



Univerzitet u Novom Sadu

*Tehnički fakultet „Mihajlo Pupin“ - Zrenjanin*

# **MODELIRANJE I REALIZACIJA VIRTUELNE MREŽNE LABORATORIJE BAZIRANE NA TEHNOLOGIJI ZA VIRTUELIZACIJU**

DOKTORSKA DISERTACIJA

Mentor: Prof. dr Borislav Odadžić

Kandidat: mr Dalibor Dobrilović

Zrenjanin, 2011. godine

UNIVERZITET U NOVOM SADU

TEHNIČKI FAKULTET „MIHAJLO PUPIN“ - ZRENJANIN

**UNIVERZITET U NOVOM SADU**

**TEHNIČKI FAKULTET „MIHAJLO PUPIN“ - ZRENJANIN**

**KLJUČNA DOKUMENTACIJSKA INFORMACIJA**

Redni broj: RBR	
Identifikacioni broj: IBR	
Tip dokumentacije: TD	Monografska dokumentacija
Tip zapisa: TZ	Tekstualni štampani materijal
Vrsta rada (dipl., mag., dokt.): VR	Doktorska disertacija
Ime i prezime autora: AU	Mr Dalibor Dobrilović
Mentor (titula, ime, prezime, zvanje): MN	Prof. dr Borislav Odadžić, redovni profesor
Naslov rada: NR	Modeliranje i realizacija virtuelne mrežne laboratorije bazirane na tehnologiji za virtuelizaciju
Jezik publikacije: JP	Srpski
Jezik izvoda: JI	srp. / eng.
Zemlja publikovanja: ZP	Srbija
Uže geografsko područje: UGP	
Godina: GO	2011.
Izdavač: IZ	autorski reprint
Mesto i adresa: MA	Đure Đakovica bb, 23000 Zrenjanin

Fizički opis rada: FO	7 poglavlja / 170 stranica / 50 slika / 3 grafikona / 193 referenci / 4 priloga
Naučna oblast: NO	Informacione tehnologije
Naučna disciplina: ND	Računarske mreže
Predmetna odrednica, ključne reči: PO	tehnologija za virtuelizaciju, virtuelne računarske mrežne laboratorije, IT i inženjerska edukacija
UDK	
Čuva se: ČU	Tehnički fakultet "Mihajlo Pupin" – Zrenjanin
Važna napomena: VN	
Izvod: IZ	<p>U fokusu ove disertacije je modeliranje i realizacija virtuelne mrežne laboratorije bazirane na tehnologiji za virtuelizaciju. Istraživanje je motivisano razvojem IKT i TCP/IP mreža što je uticalo na važnošću edukacije i stvaranja eksperata u toj oblasti. U početnoj fazi istraživanja, razvijen je model laboratorije sa lokalnim pristupom. Zajedno sa realizacijom laboratorije na osnovu modela, modelirana su i virtuelna mrežna scenarija za upotrebu u nastavi. Sledeća faza istraživanja bila je usmerena u pravcu modeliranja virtuelne mrežne laboratorije sa pristupom na daljinu. Prikazana je arhitektura modela laboratorije, softver i hardver potreban za njenu realizaciju, kao i mrežna scenarija prilagođena novom okruženju. Model je vrednovan posle određenog perioda upotrebe u nastavi iz predmeta Računarske mreže na visokoškolskoj ustanovi. Prvi pristup vrednovanju VNLab sistema je sproveden upotrebom novorazvijenog metoda zasnovanog ne teoriji grubih skupova. Ova metodologija, zajedno sa predloženim modelom, predstavlja jedno od naučnih rezultata disertacije i omogućava predavaču da analizira rad studenata i da uoči razlike između različitih nivoa studentskog znanja. Takođe, ova metoda pomaže predavaču u identifikaciji delova u materijalu za vežbe koji su malo razumljivi studentima. Dalje, virtualna mrežna laboratorija je vrednovana uz pomoć dva kontrolna testa. Treći pristup u</p>

	<p>evaluaciji modela izvršen je upotrebom formativne procene. Na kraju je ispitana upotrebljivost VNLab sistema poređenjem sa sadržajem preporučenih programa od strane ACM, IEEE i AIS organizacija za 4 obrazovna profila iz domena računarskih nauka: Računarstvo, Računarsko inženjerstvo, Informacione tehnologije i Informacioni sistemi.</p>
<p>Datum prihvatanja teme od strane NN veća: DP</p>	
<p>Datum odbrane: DO</p>	
<p>Članovi komisije: (ime i prezime / titula / zvanje / naziv organizacije / status) KO</p>	<p>predsednik: Dr Slobodan Janković, red. profesor          član: Dr Željko Trpovski, van. profesor          član: Dr Miodrag Ivković, van. profesor          član: Dr Milan Janković, docent          mentor: Dr Borislav Odadžić, red. profesor</p>

**University of Novi Sad**

**Technical faculty "Mihajlo Pupin" - Zrenjanin**

**Key word documentation**

Accession number: ANO	
Identification number: INO	
Document type: DT	Monograph documentation
Type of record: TR	Textual printed material
Contents code: CC	
Author: AU	Dobrilović Dalibor, MSc
Mentor: MN	Borislav Odadžić, PhD
Title: TI	Modeling and realization of virtual network laboratory based on virtualization technology
Language of text: LT	Serbian
Language of abstract: LA	eng. / srp.
Country of publication: CP	Serbia
Locality of publication: LP	
Publication year: PY	2011
Publisher: PU	Author reprint
Publication place: PP	Zrenjanin

Physical description: PD	7 chapters / 170 pages / 50 figures / 3 charts / 193 references / 4 appendixes
Scientific field SF	Information technology
Scientific discipline SD	Computer Networks
Subject, Key words SKW	virtualization technology, virtual computer network laboratories, IT and engineering education
UC	
Holding data: HD	Technical Faculty "Mihajlo Pupin" – Zrenjanin
Note: N	
Abstract: AB	In this Thesis the research is focused on modeling and realization of virtual network laboratory based on virtualization technology. This work is motivated with the growing development of ICT technologies and TCP/IP networks which resulted with the growing importance of educated and well trained experts in this field. In the beginning phase of the research, the model of virtual network laboratory with local access is developed. Together with the model and its realization the virtual network scenarios used for learning were modeled as well. The next phase of research is focused on modeling virtual network laboratory with remote access. The architecture of the laboratory model is presented, as well as the hardware and software components of the laboratory and network scenarios. The model is evaluated after the period of usage. The first evaluation of VNLab is conducted using newly developed methodology based on rough set theory. This methodology is one of the scientific results of this work and allows lecturer to analyze student work and to find out the differences between various levels of student knowledge. Also, it allows lecturer to identify parts in exercise material which is not understandable by students. Furthermore, the virtual network laboratory model is evaluated with two control tests. The third method of evaluation of the model is done with usage of formative assessment. At the end, usability of

	VNLab environment in various Computer Science based curriculums is shown by comparing VNLab supported features with recommended curriculums of ACM, IEEE and AIS organizations for 4 areas of computer science: Computer Science, Computer Engineering, Information Technology and Information Systems.
Accepted on Scientific Board on: AS	
Defended: DE	
Thesis Defend Board: DB	<p>president: Slobodan Janković, PhD, Full Professor</p> <p>member: Željko Trpovski, PhD, Professor</p> <p>member: Miodrag Ivković, PhD, Professor</p> <p>member: Milan Janković, PhD, Assistant professor</p> <p>mentor: Borislav Odadžić, PhD, Full professor</p>

*“Non nobis Domine, non nobis, sed Nomini Tuo da gloriam”*

*Stari zavet, Psalam 115:1*

*(srp. Ne nama, Gospode, ne nama, nego imenu svom daj slavu)*

*Koristim priliku da se zahvalim svima koji su mi pomogli u izradi ove disertacije. Zahvaljujem se mr Željku Stojanovu, dugogodišnjem prijatelju i kolegi, sa kojim sam u dugom periodu u toku izrade ove disertacije sarađivao, razmenjivao mišljenja i ideje. Rezultati našeg zajedničkog rada su u velikoj meri integrisani u ovoj disertaciji. Zahvaljujem se kolegi doc. dr Vladimiru Brtki, na saradnji i pomoći koje sam imao u pogledu teorije grubih skupova i relacije nerazberivosti (poglavlje 6.2.). Zahvaljujem se kolegici doc. dr Vesni Jevtić na korisnim savetima i pomoći oko metodološkog okvira rada. Zahvaljujem se prof. dr Borislavu Odadžiću, mentoru, na smernicama i pomoći u toku izrade ove disertacije. Za ukazano strpljenje i poverenje zahvaljujem se supruzi Ljiljani i majci Spomenki. Disertaciju posvećujem bratu Daši.*



## Sadržaj

1. Uvod	12
2. Ciljevi istraživanja i hipoteze	16
2.1. Problem i predmet istraživanja	16
2.2. Cilj istraživanja	18
2.3. Zadaci istraživanja	18
2.4. Hipoteze istraživanja	18
2.5. Očekivani rezultati istraživanja	19
2.6. Metode i tehnike istraživanja	19
2.7. Uzorak	20
2.8. Postupci i instrumenti za vrednovanje modela	20
3. Dosadašnja istraživanja	22
3.1. Tehnologija za virtuelizaciju	22
3.2. Virtuelne laboratorije zasnovane na tehnologiji za virtuelizaciju	25
3.3. Softverske komponente virtuelnih laboratorija	32
4. Modeliranje i realizacija virtuelne mrežne laboratorije sa lokalnim pristupom	37
4.1. Modeliranje laboratorije sa lokalnim pristupom	37
4.2. Modeliranje mrežnih scenarija za virtuelnu mrežnu laboratoriju	40
4.2.1. Modeliranje scenarija za RIP protokol	41
4.2.2. Modeliranje scenarija za OSPF protokol	46
4.3. Modeliranje ostalih scenarija za virtuelnu mrežnu laboratoriju	53
4.4. Testiranje performansi virtuelne laboratorije	54
4.5. Anketiranje studenata o radu sa virtuelnom laboratorijom	55
4.6. Analiza rada virtualne mrežne laboratorije sa lokalnim pristupom	57
5. Modeliranje i realizacija virtuelne mrežne laboratorije sa pristupom na daljinu	58
5.1. Modeliranje virtuelne mrežne laboratorije sa pristupom na daljinu	58

