan pristup formi za izradu predloška.



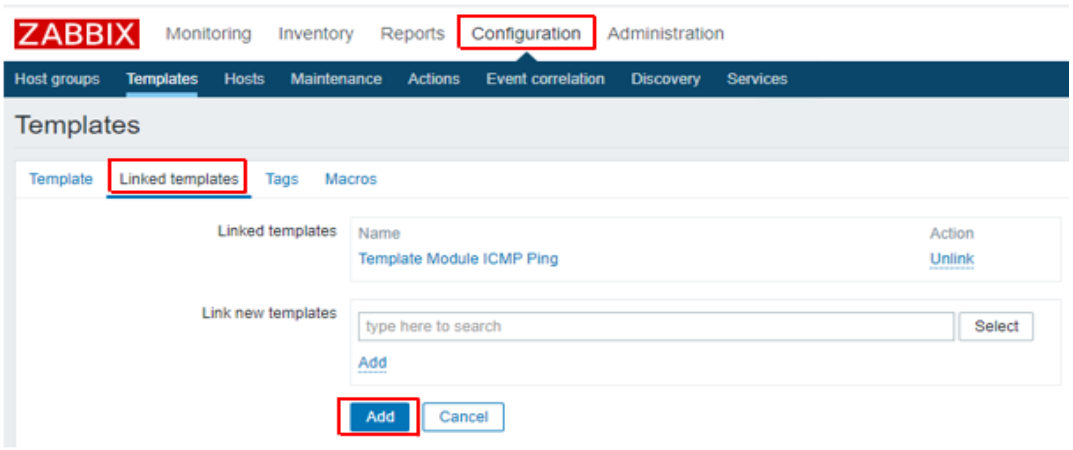
Slika 28 Pristup formi za izradu predloška

Nakon pristupa u formu za izradu predloška potrebno je prilagoditi i konfigurirati postavke predloška. U navedenom primjeru biti će kreiran predložak za otkrivanje sučelja (engl. *interface)* na usmjerivaču koji sadrži u opisu „WAN“ i postavlja crtanje prometa i paketa sa navedenog sučelja. Na početku će se definirati ima predloška kao *„Router WAN Discovery“*, postavit će se opis kako bi se znalo čemu služi predložak i smjestiti predložak unutar grupe *„Templates“* te nakon toga kliknuti na gumb *„Add“*. Na slici 29 je prikazan opisan postupak.



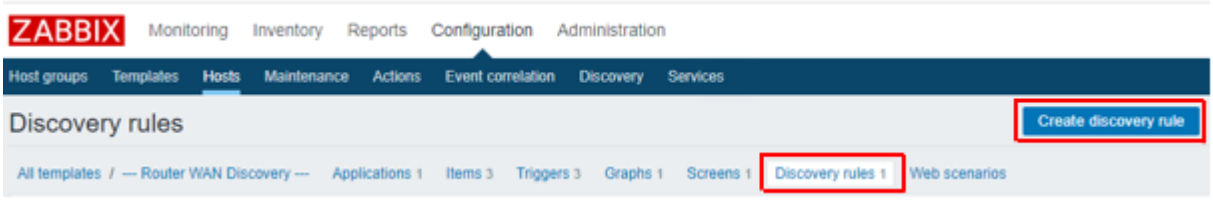
Slika 29 Prikaz postavljanja naziva i opisa predloška

Nakon što su postavljeni naziv i opis predloška, unutar forme za postavljanje predloška potrebno je odabrati opciju *„Linked templates“* gdje će se povezati postojeći, ugrađeni predložak unutar Zabbix alata za nadzor sa novim predloškom koji se upravo implementira. Povezani predložak naziva *„Template Module ICMP Ping“* služi za provjeru dostupnosti uređaja pomoću naredbe ICMP protokola – *ping*. Ukoliko promatrani uređaj ne odgovara na naredbu *ping* neko određeno vrijeme, bit će proglašen kao nedostupan. Na slici 30 je prikazan opisan postupak.



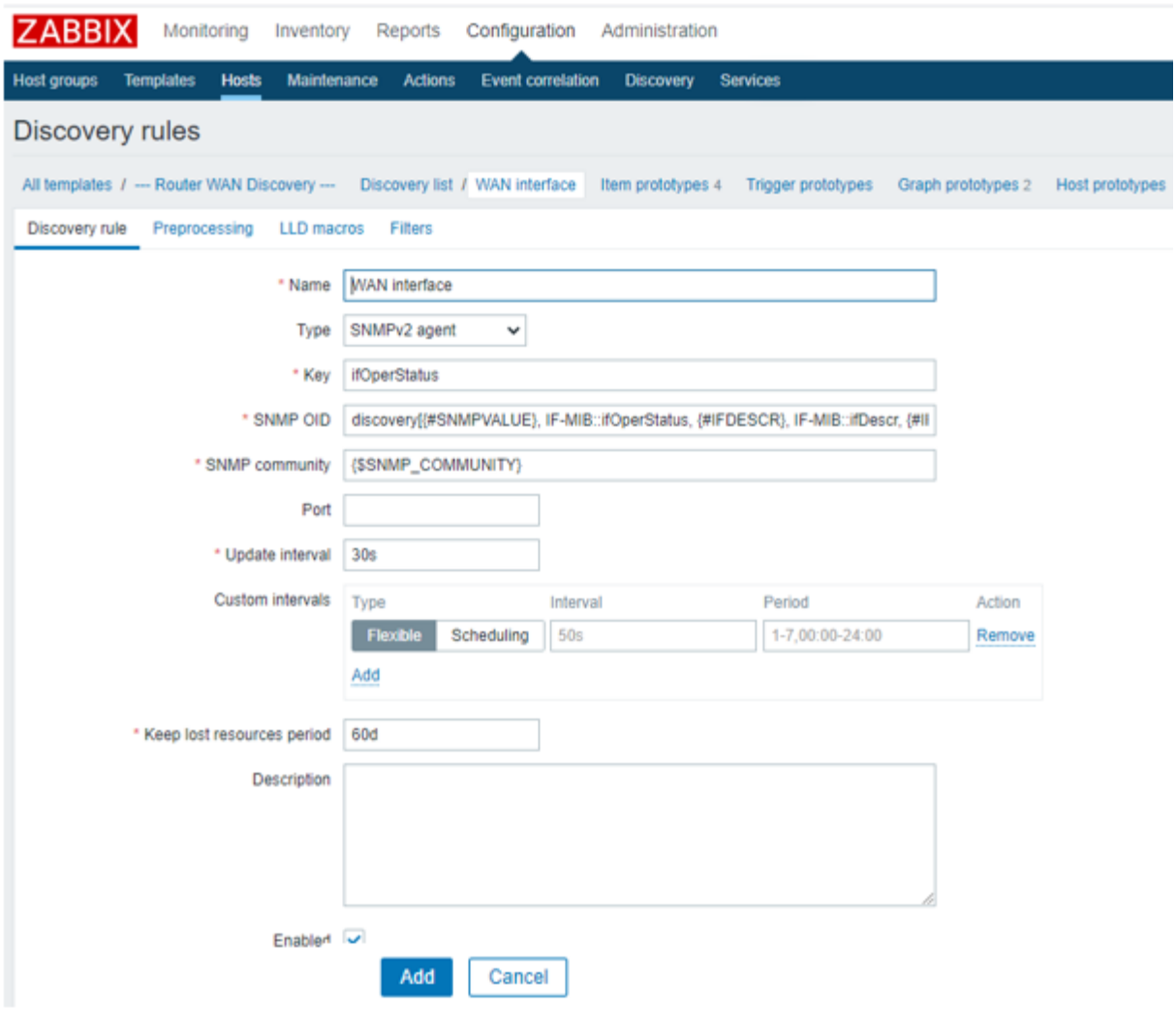
Slika 30 Prikaz povezivanja gotovog predloška unutar novog

Zatim je potrebno unutar predloška odabrati opciju *„Discovery rules“* kako bi se kreiralo pravilo za otkrivanje sučelja koje imaju u opisu „WAN“ i kliknuti na gumb *„Create disovery rule“* u gornjem desnom kutu kao što je vidljivo na slici 31.



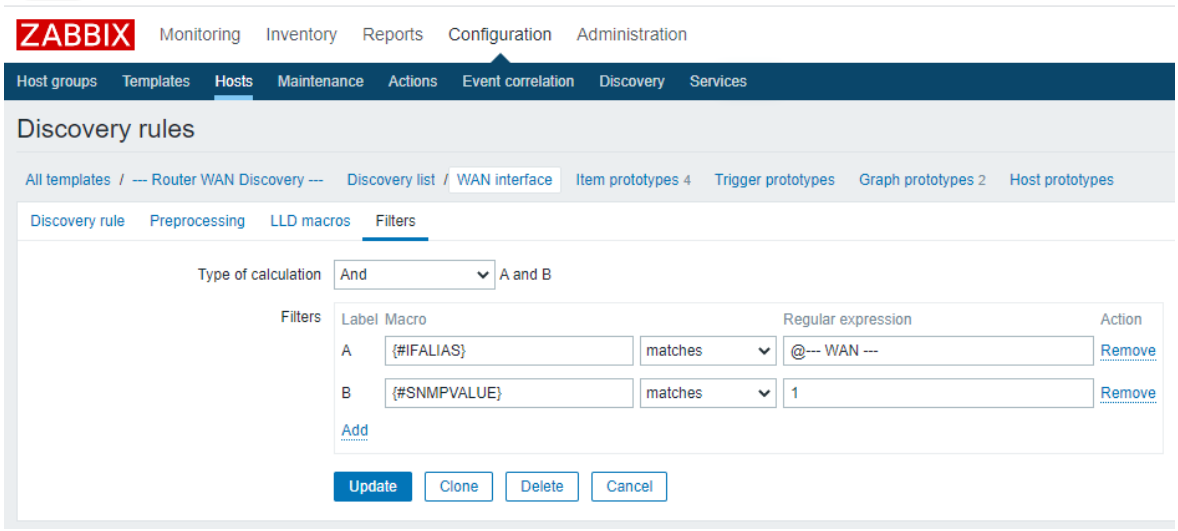
Slika 31 Prikaz kreiranja „Discovery rule“ unutar predloška

Nakon ulaska u formu *„Discovery rule“* unutar predloška potrebno je namjestiti ključne parametre kako bi pravilo za otkrivanje sučelja na usmjerivačima opisa „WAN“ bilo funkcionalno. Osim naziva pravila potrebno je postaviti ključne parametre za omogućavanje korištenja SNMP protokola za prikupljanje informacija sa CE usmjerivača. Potrebno je definirati *„SNMP OID“* koji definira koje će se informacije prikupljati sa CE usmjerivača. U navedenom slučaju na usmjerivaču će se gledati opis sučelja. Zatim je potrebno definirati varijablu za *„SNMP community“* takozvani zajednički znakovni niz koji je definiran na CE usmjerivaču, a omogućava pristup i čitanje, tj. prikupljanje informacija sa CE usmjerivača definiranih pomoću SNMP OID-a. Na slici 32 se nalazi prikaz postavljanje ključnih parametra koji su opisani u tekstu.