5.2. VNLab portal	60
5.3. Hardverske komponente modela	61
5.4. Konfiguracija i administracija VNLab okruženja	62
5.5. Modeliranje laboratorijskih scenarija u VNLab okruženju	64
5.6. VNLab vežba - IPv6	66
5.7. VNLab vežba - OSPF (Open Shortest Path First) sa jednim područjem	71
5.8. VNLab vežba - OSPF (Open Shortest Path First) sa više područja	78
5.9. VNLab vežba - BGP-4	80
5.10. VNLab vežba – Firewall	83
5.11. Iskustva u radu sa VNLab okruženjem	86
6. Vrednovanje modela virtuelne mrežne laboratorije sa pristupom na daljinu	88
6.1. Postojeći primeri vrednovanja virtuelnih laboratorija	88
6.2. Vrednovanje modela upotrebom metoda zasnovanog na teoriji grubih skupova	89
6.2.1. Relacija nerazberivosti i teorija grubih skupova	91
6.2.2. Redukcija podataka i sinteza if-then pravila	93
6.2.3. Eksperiment	94
6.2.4. Analiza podataka	95
6.3. Vrednovanje modela upotrebom kontrolnih testova	97
6.3.1. Kontrolni test br. 1	97
6.3.2. Kontrolni test br. 2	100
6.4. Vrednovanje modela na osnovu formativne procene	102
6.5. Procena šire upotrebljivosti VNLab laboratorije	105
6.5.1. Profil Informacione tehnologije	105
6.5.2. Profil Računarstvo	106
6.5.3. Profil Računarsko inženjerstvo	107
6.5.4. Profil Informacioni sistemi	108
6.5.5. Zaključak	109

7. Zaključna razmatranja	110
7.1. Rezultati istraživanja	110
7.2. Iskustva, prednosti i uočeni nedostaci	112
7.3. Zaključak	112
Literatura	114
Radovi objavljeni u toku izrade doktorske disertacije	128
Summary	129
Rečnik manje poznatih izraza	133
Dodatak A – scenario br.2 (IP adresiranje i statičko rutiranje – II deo)	138
Dodatak B – scenario br.3 (IPv6 adresiranje i statičko rutiranje)	149
Dodatak C – scenario br.7 (BGP - Border Gateway Protocol)	158
Dodatak D – scenario br.8 (Zaštita računarskih mreža - Firewall sistemi)	164

## 1. Uvod

U poslednjih nekoliko decenija svet u kome živimo doživljava izuzetno brz tehnološki napredak. Taj tehnološki napredak izražen je u mnogim sferama ljudskog društva, a jedna od svakako najznačajnijih i najočiglednijih manifestacija tog napredka ogleda se u oblasti informaciono-komunikacionih tehnologija (IKT). Značaj napretka u ovoj oblasti ogleda se i u njegovom uticaju na druge aspekte društva.

Umrežavanje računara i komunikacione tehnologije danas imaju veliki značaj. Internet je od čitavog sveta napravio globalnu komunikacionu magistralu. Od nastanka Interneta do današnjih dana, Internet doživljava konstantnu ekspanziju. Na njega se povezuje sve više i više čvorova, a pored značaja samog Interneta potrebno je napomenuti i razvoj i eksploataciju mreža koje su bazirane na TCP/IP skupu protokola. TCP/IP skup protokola danas dominira na svim područjima, od lokalnih računarskih mreža (LAN), do globalne svetske mreže (Internet). Nove tehnologije omogućuju osim prenosa podataka, prenos govora (VoIP), TV i video usluga (IPTV) i dr, a novi načini upotrebe računarskih resursa kao što su Cloud i Grid računarstvo se takođe pojavljuju.

Prema podacima Miniwatts Marketing Group u svetu trenutno postoji 2,095,006,005 Internet korisnika, što predstavlja 30.2% svetske populacije. U odnosu na 2000. god. rast Internet korisnika iznosi 480.4% (tada je broj Internet korisnika bio 360,985,492) [Wor11]. Prema podacima Republičkog zavoda za statistiku u Republici Srbiji u 2011. god. 79,5% preduzeća poseduje ožičeni LAN, a 46,9% preduzeća Wireless LAN. Dalje, 96,8% preduzeća je povezano na Internet, što je za 2,3% više u odnosu na 2009. godinu, 5,3% više u odnosu na 2008., a 6,2% više u odnosu na 2007. godinu. Tehnologije povezivanja su sledeće: xDSL vezu ima 74,1% preduzeća, kablovski Internet 26,5%, tradicionalni modem 13,5%, a mobilne veze 12,2% preduzeća u Srbiji [Sta11].

Uporedo sa širenjem, razvojem i nastankom novih tehnologija raste i značaj osposobljenih stručnjaka za rad sa njima, kako u svetu, tako i u našoj zemlji. Obučenosť eksperata i njihov kvalitet u velikoj meri se zasnivaju na uspostavljanju pouzdanog i efikasnog sistema njihovog podučavanja i obuke. To podučavanje se prvenstveno vrši na univerzitetima, u sklopu redovne nastave i programa na odgovarajućim smerovima. Takođe, podučavanje se može vršiti i u okviru dopunskih kurseva i obuka diplomiranih studenata i ljudi koji su zaposleni.

Zajedno sa isticanjem važnosti obuke i učenja ovih stručnjaka, nemoguće je ne razmišljati o procesu nastave i obuke koji ovim ekspertima treba da pruži odgovarajuća znanja. Efikasnost nastave meri se pragmatičnim znanjima koji studenti IKT tehnologija i eksperti moraju da imaju. U želji da se postigne potreban nivo praktičnih znanja, mora se razmišljati i o načinu postizanja tog cilja. U toku vršenja nastave na visokoškolskim ustanovama praktičan rad se ostvaruje kroz laboratorijske vežbe. Postojanje adekvatnih i efikasnih laboratorija predstavlja izuzetan problem koji ima veliki uticaj na krajnji ishod nastave.

Govoreći o efikasnim laboratorijama, moramo analizirati kako se praktičan laboratorijski rad može organizovati na univerzitetu. Jedan od najefikasnijih i najrealnijih okruženja za rad predstavljaju stvarne mrežne laboratorije. One mogu da pruže veliki stepen praktičnih znanja

studentima, ali sa sobom nose i odgovarajuća ograničenja. Prvo ograničenje koje se nameće u ovakvim okruženjima je njihova visoka cena. Ta cena proističe iz visokih cena opreme, naročito one koja je novija na tržištu i koja ima podršku za nove tehnologije. Drugi, ne tako zanemarljiv faktor, koji utiče na visoke cene ovakvih laboratorijskih sistema je relativno brzo zastarevanje opreme. Kako se brzo razvijaju tehnologije, tako se na tržištu pojavljuju noviji modeli uređaja sa boljim performansama. Da bi laboratorija mogla i dalje da se koristi za učenje savremenih mrežnih tehnologija, mora da se vrši njena konstantna nadogradnja.

Druga vrsta ograničenja ovih laboratorija ogleda se u sledećem: kada se formira jedna ovakva laboratorija, sa podrškom za učenje planiranih mrežnih koncepata i tehnologija, postoji opasnost da je ona suviše platformski zavisna, tj. orijentisana ka tehnologijama i platformama koje nameće proizvođač čija je oprema kupljena.

Treći ograničavajući faktor predstavlja limitiranost upotrebe ove laboratorije u pogledu broja studenata koji mogu istovremeno da rade u ovom okruženju, kao i lokacije sa koje oni mogu da pristupaju. Laboratorija je smeštena u određenom fizičkom okruženju, a tom fizičkom okruženju može da pristupa samo određeni broj studenata istovremeno i u terminima koji su obično raspoređeni u toku nastavnog ciklusa.

Na taj način, fizička mrežna laboratorija sa stvarnom mrežnom opremom pored svojih ogromnih prednosti u pogledu pružanja realnih uslova za učenje, ima i svoja ograničenja koja umanjuju njenu efikasnost. Alternativa za ovakav tip laboratorije je prvenstveno softverskog karaktera. Kao alternativa mogu se koristiti softveri za simulaciju računarskih mreža. U današnje vreme i ova grupa programa doživljava sve veći napredak. Programski paketi, kao što su IT Guru [TBL02, TLP03, PTR02, PTT01], Ns [WJ09, PMS10, WLW09, Ns11] ili OMNeT++ [WGG10], predstavljaju efikasna okruženja za rad, simulaciju, učenje i obuku. Takvi softveri se koriste u nastavi u okviru redovnih studentskih programa na velikom broju univerziteta.

Pored njihove rasprostranjenosti, efikasnosti i prvenstveno niske cene, jedno od glavnih nedostataka ovih programa jeste nedovoljni stepen realnosti koji oni pružaju svojim korisnicima tj. studentima. Upotrebom ovakvih softvera, studenti mogu da se suoče sa velikim brojem različitih problema, situacija i zadataka, ali je nivo sličnosti ovih okruženja sa stvarnim sistemima mali.

U skorije vreme, uporedo sa ponovnim naglim razvojem i ekspanzijom jedne stare tehnologije, razvoj laboratorija za računarske mreže i komunikacije dobija novu dimenziju. Tehnologija koja je to omogućila jeste tehnologija za virtuelizaciju. Zahvaljujući ponovnom razvoju i načinu upotrebe ove tehnologije dolazi do razvoja novih platformi za učenje mrežnih koncepata. To su virtuelne mrežne laboratorije koje su bazirane na tehnologiji za virtuelizaciju. U prvom redu, može da zbuni dvostruka upotreba reči virtuelan. Upotreba je potpuno opravdana. Prva reč odnosi se na virtuelne mrežne laboratorije. Virtuelne mrežne laboratorije predstavljaju virtuelna okruženja za učenje računarskih mreža i koncepata vezanih za te tehnologije kao i druge tehnologije koje koriste isto okruženje, npr. operativni sistemi, serverski i aplikativni mrežni softver i dr. Ove laboratorije mogu biti organizovane pomoću simulacionog softvera, ali i pomoću drugih tehnologija.

Okruženje koje je opisano u ovom radu, bazirano je na tehnologiji za virtuelizaciju. Kao što je napomenuto, radi se o relativno staroj tehnologiji koja je svoj ponovni razvoj doživela u zadnjih

deset godina. Poslednjih godina ona doživljava najveći razvoj i spada u sam vrh modernih tehnologija. Tehnologija za virtuelizaciju omogućava rad virtuelnih računara i to više njih na jednom fizičkom računaru. O karakteristikama ove tehnologije, biće reči u narednim poglavljima. Ukratko treba napomenuti da baš njeno svojstvo koje omogućava da se na jednom računaru instalira i konfigurira veliki broj virtuelnih računara sa različitim operativnim sistemima i softverom, daje tehnološku osnovu za uspostavljanje virtuelnih računarskih mreža.

U ovoj disertaciji opisan je pristup u modeliranju i realizaciji virtuelne mrežne laboratorije bazirane na tehnologiji za virtuelizaciju i ispitana je njena upotrebljivost i efikasnost u nastavi na visokoškolskoj ustanovi. Model predstavlja sredstvo za opisivanje najbitnijih karakteristika sistema koji se proučava. Sistem je skup stavki ili osobina koje predstavljaju zaokruženi deo realnog fenomena koji se proučava, u ovom slučaju fenomena virtuelne mrežne laboratorije. Model mora posedovati prikaz objekata unutar sistema, tzv. komponenti sistema, kao i prikaz aktivnosti pod kojima će ti objekti međusobno delovati. Prema tome, model reflektuje razumevanje realnog procesa, njegovih komponenti i njihove interakcije koje potiču od strane samog modelara [Kov10]. Model je svaki teorijski, tj. pojmovni ili stvarni, predmetu istraživanja analogni sistem ( $S_0$ ), pomoću koga se istražuje izvestan osnovni predmet ili sistem ( $S_1$ ), gde je sistem  $S_0$  original ili uzor, a  $S_1$  model (sistem analogan originalu) [Šeš71]. Modeliranje sistema predstavlja proces kojim se opisuje, vizuelizuje, analizira i transformiše arhitektura sistema. Sistem se u ovom slučaju sastoji od softverskih i hardverskih komponenti i veza između njih. Modeliranje sistema se koristi da pomogne u razvoju i održavanju velikih sistema, sa fokusom na fazu konstrukcije [Eis96]. U cilju vizuelizacije veza koje postoje između promenljivih u modelu koristi se grafička reprezentacija modela. U nedostatku standardnih tehnika, za opis modela virtuelne mrežne laboratorije primenjeni su standardni mrežni dijagrami i dijagrami za opis tehnologije za virtuelizaciju i softverskih komponenata.

Struktura rada je sledeća: u drugom poglavlju pod naslovom "Ciljevi istraživanja i hipoteze", posle uvoda, navodi se metodološki okvir rada. Navode se ciljevi i zadaci istraživanja, postavlja se hipoteza kao i način njenog dokazivanja. U narednom poglavlju, "Dosadašnja istraživanja", prikazane su bitne komponente virtuelnih laboratorija. Prvenstveno se obraća pažnja na samu tehnologiju za virtuelizaciju, njene karakteristike kao i veliki broj proizvoda iz te oblasti koji se mogu naći na tržištu. Na kraju poglavlja navedene su softverske komponente virtuelnih mrežnih laboratorija koje predstavljaju drugi značajan faktor u kreiranju ovih okruženja pored tehnologije za virtuelizaciju.

Sledeće poglavlje "Modeliranje i realizacija virtuelne mrežne laboratorije sa lokalnim pristupom", podeljeno je na nekoliko celina. U prvom delu se vrši prikaz prvih koraka u modeliranju ovog okruženja, potreba za njegovim nastankom, kao i nastanak i razvoj ideje. U nastavku se opisuje modeliranje scenarija za laboratoriju, njena upotreba, zajedno sa prikazom rezultata i analizom iskustava u njenoj upotrebi. Na kraju poglavlja, iznete su sve poteškoće u radu sa laboratorijom koje su direktno uticale na stvaranje novog modela i njegove specifične arhitekture i prelazak na VNLab.

Poglavlje "Modeliranje i realizacija virtuelne mrežne laboratorije sa pristupom na daljinu" prikazuje arhitekturu novonastalog modela virtuelne mrežne laboratorije sa pristupom na daljinu, njegove hardverske i softverske komponente. Zatim, opisuje VNLab portal, njegovu strukturu i namenu. Opisana su važna tehnička pitanja kao što su konfiguracija, administracija i održavanje laboratorije. Modeliranje i prilagođavanje karakterističnih vežbi za novo okruženje i

njihova struktura prikazani su u narednom delu poglavlja. Na kraju, opisane su onlajn laboratorijske vežbe u VNLab okruženju kroz IPv6, OSPF i BGP-4 protokole i fajervol sisteme (eng. firewall; srp. mrežna barijera, vatreni zid).

Poglavlje „Vrednovanje modela virtuelne mrežne laboratorije sa pristupom na daljinu“ sadrži u sebi najznačajniji deo istraživanja. Ono prikazuje iskustva i načine vrednovanja VNLab okruženja koja se nameću posle njene višegodišnje upotrebe. Nakon kratkog osvrtu na moguće metode i alate vrednovanja laboratorije, predstavlja se opis prve metode za njeno vrednovanje. Radi se o metodi evaluacije modela zasnovanoj na teoriji grubih skupova. U sklopu te sekcije objašnjava se relacija nerazberivosti i teorija grubih skupova, koje predstavljaju osnove ovog evaluacionog metoda. Nakon toga, objašnjava se redukcija postojećih podataka, njihovo smanjenje na odgovarajući broj i sinteza „if-then“ pravila koja su rezultat ove metode. U sekciji „Eksperiment“ opisuje se primena ovog metoda na analizu rada VNLab-a. Sledi vrednovanje laboratorije upotrebom kontrolnih testova i na osnovu formativne procene. Na kraju poglavlja, data je procena opšte upotrebljivosti laboratorije VNLab u poređenju sa različitim profilima i nastavnim programima koji su preporučeni u nastavi računarstva od strane sledećih svetskih organizacija: ACM (Association for Computer Machinery), IEEE (Institute of Electrical and Electronics Engineers) i AIS (Association for Information Systems).

U završnom poglavlju prikazuju se završna razmatranja. Ocenjuje se upotrebljivost laboratorije i dokazuje polazna hipoteza. Navode se uočeni nedostaci i načini da se oni prevaziđu. Kao dalji pravci razvoja navode se moguća proširenja ove laboratorije, promena njene namene i upotreba novih tehnologija.

Posle završnog poglavlja navode se literaturni izvori koji su korišćeni u toku izrade disertacije. Literaturni izvori su stranog porekla, pretežno na engleskom jeziku, uz nekoliko izvora na srpskom (šest), španskom (jedan) i portugalskom jeziku (jedan izvor). Kao literaturni izvori pretežno su korišćene knjige, radovi iz indeksiranih časopisa i sa međunarodnih konferencija. Veb sajtovi su navođeni kao literaturni izvori u slučaju softverskih proizvoda i pojedinih postojećih rešenja u domenu virtuelnih laboratorija, kada nisu postojali drugi relevantni izvori. Zatim su posebno su izdvojeni radovi koji su objavljeni od strane autora disertacije, a koji su proistekli iz istraživanja i rada na njoj. Sledi rečnik manje poznatih reči i izraza i sažetak (summary) na engleskom jeziku.

U dodacima A, B, C i D priložene su vežbe u obliku u kojem se daju studentima u nastavi. U dodacima su priložene karakteristične vežbe, a neke od njih su slične sa vežbama koje su opisane u poglavlju 5. Pored te sličnosti, u dodacima su priložene vežbe u originalnom obliku, sa komentarima i informacijama koje se daju studentima sa ciljem efikasnijeg učenja i sticanja praktičnih znanja. Te vežbe su: IPv4 adresiranje i statičko rutiranje, IPv6 adresiranje i statičko rutiranje, BGP-4 protokol i fajervol.

## 2. Ciljevi istraživanja i hipoteze

### 2.1. Problem i predmet istraživanja

Problem ovog istraživanja jeste nastava na visokoškolskim ustanovama na predmetu Računarske mreže, ali i sličnim predmetima koji u okviru svog izučavanja imaju koncepte za koje je potrebna mrežna infrastruktura. Kada je potrebno da studenti steknu praktična i funkcionalna znanja iz oblasti u vezi sa računarskim mrežama i sistemima, postoje brojni načini da bi se to postigne. Jedan od načina je svakako upotreba stvarnih mrežnih laboratorija. Te laboratorije se sastoje od stvarne mrežne opreme i studentima pružaju praktičan rad u realnom okruženju. Problem sa stvarnim mrežnim laboratorijama jeste njihova cena. Pored cene, problem je i njihova dostupnost. Stvarne laboratorije imaju limitiran pristup u pogledu istovremnog rada određenog broja studenata u njima. Da bi se svi ovi problemi prevazišli koriste se razni drugi načini u izvođenju nastave.