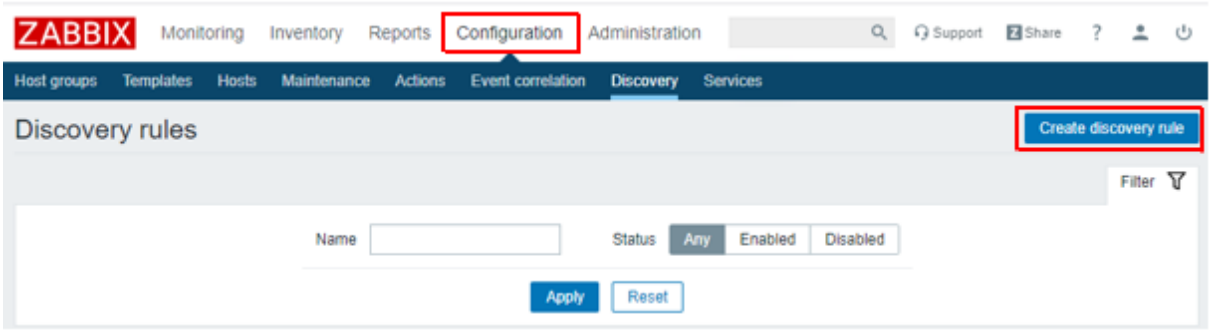
Slika 32 Prikaz postavljanja ključnih parametara u formi „Discovery rule“ unutar predloška

Nakon toga je potrebno unutar forme *„Discovery rule“* unutar predloška odabrati opciju *„Filters“* kako bi točno definirali uvjeti po kojima SNMP agent pretražuje sučelja usmjerivača. U navedenom slučaju kada SNMP agent pronađe navedeni opis „WAN“ na sučelju, prikuplja podatke s tog sučelja i na temelju njih će se kasnije crtati grafovi prometa i paketa. Na slici 33 je prikaz konfiguracije koji je opisan u tekstu.



Slika 33 Prikaz konfiguriranja filtera za pretraživanje sučelja na usmjerivaču

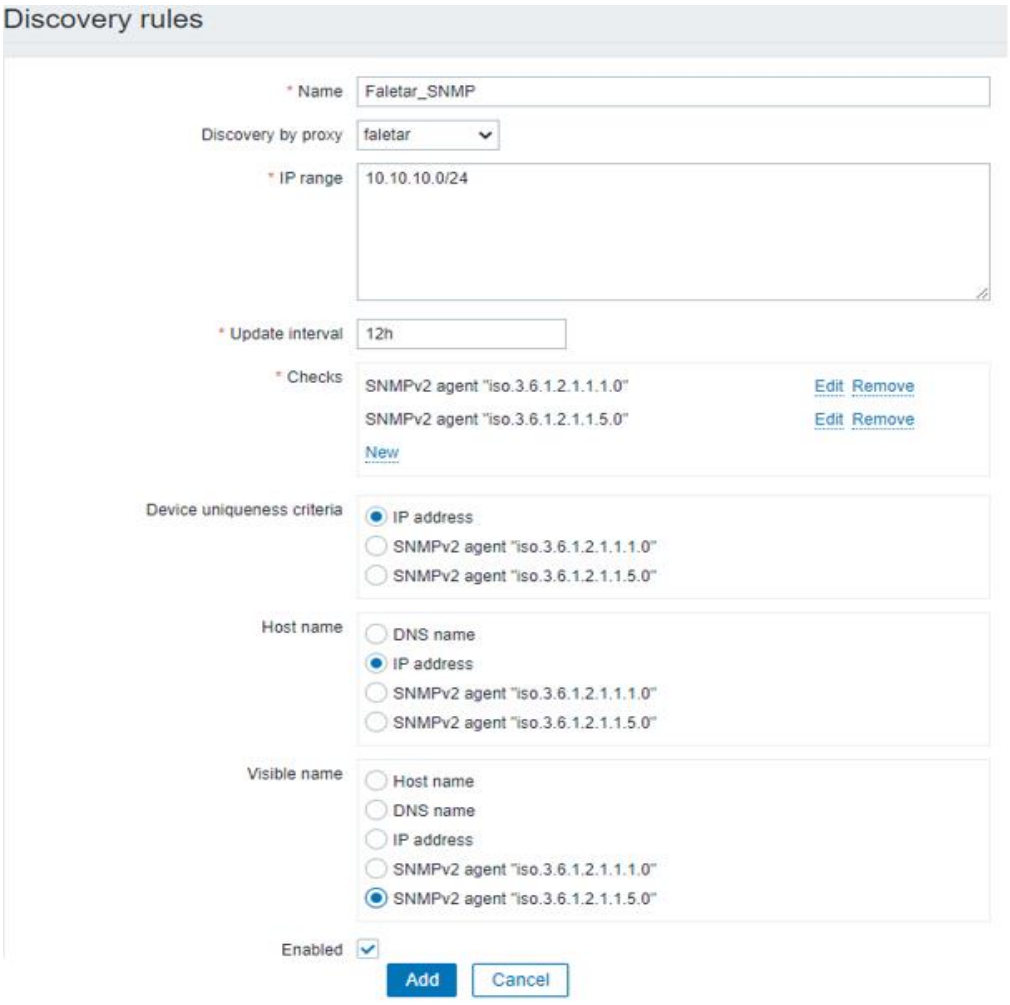
Nakon što je uspješno kreiran predložak, potrebno je kreirati *„Discovery rule“* takozvana pravila otkrivanja uređaja u mreži. Na temelju toga definirat će se unutar koje MPLS VPN mreže će se obavljati nadzor te prikupljati podaci i koje IP adrese se koriste unutar navedene MPLS VPN mreže. Za kreiranje pravila otkrivanja *(*engl. *discovery rule)* uređaja potrebno je u prvoj traci odabrati opciju *„Configuration“* nakon čega će u drugoj traci biti ponuđeno nekoliko opcija, a odabrat će se opcija *„Discovery“* te zatim u gornjem desnom uglu gumb *„Create discovery rule“* koji služi za ulaz u formu za kreiranje pravila otkrivanja uređaja. Na slici 34 je prikazan postupka za kreiranje *„discovery rule“*.