Jedan pristup jeste upotreba softvera za simulaciju rada i učenje računarskih mreža [FN10, Sar06]. Postoji veliki broj softvera te vrste. Jedan od njih je OPNet ITGuru. To je mrežni simulator diskretnih događaja (eng. *discrete-event network simulator*) i on će biti opisan detaljnije u nastavku. Pored ovog softvera, postoje i drugi slični simulacioni softveri kao što su ns-2 (Network Simulator), ns-3, Bosom NetSim [Bos11], NetSim i dr.

Ns-2 (Network Simulator) je okruženje za simulaciju TCP/IP mreža [Com00, Kom00], rutiranja i multikast protokola preko žičnih i bežičnih mreža. Ns-2 je nastao kao varijanta REAL network simulatora 1989. god. i razvio se u poslednjih nekoliko godina. 1995. god. njegov razvoj je podržan od strane agencije DARPA kroz VINT projekat [Ns11, Vin11]. Ns-3 [WJ09, PMS10, WLW09] je simulator diskretnih događaja [Wai09] koji se koristi za istraživanja i edukaciju u oblasti računarskih mreža. To je besplatan softver pod GNU GPLv2 licencom. Koristi se za nastavu na sledećim univerzitetima: Institut za tehnologiju Džordžija (eng. *Georgia Institute of Technology*), Univerzitet Kanzasa (eng. *University of Kansas*), Univerzitet Pensilvanije (eng. *University of Pennsylvania*), Univerzitet Brigam Jang (eng. *Brigham Young University*), Univerzitet Alto (eng. *Aalto University*) i Indijski institut za tehnologiju Bombaj (eng. *Indian Institute of Technology Bombay*). Oba softvera koriste se za istraživanje i edukaciju.

NetSim je [Nes11] popularni alat za simulaciju računarskih mreža. Koristi se za učenje računarskih mreža, za laboratorijske vežbe, eksperimente i istraživanje. Obuhvata veliki broj tehnologija poput Ethernet-a, WLAN, TCP/IP mreža i sl. Takođe se koristi na velikom broju svetskih univerziteta.

OPNet ITGuru Academic Edition softver predstavlja najznačajni softver ove vrste za ovo istraživanje. Laboratorijske vežbe za OPNET ITGuru su poslužile kao osnova za modeliranje mrežnih scenarija za VNetLab – model virtuelne mrežne laboratorije. Zbog toga, OPNet ITGuru je opisan detaljnije u poglavlju 4.

Predmet istraživanja jeste upotreba tehnologije za virtuelizaciju u cilju kreiranja virtuelnih mrežnih laboratorija. Obe ove oblasti su detaljnije opisane u poglavlju 3, zajedno sa pregledom istraživanja koja su do sada obavljena u svetu. Softver za virtuelizaciju ima i širu primenu u oblasti kreiranja virtuelnih računarskih mreža. Primer za to je PlanetLab [Pla11, PAC02, BBC04,



PBF06]. PlanetLab se sastoji od grupe umreženih računara koji se koriste kao test poligon za oblast računarskog umrežavanja i distribuiranih sistema, kao i za istraživanje u tim oblastima. Osnovan je 2002. god. pod rukovodstvom Lari Petersona (Larry L. Peterson), a do jula 2011. god. ima u svom sastavu 1084 umreženih čvorova na 538 lokacija širom sveta [MPF06, SPF07]. Iako je primarna namena ovog sistema infrastruktura za testiranje softvera, sistem se koristi na raznim univerzitetima i za nastavu [WWG07]. Virtuelno okruženje računarske mreže je omogućeno softverom za virtuelizaciju.

Potrebe za ovakvom vrstom istraživanja su višestruke. Dosadašnji projekti, koji su posebno opisani u poglavlju 3., imali su za primarni cilj stvaranje virtuelnog okruženja za emulaciju računarskih mreža namenjenih za testiranje i eksperimentisanje. Upotreba ovih okruženja u procesu nastave bila je samo sporedan rezultat, s obzirom na to da nijedan od projekata nije imao za cilj da kreira okruženje koje će se koristiti u procesu nastave, već samo u pojedinim njenim segmentima, a koje će biti prilagođeno edukaciji inženjerskih profila. Prema tome, ovo istraživanje bi se moglo svrstati u red pionirskih istraživanja na polju implementacije tehnologije za virtuelizaciju, prvenstveno u cilju stvaranja virtuelne mrežne laboratorije koja bi se koristila u nastavi na visokoškolskim ustanovama. Kao takva, laboratorija bi mogla da se koristi i u vannastavnim procesima, na primer za samostalan rad naprednih studenata i njihovo usavršavanje. Takođe, ovo istraživanje obezbeđuje nastavak rada na prilagođavanju tehnologije za virtuelizaciju u domenu emulacije računarskih mreža i omogućava stvaranje virtuelne mrežne laboratorije sa udaljenim pristupom. Udaljeni pristup laboratoriji predstavlja fleksibilnije okruženje za rad i upotrebu u nastavi i prevazilazi ograničenja koja nameću standardne fizičke laboratorije.

U okviru istraživanja, kao njegov rezultat, potvrđuje se upotrebljivost ovakve tehnologije za kreiranje laboratorije u nastavi na visokoškolskim ustanovama. Rezultati istraživanja se mogu upotrebiti u sklopu sistema za učenje na daljinu, što otvara nove pravce u toj oblasti i podiže nivo nastavnog procesa na računarskim profilima na visokoškolskim ustanovama.

Sa druge strane, kreiranje mrežne laboratorije sa daljinskim pristupom omogućava edukaciju i svih zainteresovanih osoba, van nastavnog procesa, koje imaju potrebu da detaljnije upoznaju koncepte računarskih mreža i Internet tehnologija. Ovakva nastojanja su razumljiva, s obzirom na sve širu upotrebu mrežnih tehnologija i njihov sve veći razvoj, povećana je i njihova upotreba u svakodnevnom životu.

Pored virtuelnih laboratorija koje su u centru istraživanja, kreirani model virtuelne laboratorije je primenljiv i na stvarne mrežne laboratorije čime se omogućava fleksibilno, ekonomično i lako prilagodljivo rešenje za nastavu u svim predmetima za koje su potrebne računarske mreže, operativni i distribuirani sistemi i slične oblasti.

Jedan od razloga za ovo istraživanje jeste i popularizacija tehnologije za virtuelizaciju, koja nije zanemarljiva, s obzirom na nepostojanje takvih i sličnih istraživanja na ovim prostorima. Istraživanjem se ukazuje na pouzdanost same tehnologije, kao i na mogućnost njene primene u drugim oblastima, a ne samo u domenu obrazovanja, i posebno u poslovnoj sferi.

## 2.2. Cilj istraživanja

Cilj istraživanja jeste razvoj efikasnog modela za upotrebu tehnologija za virtuelizaciju u nastavi na visokoškolskim ustanovama. Da bi se ostvario ovaj cilj, laboratorijske vežbe za predmet računarske mreže prilagođene su ovakvom okruženju. Nakon toga je razvijen model virtuelne mrežne laboratorije za rad u lokalu i sa daljinskim pristupom, bazirano na TCP/IP skupu protokola.

## 2.3. Zadaci istraživanja

Zadaci istraživanja su:

- prikaz i analiza tehnologija za virtuelizaciju,
- prikaz i analiza virtuelnih mrežnih laboratorija baziranih na tehnologijama za virtuelizaciju,
- odabir najpogodnije tehnologije za razvoj virtuelne laboratorije,
- modeliranje i realizacija virtuelne laboratorije,
- modeliranje scenarija za laboratorijske vežbe,
- kreiranje programiranog materijala za laboratorijske vežbe,
- analiza kreiranog modela kroz upotrebu u nastavi,
- izvođenje zaključaka,
- prikaz retrospektive i perspektive istraživanja.

## 2.4. Hipoteze istraživanja

Glavna hipoteza disertacije glasi: *moguće je kreirati efikasnu virtuelnu mrežnu laboratoriju baziranu na tehnologiji za virtuelizaciju, kao okruženje za emulaciju računarskih mreža sa mogućnošću lokalnog i daljinskog pristupa, koja je pogodna za upotrebu u nastavi na visokoškolskim ustanovama.*

Pothipoteze su:

- moguće je kreirati virtuelnu mrežnu laboratoriju baziranu na tehnologiji za virtuelizaciju,
- kreirana mrežna laboratorija predstavlja emulaciju okruženja realnih računarskih mreža,
- nastava iz pojedinih predmeta na visokoškolskim ustanovama može se organizovati u okruženju virtuelne mrežne laboratorije,
- virtuelna mrežna laboratorija može se realizovati i kao deo sistema za učenje na daljinu.

## 2.5. Očekivani rezultati istraživanja

Osnovni očekivani rezultat ovog istraživanja i naučni doprinos u oblastima informacionih tehnologija i računarskih mreža i komunikacija, kao i njihove emulacije jeste metodologija razvoja modela virtuelne mrežne laboratorije bazirane na tehnologiji za virtuelizaciju sa lokalnim i pristupom na daljinu. Takva metodologija omogućava novi vid primene tehnologije za virtuelizaciju u ovom domenu, kao i oblik realizacije nastave.

Praktični rezultat istraživanja jeste prilagođeni sistem za upotrebu u nastavi inženjerskih profila u okviru predmeta računarske mreže, baziran na tehnologiji za virtuelizaciju koji može da se primenjuje u nastavi na visokoškolskim ustanovama. Takav sistem može da se koristi kao sredstvo za popularizaciju osnovnih koncepata računarskih mreža i Interneta, sa mogućnošću primene u drugim predmetima kao što su Operativni sistemi [DS06], Distribuirani sistemi itd.

Očekivani rezultati istraživanja su i:

- model virtuelne mrežne laboratorije sa lokalnim i daljinskim pristupom koji je primenljiv i na realne mrežne laboratorije, tj. model za udaljeni pristup, rad i kontrolu u realnoj mrežnoj laboratoriji,
- metodologija modeliranja sistema i okruženja virtualne mrežne laboratorije,
- metodologija modeliranja laboratorijskih vežbi za nastavu na predmetu Računarske mreže za virtuelnu mrežnu laboratoriju i potpuna ili delimična pokrivenost vežbi iz tog predmeta,
- metod vrednovanja i verifikacije virtuelnih mrežnih laboratorija i njihove upotrebljivosti u nastavi zasnovan na teoriji grubih skupova,
- promovisanje tehnologije za virtuelizaciju i koncepta virtuelnih mrežnih laboratorija na ovim prostorima, tj. stvaranje mogućnosti da se ta tehnologija u većoj meri primenjuje na ovim prostorima,
- mogućnost primene tehnologije za virtuelizaciju u drugim domenima, pre svega u poslovnom domenu.

## 2.6. Metode i tehnike istraživanja

U istraživanju se koriste metode teorijske analize, neophodne za proučavanje dosadašnjih istraživanja iz navedenih oblasti, kao i najnovijih rešenja u oblastima interesa. Za analizu i interpretaciju podataka korišćene su metode analize i sinteze, kao i metode poređenja.

Za modeliranje sistema, njegovo prilagođavanje okruženju za pristup na daljinu, kao i za kreiranje mrežnih scenarija i njima odgovarajućih laboratorijskih vežbi primenjena je deskriptivna metoda.

Istraživački rad se odnosi na prikupljanje podataka o korišćenju i performansama laboratorije, mišljenjima studenata i prevashodno njihovom uspehu u radu, kako bi se odredila efikasnost

razvijenog modela. Takođe, definišu se parametri vrednovanja modela. Primenjene metode i instrumenti u ovom slučaju su:

- Anketiranje,
- Definisane kriterijuma o nivou usvojenog znanja na osnovu kontrolnih testova,
- Formativna procena nivoa usvojenih znanja studenata,
- Analiza postignutog uspeha studenata na osnovu metoda zasnovanog na teoriji grubih skupova

## **2.7. Uzorak**

Uzorak istraživanja su studenti informacionih tehnologija na Tehničkom fakultetu „Mihajlo Pupin” – Zrenjanin (II i III godina). Da bi se postigla maksimalna efikasnost, istraživanje je obavljeno u dužem vremenskom periodu.

U prvoj fazi u kojoj se vršilo istraživanje, predmet Računarske mreže se držao na III godini studija sledećih profila: Inženjer informatike, Inženjer poslovne informatike i Profesor informatike. U drugoj fazi istraživanja, koja se vremenski poklapa sa prelaskom Tehničkog fakulteta „Mihajlo Pupin“ na nove, akreditovane programe, predmet Računarske mreže drži se na II godini obrazovnog profila Informacione tehnologije. U skladu sa vremenom istraživanja, ciljne grupe istraživanja odgovaraju studentima aktuelnih obrazovnih profila, tj. smerova.

Mesto eksperimentalnog istraživanja bio je Tehnički fakultet “Mihajlo Pupin” – Zrenjanin. Istraživanje je izvedeno na postojećoj mrežnoj infrastrukturi i postojećim računarima, kao i u virtuelnoj mrežnoj laboratoriji VNLab – koja je rezultat ovog istraživanja. U toku istraživanja koristile su se razne PC platforme sa Microsoft Windows operativnim sistemima, budući da je to preovlađujuća platforma na mestu istraživanja.

Broj studenata obuhvaćen istraživanjem je prikazan u poglavlju 6., a on iznosi:

- 55 studenata u analizi primenom metoda baziranog na teoriji grubih skupova,
- 48 studenata u prvom kontrolnom testu,
- 126 studenata u drugom kontrolnom testu,
- 49, 68 i 72 studenta u toku tri godine primene formativne procene.

## **2.8. Postupci i instrumenti za vrednovanje modela**

Model je vrednovan na osnovu:

- analize rada virtuelne mrežne laboratorije VNLab utvrđivanjem stepena sličnosti sa realnim okruženjem,
- formativne procene na osnovu koje se koriguje upotreba laboratorije i omogućava povećanje njene efikasnosti,

- metoda zasnovanog na teoriji grubih skupova primenjene za utvrđivanje nedostataka u korišćenim laboratorijskim vežbama,
- analize uspeha studenata i nivoa usvojenog znanja na osnovu kontrolnih testova,
- upotrebljivost laboratorije je ispitana poređenjem sa preporučenim nastavnim programima od strane zvaničnih svetskih organizacija (ACM, IEEE i AIS) za obrazovne profile i zvanja iz oblasti računarstva: Računarske nauke, Računarsko inženjerstvo, Informacione tehnologije i Informacioni sistemi.

### 3. Dosadašnja istraživanja

Dosadašnja istraživanja koja se odnose na virtuelne mrežne laboratorije mogu se svrstati u tri oblasti. Prva oblast od interesa je tehnologija za virtuelizaciju [CN01, McE02, AA06, Gor06], jer se ona koristi kao osnova za kreiranje virtuelnih mrežnih laboratorija. Druga oblast su same virtuelne mrežne laboratorije. Na kraju, treća oblast od interesa, bavi se softverskim alatima koji omogućavaju osnovnu funkcionalnost ove laboratorije. U skladu sa nevedenim formirana je i struktura ovog poglavlja.

S obzirom na važnost tehnologije za virtuelizaciju za ovo istraživanje, u nastavku je dat njen kratak istorijat i razvoj, prikazana je njena arhitektura i najbitnije karakteristike. Zajedno sa arhitekturom, prikazani su i karakteristični softverski paketi za virtualizaciju. S obzirom da se radi o rasprostranjenosti oblasti, koja obuhvata veliki broj različitih tehnoloških rešenja, navedeni su samo karakteristični primeri. Tako je omogućen uvid u trenutnu ponudu na tržištu i na sve relevantne aspekte tehnologije.

U sledećoj sekciji prikazana su postojeća rešenja za virtuelne mrežne laboratorije koja se koriste u svetu. Prezentovane su njihove najbitnije karakteristike, način upotrebe i tehnologija na kojoj su zasnovane.

Na kraju, prikazane su softverske komponente bitne za funkcionisanje virtuelnih mrežnih laboratorija. Te softverske komponente omogućavaju osnovnu funkcionalnost virtuelne mreže [Lur05a, Lur05b]. Mada u tu kategoriju spada celokupan serverski i klijentski softver koji može da se upotrebi za učenje različitih mrežnih koncepata, u ovom delu fokus je usmeren isključivo na softver za rutiranje. Budući da su ruteri (usmerivač ili mrežna skretnica, *eng. router*) jedan od bitnih elemenata virtuelnih mreža, softver koji omogućava usmeravanje paketa u TCP/IP mrežama, predstavlja svakako najvažniju softversku komponentu ovih okruženja.

#### 3.1. Tehnologija za virtuelizaciju

U osnovi virtuelnih mrežnih laboratorija nalazi se tehnologija za virtuelizaciju [SC05, KDC03, McE02, RR09, WH07]. Uprkos velikoj popularnosti ove tehnologije u današnje vreme, radi se o veoma staroj tehnologiji. Nastanak ove tehnologije vezuje se sa ranim 70-im godinama prošlog veka [Goldberg 1973, Gol74]. Naime, 1972. god. IBM je pustio na tržište svoju prvu komercijalnu verziju virtuelizacije celokupnog operativnog sistema. Prva komercijalna verzija, nazvana VM/370, razvijena je za njihov sistem IBM System/370. Pre toga, postojale su starije verzije nekomercijalnih sistema za virtuelizaciju koje su se koristile u IBM laboratorijama kasnih 60-ih godina, ali je VM/370 implementacija poznata kao puna virtuelizacija i omogućavala je rad svih softvera na određenom hardveru [Cre81].

Razlozi koji su doveli do nastanka, a kasnije razvoja i popularnosti ove tehnologije bili su bolja iskorišćenost postojećih računarskih resursa. Računarski sistemi toga perioda su bili veoma skupi, te su njihova racionalna upotreba i veći stepen iskorišćenja bili od vitalnog značaja. Virtualizacija je omogućila da se snaga računarskih sistema iskoristi u što većoj meri. U narednom periodu, sa padom cena hardvera i razvojem personalnih računara dolazi i do pada popularnosti tehnologije za virtuelizaciju.

Njen preporod počinje 1988. god. U to vreme, manje poznata kompanija, Connectix razvija softverski proizvod pod nazivom Virtual PC [CBE10, LRS03]. Zahvaljujući njegovom uspehu, kompanija Microsoft kupuje Connetix 2003. god. Nakon toga dolazi do razvoja novih proizvoda. Prvo je to bio MS Virtual Server 2005, koji je razvijen za pokretanje serverskih operativnih sistema, a zatim i Type-1 sistem, Hyper-V.