Slika 34 Prikaz postupka za kreiranje „discovery rule“

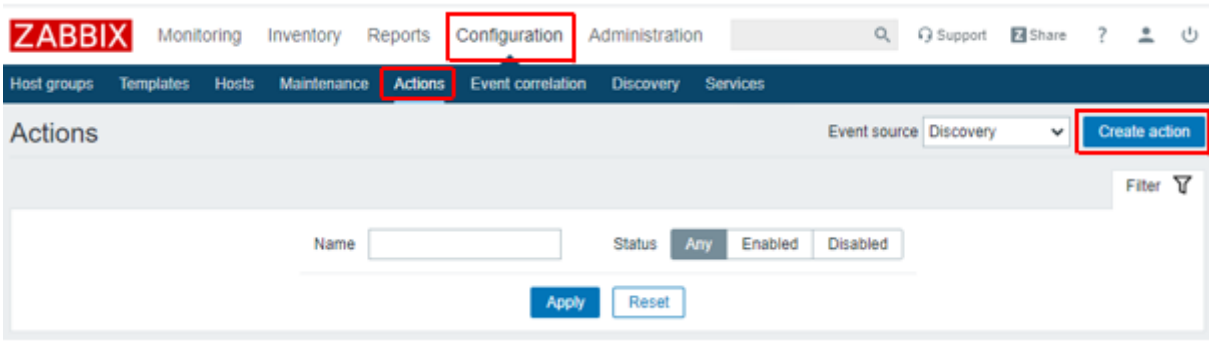
Nakon ulaska u formu za kreiranje *„discovery rule“,* takozvanih pravila za otkrivanje uređaja u mreži potrebno je osim naziva pravila otkrivanja, postaviti sljedeće parametre: iz padajućeg izbornika izabrati postojeći *proxy* servis koji je ranije kreiran i označava unutar koje MPLS VPN mreže će se obavljati nadzor uređaja, raspon IP adresa unutar kojeg se nalaze adrese CE usmjerivača promatrane MPLS VPN mreže, vremenski interval unutar kojega će se

ponavljati otkrivanje uređaja te postaviti SNMP OID-ove na temelju kojih će se prikupljati samo određeni podaci sa CE usmjerivača. U navedenom primjeru naziv pravila za otkrivanje će biti „Faletar_SNMP“, a iz padajućeg izbornika je odabran *proxy* servis naziva „faletar“. Unesen je raspon IP adresa 10.10.10.0/24 i postavljeno je razdoblje od 12 sati što označava da će se svakih 12 sati ponavljati proces otkrivanja uređaja. Definirani su SNMP OID-ovi iso.3.6.1.2.1.1.1.0 i iso.3.6.1.2.1.1.5.0 na temelju kojih će se sa CE usmjerivača prikupljati određeni podaci. OID iso.3.6.1.2.1.1.1.0 otkriva model i naziv uređaja (npr. Cisco ME3400), dok OID iso.3.6.1.2.1.1.5.0 otkriva konfigurirani naziv *(*engl. *hostname)* na CE usmjerivaču. Primjerice CE usmjerivač u poslovnici firme Faletar u Zagrebu ima konfiguriran *„hostname“*, tj. naziv usmjerivača „P1 Faletar - Zagreb MPLS“ te će taj naziv biti prikazan u Zabbixu što je definirano u navedenom primjeru kao parametar *„Visible name“*. Nakon unosa svih podataka i parametara potrebno je kreirati navedeno pravilo otkrivanja uređaja klikom na gumb *„Add“* koji se nalazi na dnu forme. Na slici 35 se nalazi prikaz forme i parametara koji su opisani u tekstu.