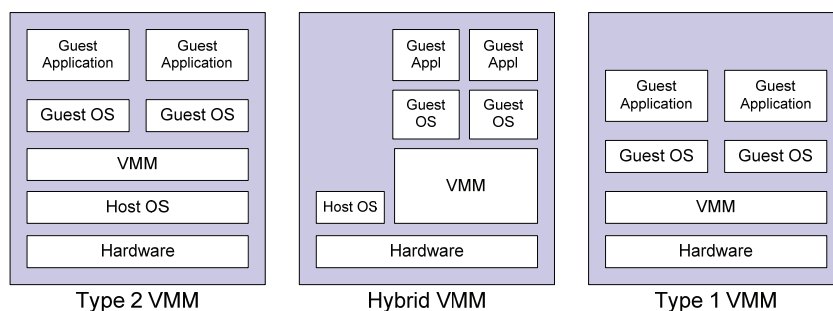
Kompanija Microsoft nije jedina na ovom tržištu. Kompanija VMware je osnovana 1998. god., kada se i počelo sa razvojem njihovih desktop orijentisanih proizvoda za virtuelizaciju. Ostale konkurentske kompanije su Citrix, sa Xen hypervisor paketom i razne open-source implementacije. Različiti proizvodi za virtuelizaciju će biti detaljnije opisani u nastavku teksta, zajedno sa osnovnim karakteristikama tehnologije. Međutim, i pored velike konkurencije, Microsoft i VMware pokrivaju trenutno preko 70% tržišta [CBE10].

Što se tiče same tehnologije, ona se može klasifikovati u tri tipa, ako se klasifikacija vrši prema njenoj strukturi (Slika 3.1.) ili platformi na kojoj je izgrađena. Ta tri tipa su Type I, Type II i Hybrid VMM (monitori za virtuelne mašine ili *eng. virtual machine monitors*).

Type II VMM, nazivaju se *hosted* i u potpunosti su izgrađeni na host operativnom sistemu, tj. operativnom sistemu koji je instaliran na računaru. Ovaj tip je fokusiran na virtuelizaciju procesa, a ne na virtuelizaciju servera. Type II arhitektura je implementirana u slučaju Java Virtual Machines i Microsoft Common Language Runtime okruženja, tj. u slučaju kada je potrebno obezbediti *runtime* okruženje. Ovo okruženje pruža mogućnost pokretanja komandi ili procesa nezavisno od operativnog sistema [CBE10].

Hybrid i Type I arhitektura su bazirane na virtuelizaciji hardvera. Hybrid arhitektura (naziva se još *full*) pokreće VMM ili softver za upravljanje virtualnim mašinama zajedno sa operativnim sistemom. Oni rade uglavnom direktno na hardveru, ali koriste host operativni sistem (host OS) za I/O operacije. VMM se instalira kao aplikacija na operativnom sistemu i emulira celokupno fizičko okruženje, tj. hardver računara na kojem se nalazi. Tako se omogućava virtuelizacija celokupnog okruženja operativnog sistema na virtuelnoj mašini. Primeri za ovaj tip arhitekture su IBM's VM/370, VMware Server i VMWare Workstation [N000, SWL01], Microsoft Virtual PC i Microsoft Virtual Server [LC07].

Type I VMM (*eng. stand-alone*), su implementirani direktno na hardveru računarskog sistema (*eng. bare metal*) i predstavljaju najefikasniji način za virtuelizaciju. Drugim rečima, oni izbegavaju posredovanje host operativnog sistema što im i omogućuje najbolje performanse. Primeri softverskih rešenja za ovaj tip su Hyper-V, Citrix XenServer i VMware ESX Server [Tul10].



Slika 3.1. Tip virtuelnih mašina

Druga klasifikacija, bazirana na performansama tehnologija za virtuelizaciju, se može izvršiti po sledećim kategorijama [HN10]:

- gostujući OS / host OS (eng. *guest OS/host OS*)
- hipervizor (eng. *Hypervisor*)
- emulacija
- virtuelizacija na nivou kernela (eng. *kernel-level virtualization*)
- virtuelizacija deljenog kernela (eng. *shared kernel virtualization*)

### ***Gostujući OS/Host OS***

Kategorija virtuelnih mašina koje se mogu nazvati klasičnom ili *hosted* virtualizacijom čine ovu grupu. Host operativni sistem je sistem koji je instaliran na računaru i koji je u potpunosti operativan. Softver za virtuelizaciju pruža mogućnost kreiranja više virtuelnih mašina (VM), koje predstavljaju virtuelne računare. Svaka od tih virtuelnih mašina može da instalira poseban operativni sistem – gostujući ili guest OS. Tipični predstavnici za ovu kategoriju su VMware Server i Oracle VirtualBox (ranije Sun xVM) [Ora11]. Oba ova proizvoda su razvijena za rad u manjim okruženjima, testiranje i za ličnu upotrebu.

### ***Hipervizor***

Ovaj tip softvera za virtuelizaciju instaliran je direktno na hardver računarskog sistema. Operativni sistem se instalira na hypervisor, što predstavlja jednu virtuelnu mašinu (VM). Predstavnici ovakvog tipa softvera su Citrix Xen [BDF03, WG07], koji je započeo svoj razvoj kao istraživački projekat Grupe za istraživanje sistema u okviru Računarske laboratorije na Univerzitetu K embriđ (eng. *Systems Research Group, University of Cambridge Computer Laboratory*), VMware ESX [Wal02], koji pokreće visoko specijalizovan Linux kernel - vmkernel [MW05], njegova novija verzija VMware ESXi i Microsoft Hyper-V [KVV09]. Svi ovi proizvodi su namenjeni virtuelizaciji visokog nivoa i zahtevnim sistemima u poslovnom okruženju.

### ***Emulacija***

Ovaj tip softvera za virtuelizaciju pruža mogućnost emulacije određenog tipa hardvera za instalirani *guest OS*, koji nije prisutan na fizičkom računaru, bez obzira na host operativni sistem. To omogućava da se instalira Sparc verzija operativnog sistema Solaris na ne-Sparc host računarima. Najbolji primer za ovu vrstu softvera su Bochs, QEMU, Microsoft Virtual PC i Virtual Server. Bosch i QEMU su besplatni, open-source programi, dok Virtual PC i Virtual Server 2005 predstavljaju virtualizacioni softver razvijen od strane Microsoft-a. Postoje u 32-bitnim i 64-bitnim verzijama Windows operativnog sistema, ali sa podrškom samo za 32-bitne *guest* operativne sisteme [CBE10].

### ***Virtuelizacija na nivou kernela***

Virtuelizacija na nivou kernela znači da svaka virtuelna mašina koristi svoj sopstveni kernel za podizanje sistema bez obzira na kernel host računara. Predstavnici ove tehnologije su KVM i



User-Mode Linux. Kernel-based Virtual Machine (KVM) podržava Windows/Linux/Unix guest operativne sistema u 32-bitnom i 64-bitnom okruženju i uključen je u kernel od verzije 2.6.20 od 2007. god. [Sha08]. User-Mode Linux (UML) koristi izvršni kernel i UML-kreirani root sistem datoteka (eng. *filesystem*) da bi se kreirala virtuelna mašina [Dik01, Dik00, Dik02, HBS02]. Kao i KVM, UML je uključen je od verzije 2.6.x kernela. UML predstavlja veoma popularnu platformu za kreiranje virtuelnih okruženja za učenje i testiranje [GFR04, GF04, Rim07, SMR05, Al-04, Luruo2005, LMZ01, KSR05].

### ***Virtuelizacija deljenog kernela***

Virtuelizacija deljenog kernela koristi sposobnost UNIX i Linux operativnih sistema da dele kernel sa ostalim procesima. Primeri su Solaris Containers (Zones) i OpenVZ.

### **3.2. Virtuelne laboratorije zasnovane na tehnologiji za virtuelizaciju**

Softver za virtuelizaciju pokazao se kao dovoljno efikasan za kreiranje virtuelnih laboratorija [Kuz05, HV04, Lur05a, Lur05b]. Te laboratorije imaju višestruku primenu. Njihova osnovna primena jeste stvaranje okruženja za eksperimentisanje u oblasti novih tehnologija, testiranje i razvoj mrežnog softvera. Njihova druga, ali ne manje značajna namena je upotreba u obuci i obrazovanju IT eksperata, tj. u oblasti inženjerske edukacije (eng. *engineering education*). Niz primera može da potvrdi ovu tvrdnju. Primena ovih laboratorija se može ostvariti u oblasti operativnih sistema [DS06], sistem administracije i sigurnosti mreža [Bor07, WHY10, SHA09, Yan07], razvoj serverskog i klijentskog softvera, bazama podataka [BBS06] i za distribuirane i mrežne platforme kao i u ostalim oblastima [AJD09]. Takođe, postoji primena ovih laboratorija u oblasti edukacije iz računarskih mreža i ta primena ima najveću popularnost. To se može zaključiti i po broju različitih platformi koje su razvijene posebno za tu svrhu ili su primarno razvijene za eksperimentisanje i razvoj ali su ipak prilagođene za učenje koncepata računarskih mreža [Net06, Net11b].

Upotrebom ove tehnologije i odgovarajućeg pratećeg softvera mogu se kreirati virtuelna mrežna okruženja koja u potpunosti odgovaraju stvarnim sistemima. Virtuelizacija omogućava kreiranje većeg broja virtuelnih računara sa različitim operativnim sistemima na jednom fizičkom računaru. Ti virtuelni računari mogu imati ulogu servera, radne stanice, a po potrebi i mrežnih uređaja kao što je ruter. Uloga rutera se u potpunosti može emulirati računalom, tj. virtuelnom mašinom uz upotrebu odgovarajućeg softvera. Svaki virtuelni računar, tj. virtuelna mašina (VM), koristi postojeći hardver fizičke mašine na kojoj se nalazi. Budući da se sve hardverske komponente host računara emuliraju, emulira se i njegova mrežna kartica (eng. *network interface card* - NIC). Svaka virtuelna mašina može da emulira više mrežnih kartica na osnovu jedne fizičke. Između virtuelnih mašina može se uspostaviti virtuelna mrežna konekcija tako da se virtuelne mrežne kartice povezuju u virtuelnu mrežu ili mrežni segment.