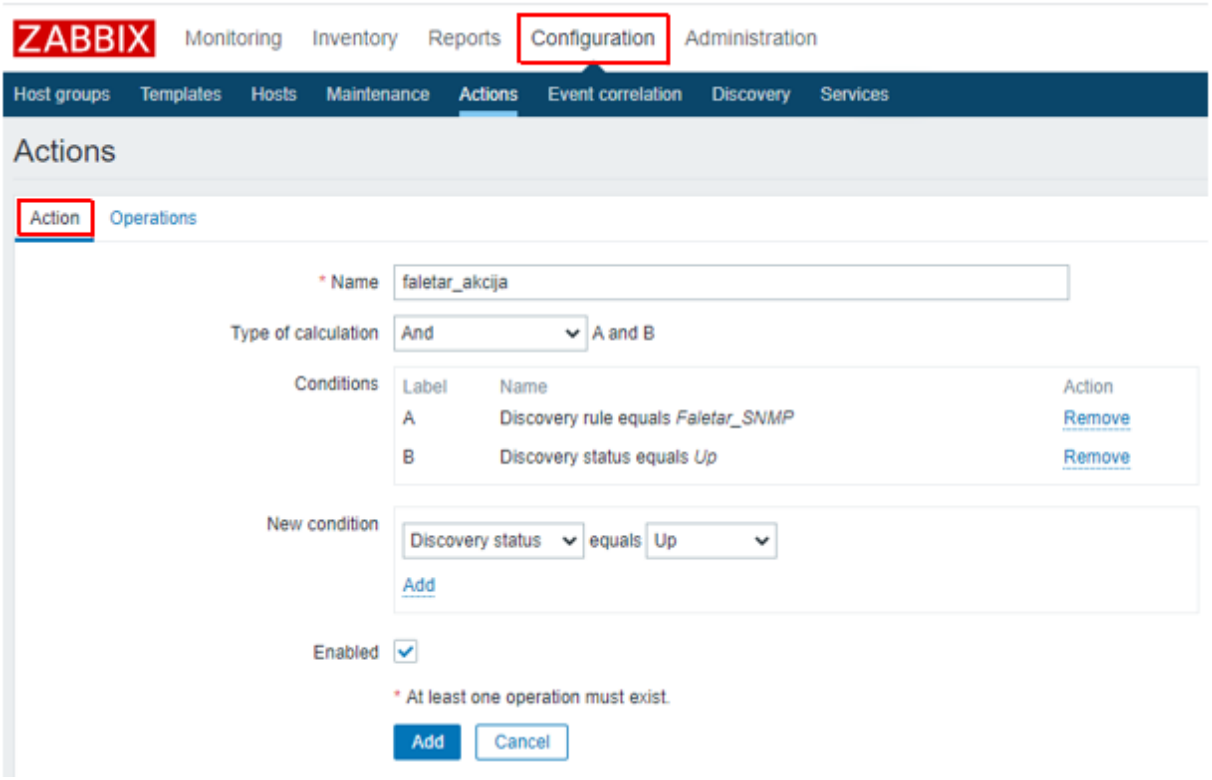
Slika 35 Prikaz forme i parametara za kreiranje pravila za otkrivanje uređaja

Nakon uspješnog kreiranja *„discovery-a“* potrebno je još kreirati akciju *(*engl. *action)* kako bi bilo moguće obavljati nadzor usmjerivača. Na temelju akcije moguće je pod određenim uvjetima otkrivene uređaje smjestiti u određenu *„host“* grupu, definirati koji će se predložak koristiti i ostalo. Za kreiranje akcije potrebno je u prvoj traci odabrati opciju *„Configuration“* nakon čega će u drugoj traci biti ponuđeno nekoliko opcija, a odabrat će se opcija *„Actions“* te zatim u gornjem desnom uglu gumb *„Create action“* koji služi za ulaz u formu za kreiranje akcije. Na slici 36 je prikazan je postupak koji je opisan u tekstu.