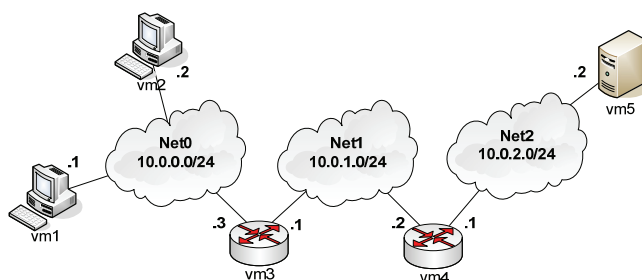
U nastavku sekcije biće opisana najkarakterističnija okruženja virtuelnih mrežnih laboratorija, zajedno sa njihovim osnovnim karakteristikama.

### ***VNUML (Virtual Network User Mode Linux)***

VNUML je open-source alat opšte namene baziran na mrežnim scenarijima. Ovaj alat je prvobitno razvijen zbog simulacije rada računarskih mreža i kao sredstvo za kreiranje test poligona za virtuelne mreže. Baziran je na User-mode Linux-u (UML) tehnologiji. Razvijen je 2004. god. na Odseku Telematics Engineering Department - DIT na Tehničkom univerzitetu Madrid (*Technical University of Madrid - UPM*). VNUML je razvijen kao deo Euro6IX

istraživačkog projekta za eksperimentisanje u oblasti IPv6 mrežnih sistema. Kao rutin softver u okruženju se koristi Zebra/Quagga open-source softver za rutiranje [GF04, FGM04, GFR04, FPG05, GF04, GF07, GVF08, GFF09, Arn05].

VNUML se sastoji od dve komponente: XML-baziranog jezika koji omogućava opis mrežnog scenarija. Taj jezik se naziva VNUML jezik sa specifikacijom. Druga komponenta sistema je interpreter jezika koji kreira i pokreće scenarija - VNUML parser. Ova komponenta omogućava definisanje i pokretanje mrežnog scenarija sa GNU/Linux virtuelnim mašinama. Sve virtuelne mašine su međusobno povezane u okviru virtuelnih mrežnih segmenata na jednom fizičkom računaru – serveru virtuelne mreže. VNUML se koristi i na univerzitetu Koblenz-Landau (Nemačka) na Institutu za računarske nauke (eng. *Institute for Computer Science*) u okviru Radne grupe Steigner (eng. *Working Group Steigner*) kao okruženje za eksperimente sa računarskim mrežama.



Slika 3.2. Primer VNUML scenarija

Primer VNUML scenarija prikazan je na slici 3.2., a datoteka koji definiše taj scenario napisan je VNUML XML baziranim jezikom i prikazan je na listingu 3.1.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE vnuml SYSTEM "/usr/share/xml/vnuml/vnuml.dtd"?>
<vnuml>
  <global>
    <version>1.8</version>
    <vm_defaults>
      <filesystem>/usr/share/vnuml/fs/root_fs</filesystem>
      <kernel>/usr/share/vnuml/kernels/linux</kernel>
    </vm_defaults>
  </global>
  <net name="Net0" />
  <net name="Net1" />
  <net name="Net2" />
  <vm name="vm1">
    <if id="1" net="Net0">
      <ipv4>10.0.0.1/24</ipv4>
    </if>
    <route type="ipv4" gw="10.0.0.3">default</route>
  </vm>
  <vm name="vm2">
    <if id="1" net="Net0">
      <ipv4>10.0.0.2/24</ipv4>
    </if>
    <route type="ipv4" gw="10.0.0.3">default</route>
  </vm>
  <vm name="vm3">
    <if id="1" net="Net0">
      <ipv4>10.0.0.3/24</ipv4>
    </if>
    <if id="2" net="Net1">
      <ipv4>10.0.1.1/24</ipv4>
    </if>
  </vm>
  <vm name="vm4">
    <if id="1" net="Net1">
      <ipv4>10.0.1.2/24</ipv4>
    </if>
    <if id="2" net="Net2">
      <ipv4>10.0.2.1/24</ipv4>
    </if>
  </vm>
  <vm name="vm5">
    <if id="1" net="Net2">
      <ipv4>10.0.2.2/24</ipv4>
    </if>
  </vm>
</vnuml>
```

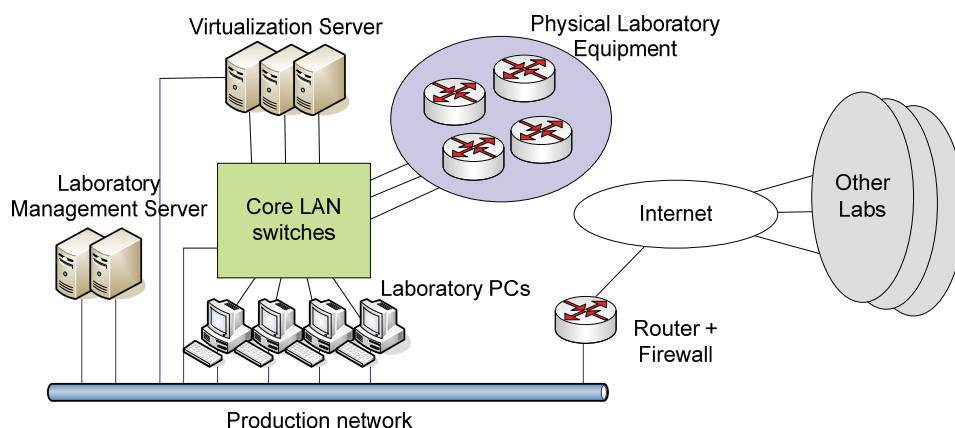
```