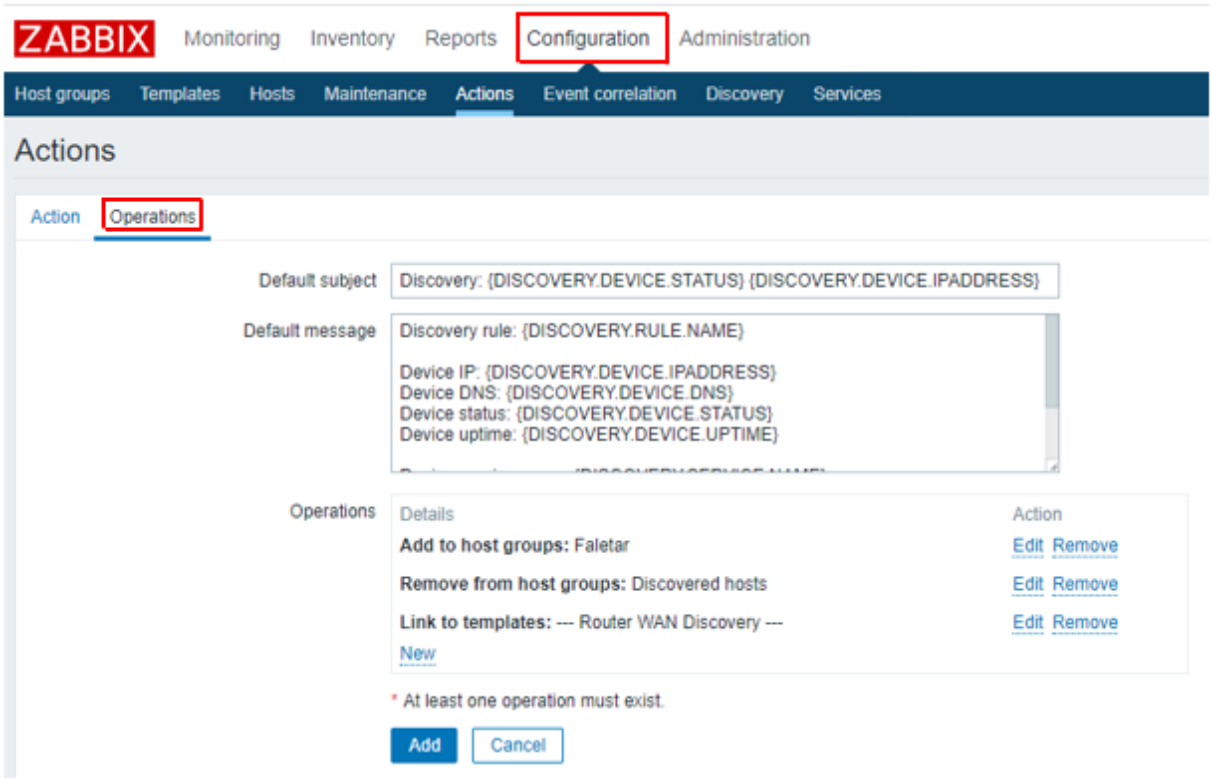
Slika 36 Prikaz postupka za kreiranje akcije

Nakon ulaska u formu za kreiranje akcije potrebno je definirati određene parametre. Osim samog naziva akcije potrebno je definirati i uvjete pod kojima će se akcija izvršavati samo ako su svi uvjeti zadovoljeni. U navedenom primjeru naziv akcije biti će „faletar_akcija“ te će biti postavljena dva uvjeta. Prvi uvjet odnosi se na ranije kreirano pravilo za otkrivanje uređaja naziva „Faletar_SNMP“ gdje se na temelju OID-a provjerava o kojem uređaj i modelu uređaja se radi (npr. usmjerivač Cisco 1812). Drugi uvjet provjerava je li otkriveni uređaj dostupan. Na slici 37 nalazi se prikaz forme i opisanih uvjeta.



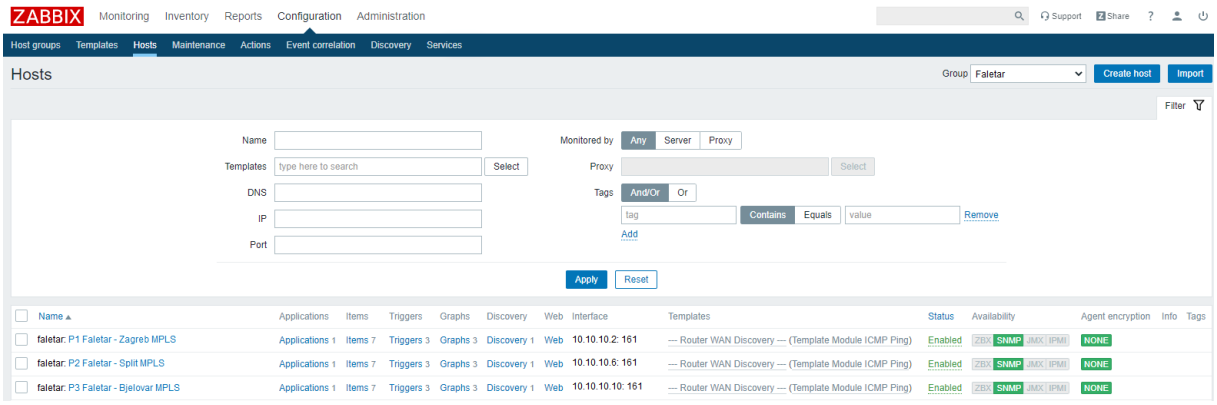
Slika 37 Prikaz definiranja uvjeta unutar forme akcije

Ukoliko su zadovoljeni uvjeti definirani unutar forme akcije na slici 37, potrebno je definirati koje radnje će se izvršavati. Radnje je potrebno definirati unutar forme akcije pod opcijom *„Operations“*. U navedenom primjeru, ako su zadovoljena oba uvjeta definirana na slici iznad, izvršit će se sljedeće radnje: otkriveni usmjerivač dodat će se u *„host“* grupu *„Faletar“*, izbrisat će otkriveni usmjerivač iz *„host“* grupe *„Discovered hosts“* unutar koje se inače spremaju svi otkriveni uređaji, te će se nad otkrivenim usmjerivačem primjenjivati ranije kreiran predložak *„Router WAN Discovery“* koji služi za crtanje grafova prometa i paketa sa sučelja usmjerivača, a čak sadrži i ugrađen predložak za provjeru dostupnosti uređaja. Nakon što se definiraju svi uvjeti i radnje unutar forme akcije prikazane na slici 38, potrebno je kreirati akciju klikom na gumb *„Add“* koji se nalazi na dnu forme.



Slika 38 Prikaz definiranja radnji unutar forme akcije

Ukoliko je implementacija i konfiguracija nadzora odrađena bez grešaka kroz sve korake i ukoliko su dostupni usmjerivači unutar definirane MPLS VPN mreže, usmjerivači bi unutar definiranog razdoblja od 12 sati trebali biti vidljivi u Zabbix sustavu za nadzor.



Slika 39 Prikaz otkrivenih usmjerivača unutar „host“ grupe Faletar

Na slici 39 nalazi se prikaz usmjerivača unutar *„host“* grupe „Faletar“ nad kojima se obavlja nadzor.

8. ZAKLJUČAK

Razvojem postojećih i pojavom novih tehnologija, računalne mreže su napredovale i postale sveprisutne u današnjim sustavima (poslovnim, obrazovnim, zdravstvenim, vojnim, itd.). Budući da komunikacijske mreže imaju sve veću ulogu u svim aspektima poslovanja, potrebno je omogućiti nadzor svih elemenata sustava. Nadzor komunikacijskih i računalnih mreža definira se kao praćenje stanja mreže i spojenih uređaja te detekcija problema unutar mrežnog sustava. Konfiguracija mreža podrazumijeva podešavanje parametara mrežnih uređaja i servisa u svrhu njihovog efikasnijeg rada. Mrežni protokoli korišteni za nadzor i konfiguraciju komunikacijskih mreža u ovome diplomskom radu su: SNMP, ICMP, SSH, TELNET.

Telekomunikacijski operatori za pružanje mrežnih usluga poslovnim korisnicima koji imaju više udaljenih poslovnica često koriste MPLS L3 VPN mreže. Udaljene poslovnice se spajaju koristeći virtualne privatne mreže koje su realizirane preko MPLS mreže. Unutar MPLS L3 VPN-a, informacije o usmjeravanju od jednog korisnika su u potpunosti odvojene od ostalih korisnika i proslijeđene preko MPLS mreže davatelja usluga. Aktualno pitanje s kojim se danas susreće većina telekomunikacijskih operatora je kako obavljati nadzor unutar MPLS L3 VPN mreže. U diplomskom radu je razvijeno i opisano rješenje za nadzor MPLS L3 VPN mreže uz pomoć Zabbix *proxy* servisa pokretanog unutar alata *Docker Engine* koristeći kontejnere, tzv. *dockere*. Kako bi nadzor bio uspješan, a samim time osigurano zadovoljstvo krajnjeg korisnika, prilikom razvijanja navedenog rješenja za nadzor MPLS L3 VPN mreža unaprijed je dogovoreno s korisnikom koji dio mreže i koje parametre želi imati pod nadzorom te je obraćena pozornost na važne elemente koji su osigurani, a to su: dostupnost, sigurnost, funkcionalnost i jednostavnost, rast nadzornog rješenja kao i sama prezentacija prema krajnjem korisniku.

Razvijeno rješenje omogućilo je uz minimalne troškove obavljanje nadzora MPLS L3 VPN mreža, čime se smanjuje vrijeme nedostupnosti mrežnih uređaja, a povećava dostupnost i efikasnost usluga koju pruža alternativni telekomunikacijski operator. U današnje vrijeme ispad pojedinog elementa mreže kod poslovnih korisnika može dovesti do velikih financijskih gubitaka, pogotovo ako je riječ o kvaru za čiju sanaciju je potrebno više vremena. Upravno zato razvijeno rješenje omogućava telekomunikacijskim operatorima obavljanje proaktivnog nadzora kako bi se kvar mogao predvidjeti ili uočiti u trenutku nastanka. Brzom reakcijom operatora smanjuje se vrijeme nedostupnosti mrežnih uređaja, a samim time povećava učinkovitost usluga i ono najvažnije - zadovoljstvo korisnika.

LITERATURA

[1] De Ghein L. MPLS Fundamentals. Indianapolis: Cisco Press; 2007.

[2] Mrvelj Š. Predavanja iz kolegija „Tehnologija telekomunikacijskog prometa II“. Zagreb: Fakultet prometnih znanosti; 2017.

[3] Lewis C, Pickavance S. Selecting MPLS VPN Services. Indianapolis: Cisco Press; 2006.

[4] Odom W. CCENT/CCNA ICND1 100-105 Official Cert Guide. Indianapolis: Cisco Press; 2016.

[5] Turnbull J. The Art of Monitoring. Brooklyn, New York: Turnbull Press; 2016.

[6] Olups R, Dalle Vacche A, Uytterhoeven P. Zabbix: Enterprise Network Monitoring Made Easy. Birmingham: Packt Publishing Ltd; 2017.

[7] Josephsen D. Building a Monitoring Infrastructure with Nagios. Boston: Pearson Education Inc; 2007

[8] URL: https://laptrinhx.com/multiprotocol-label-switching-mpls-explained-3878431457/ (pristupljeno: svibanj, 2020.)

[9] URL: https://tools.ietf.org/html/rfc5036 (pristupljeno: svibanj, 2020.)

[10] URL: https://networklessons.com/mpls/mpls-layer-3-vpn-explained (pristupljeno: svibanj, 2020.)

[11] URL: http://tfotovic.tripod.com/ni_protokoli.htm (pristupljeno: lipanj, 2020.)

[12] URL: https://www.researchgate.net/figure/ICMP-packet-structure_fig5_316727741 (pristupljeno: lipanj, 2020.)

[13] URL: https://www.cert.hr/wp-content/uploads/2009/08/CCERT-PUBDOC-2009-08-272.pdf (pristupljeno: lipanj, 2020.)

[14] URL: https://www.cert.hr/wp-content/uploads/2019/04/NCERT-PUBDOC-2010-09-313.pdf (pristupljeno: lipanj, 2020.)

[15] URL: https://www.incibe-cert.es/en/blog/snmp-it-simple-name-implies (pristupljeno: lipanj, 2020.)

[16] URL: https://www.ittsystems.com/snmp-monitoring-tools/ (pristupljeno: lipanj, 2020.)

[17] URL: https://www.nagios.com/ (pristupljeno: lipanj, 2020.)

[18] URL: https://www.cacti.net/what_is_cacti.php (pristupljeno: lipanj, 2020.)

[19] URL: https://en.wikipedia.org/wiki/Cacti_(software) (pristupljeno: lipanj, 2020.)

[20] Davidović G, Petričušić I, Žigman D. SUSTAV ZA UPRAVLJANJE I NADZOR RAČUNALNE MREŽE – ZENOSS. Polytechnic and design. 2014; 2 (1), 124-129. URL: https://doi.org/10.19279/TVZ.PD.2014-2-1-14 (pristupljeno: lipanj, 2020.)

[21] URL: https://commons.wikimedia.org/wiki/File:Zenoss_Core_Dashboard.png (pristupljeno: lipanj, 2020.)

[22] URL: https://www.zabbix.com/documentation/4.4/manual/introduction/about (pristupljeno: lipanj, 2020.)

[23] URL: https://www.sans.org/reading-room/whitepapers/networkdevs/how-to-securely-snmp-bgp-mpls-vpn-network-245 (pristupljeno, kolovoz, 2020.)

POPIS KRATICA I AKRONIMA

KRATICE	ZNAČENJE KRATICA
ASN	engl. Autonomous System Number
ATM	engl. Asynchronous Transfer Mode
BGP	engl. Border Gateway Protocol
BoS	engl. Bottom of Stack
CBC-DES	engl. Cipher Block Chaining-Data Encryption Standard
CE	engl. Customer edge
CLI	engl. Command Line Interface
CoS	engl. Class of Service
EIGRP	engl. Enhanced Interior Gateway Routing Protocol
FEC	engl. Forwarding Equivalence Class
FTP	engl. File Transfer Protocol
HMAC-MD5	engl. Hash-based Message Authentication Code- Message-Digest algorithm 5
HMAC-SHA	engl. Hash-based Message Authentication Code -Secure Hash Algorithm
ICMP	engl. Internet Control Message Protocol
IGP	engl. Interiror Gateway Protocol
IS-IS	engl. Intermediate System-to Intermediate System
ISP	engl. Internet Service Provider
LAMP	engl. Linux, Apache, MySQL, PHP
LDP	engl. Label Distribution Protocol
LFIB	engl. Label forwarding information base
LIB	engl. Label information base
LSP	engl. Label switched path
LSR	engl. Label Switch router
MIB	engl. Management Iniformation Base
MP-BGP	engl. Multiprotocol Border Gateway Protocol
MPLS	engl. Multi-Protocol Label Switching
MPLS QoS	engl. MPLS Quality of Service
MPLS TE	engl. MPLS Traffic Engineering

MPLS VPN	engl. MPLS Virtual Private Networks
MRTG	engl. Multi Router Traffic Grapher
NMS	engl. Network Management System
OID	engl. Object Identifier
OSPF	engl. Open Shortest Path First
P	engl. Provider
PE	engl. Provider edge
QoS	engl. Quality of Service
RBAC	engl. Role-based access control
RD	engl. Route Distinguisher
RDD	engl. Round-robin Database
RT	engl. Route Target
RTT	engl. Round Trip Time
SNMP	engl. Simple Network Management Protocol
SSH	engl. Secure Shell
TCP	engl. Transmission Control Protocol
TTL	engl. Time to Live
UDP	engl. User Datagram Protocol
VPN	engl. Virtual Private Network
VRF	engl. Virtual Routing and Forwarding

POPIS SLIKA

POPIS TABLICA

Sveučilište u Zagrebu
Fakultet prometnih znanosti
10000 Zagreb
Vukelićeva 4

IZJAVA O AKADEMSKOJ ČESTITOSTI I SUGLASNOST

Izjavljujem i svojim potpisom potvrđujem kako je ovaj diplomski rad isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu što pokazuju korištene bilješke i bibliografija.

Izjavljujem kako nijedan dio rada nije napisan na nedozvoljen način, niti je prepisan iz necitiranog rada, te nijedan dio rada ne krši bilo čija autorska prava.

Izjavljujem također, kako nijedan dio rada nije iskorišten za bilo koji drugi rad u bilo kojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

Svojim potpisom potvrđujem i dajem suglasnost za javnu objavu diplomskog rada pod naslovom NADZOR MPLS MREŽE ALTERNATIVNOG TELEKOMUNIKACIJSKOG OPERATORA na internetskim stranicama i repozitoriju Fakulteta prometnih znanosti, Digitalnom akademskom repozitoriju (DAR) pri Nacionalnoj i sveučilišnoj knjižnici u Zagrebu.

Student/ica:

U Zagrebu, 16.9.2020 Dino Faletar
(potpis)