</if>
<route type="ipv4" gw="10.0.1.2">10.0.2.0</route>
<forwarding type="ip">
</vm>
<vm name="vm4">
  <if id="1" net="Net1">
    <ipv4>10.0.1.2/24</ipv4>
  </if>
  <if id="2" net="Net2">
    <ipv4>10.0.2.1/24</ipv4>
  </if>
<route type="ipv4" gw="10.0.1.1">default</route>
<forwarding type="ip">
</vm>
<vm name="vm5">
  <if id="1" net="Net2">
    <ipv4>10.0.2.2/24</ipv4>
  </if>
<route type="ipv4" gw="10.0.2.1">default</route>
<filetree seq="webup@" root="/var/www">/opt/vnuml/>vm5</filetree>
<exec seq="webup" type="verbatim">/etc/init.d/apache start</exec>
<exec seq="webdown" type="verbatim">/etc/init.d/apache stop</exec>
<exec seq="webdown" type="verbatim">rm /var/www/index.html</exec>
</vm>
</vnuml>

```

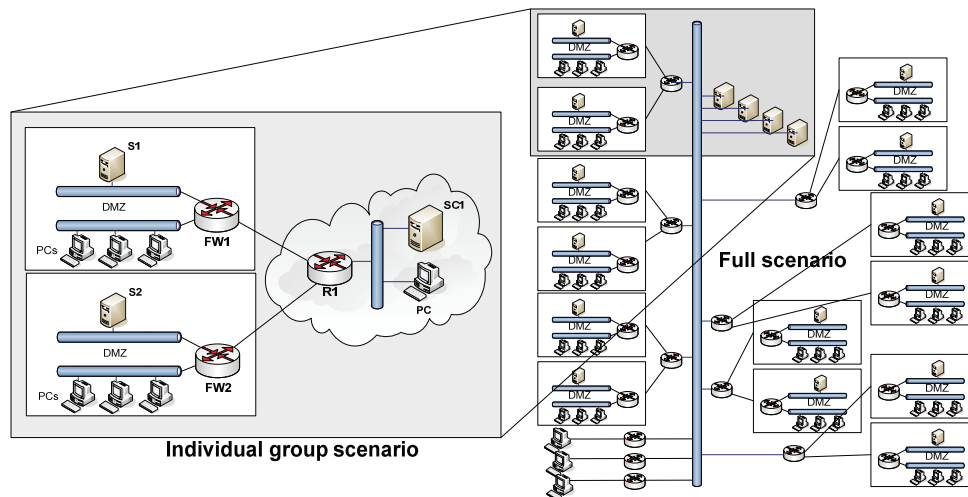
Listing 3.1. VNUML konfiguracija mrežnog scenarija prikazanog na slici

VNUML se kao okruženje, u toku svog razvoja, transformisao od laboratorije sa lokalnim pristupom (dostupne i kao *Live CD*) u okruženje u kome je omogućen daljinski pristup. Slika 3.3. pokazuje generalnu arhitekturu mrežne laboratorije sa daljinskim pristupom. Jezgro laboratorije se sastoji od nekoliko LAN komutatora, tj. svičeva (*eng. switch*) sa VLAN podrškom. Sva ostala oprema (ruteri, serveri za virtualizaciju i studentske radne stanice) povezana je na osnovne LAN svičeve, koji su na slici prikazani kao *Core LAN Switches*, koji ujedno predstavljaju infrastrukturno jezgro laboratorije.



Slika 3.3. Arhitektura VNUML laboratorije

Na slici 3.4. prikazan je jedan scenario na kome je simultano radilo 28 studenata, podeljeno u 14 grupa. Šematski je prikazan kompletan „simultani scenario“, kao i deo virtuelne mreže izdvojen za jednu od studenskih grupa. Tema scenarija je konfiguracija fajervol sistema, a za njegovo funkcionisanje bilo je potrebno 66 virtuelnih računara. Oni su podeljeni u osam podscenarija i raspodeljeni na dva Dell PE840 servera (Dual core Xeon 3060 sa 2 GB RAM-a).



Slika 3.4. Prikaz scenarija u VNUML okruženju sa daljinskim pristupom

Glavno ograničenje ovakvog sistema, a ujedno i njegova najveća mana, je nemogućnost UML softvera za virtuelizaciju da pokrene neke druge operativne sisteme osim GNU/Linux OS. Tako da u ovim okruženjima ne postoji podrška za Windows i BSD operativne sisteme.

### Netkit

Netkit [Net06] predstavlja drugi najznačajniji sistem za emulaciju mreža, baziran na istoj tehnologiji za virtuelizaciju. Sistem je u potpunosti razvijen na *open-source* softveru. Sastoji se od četiri komponente: kernela, imidž datoteka sistema, softvera za virtuelni hub i skupa definisanih korisničkih komandi. Okruženje je razvijeno za eksperimentalnu upotrebu u domenu računarskih mreža. Projekat je započet 2005. god. u okviru Grupe za istraživanja u računarskim mrežama (eng. *Computer Networks Research Group*) na Univerzitetu Roma Tre (*University of Roma Tre*) kao deo Linux User Group LUG Roma 3 projekta sa namerom da se kreira jeftino okruženje za edukaciju. Projekat je proširen XML – baziranim jezikom koji je nazvan NetML. NetML je dizajniran sa ciljem da opiše mrežne topologije koje se koriste u mrežnom scenariju tog okruženja [Rim07, PR08].

Mrežni čvorovi u okviru Netkit sistema emuliraju se upotrebom User-mode Linux-a koji se koristi zajedno sa Debian GNU/Linux operativnim sistemom. Bilo koja virtuelna mašina može se pretvoriti u određeni mrežni uređaj (npr. ruter) pokretanjem određenog softvera za tu svrhu. U ovom okruženju se koriste Quagga [Qua11] ili XORP [HHK03, Xor11].

Vežbe koje su obuhvaćene ovom laboratorijom grupisane su na sledeći način. Prva grupa vežbi predstavlja osnovne teme, a to su: rad na konfiguraciji jednog i dva hosta, statičko rutiranje, ARP i RIP protokoli. Druga grupa vežbi se zove aplikacioni nivo (eng. *Application level*) i obuhvata rad sa DNS i e-mail protokolima (SMTP, POP, IMAP). Napredne teme obuhvataju: *bridging*, STP protokol i MPLS. Na kraju, grupa vežbi obuhvata teme rutiranja između domena (eng. *Interdomain Routing*). Postoji 10 vežbi: *Simple peering*, *simple announcement*, *prefix filtering*, *Stub AS*, *Stub AS (static routes)*, *Multi homed stub AS*, *Large multi homed stub AS*, *Multi homed AS*, *Small Internet*, *Transit AS*. Cela grupa vežbi se odnosi na rad sa autonomnim sistemima i BGP protokolom, u osnovnim i naprednim konfiguracijama [Net11b].

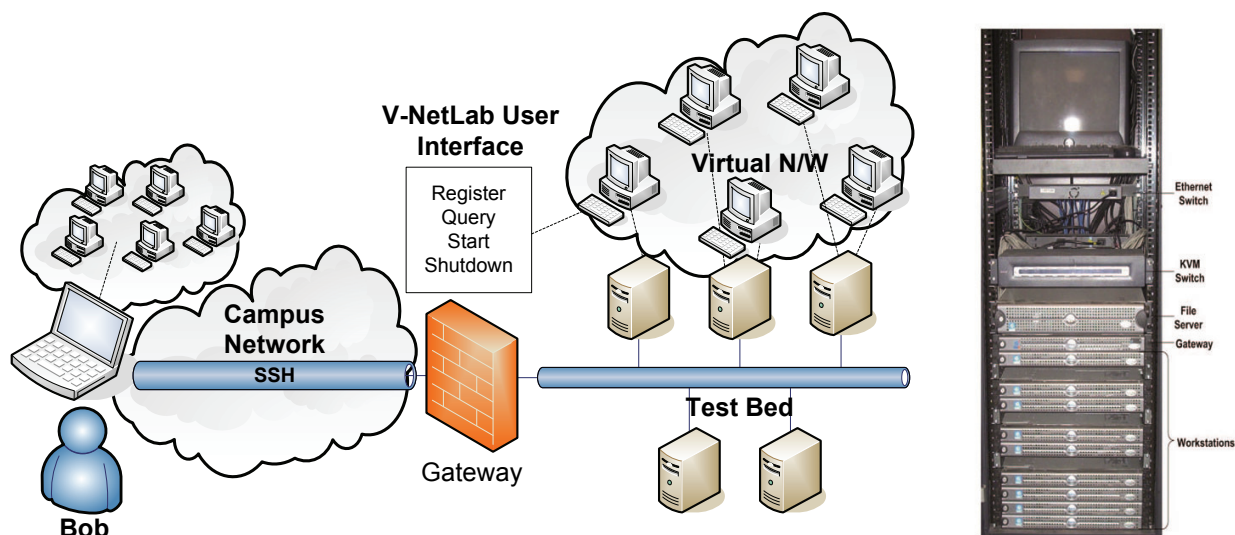
Netkit se koristi na velikom broju univerziteta kao sredstvo za nastavu: Roma Tre univerzitet, Univerzitet Kalabrija, Univerzitet Piza, Univerzitet kralj Huan Karlos, Univerzitet San Pablo CEU, Moskovski državni tehnički univerzitet, INSA Ruen, Ukrajinska računarska akademija SHAG, De

Montfort univerzitet, Leicester, La Sapiencia univerzitet, Univerzitet Sao Paulo, Univerzitet Limož [Net11a].

Sistem je namenjen za rad sa lokalnim pristupom. Pored instalacionog paketa, postoji i *Live CD* verzija. Najveći nedostatak sistema je, kao i kod VNUML-a, nemogućnost UML tehnologije da podrži drugi operativni sistem osim Linux-a.

### V-NetLab

Ovo okruženje predstavlja kompleksnu strukturu koja se sastoji od jednog NFS servera. Server se koristi za smeštaj virtuelnih mašina i njihovih disk imidž (eng. *image*) datoteka. Druga komponenta sistema jeste devet radnih stanica sa Linux operativnim sistemom i instaliranim VMware softverom za virtuelizaciju. Treća komponenta sistema jeste gejtvej ili mrežni prolaz (eng. *gateway*), koji omogućuje korisnicima pristup virtuelnom mreži i interfejsu za upravljanje. Infrastruktura sistema prikazana je na slici 3.5., a njegoa arhitektura prikazana je na slici 3.6. [KSR05, SKK08].

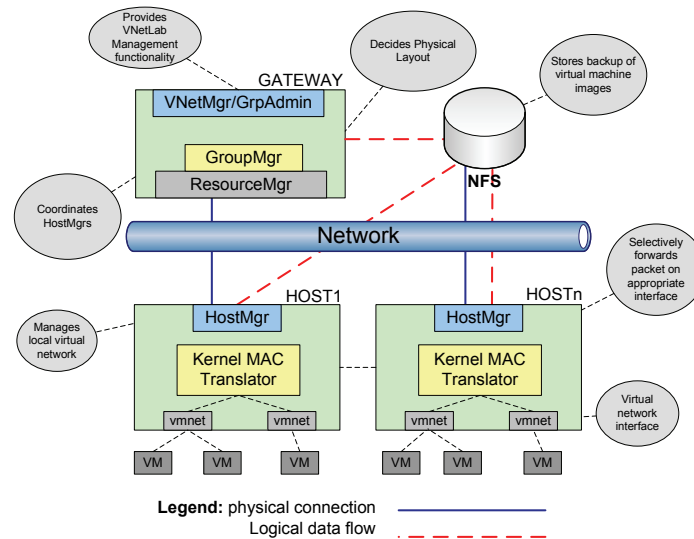


Slika 3.5. Prikaz infrastrukture V-NetLab sistema

Sistem se sastoji od jednog NFS servera. Server omogućava smeštaj za imidž datoteke diskova virtuelnih mašina. Server koristi RAID sistem i ima kapacitet od 600GB, što je dovoljno za 200 do 500 virtuelnih mašina. Pored servera u sistemu se nalazi 9 radnih stanica. Svaka ima 3 do 4 GB RAM-a i 2.8 ili 3.0 GHz Intel dual-procesore pod Linux operativnim sistemom. Kao softver za virtuelizaciju koristi se VMware Inc. [Vmw11]. U sistemu se 6 radnih stanica koristi u tzv. „modu za kurseve“, tj. na njima se izvode kursevi, a 3 radne stanice su u „modu za razvoj“ i koriste se za razvoj i testiranje V-NetLab softvera. Gejtvej host se koristi za logovanje korisnika na sistem sa spoljne mreže. Taj host ima na sebi upravljački sistem koji omogućava studentima i drugim korisnicima pristup V-NetLab-u i izvođenje vežbi. U realnom radu ova platforma koristi od 130 do 300 virtuelnih mašina istovremeno. To omogućava istovremeni pristup od 30 do 60 studenata. Svaki od tih studenata ima svoju posebnu virtuelnu mrežu sa kojom radi. Ceo hardversko-softverski sistem koštao je oko \$29,000 u vreme implementacije. Ovaj projekat je potpomognut od strane NSF grant DUE-0313858 i ONR grant N000140110967.

Vežbe koje se izvode u ovakvom okruženju su:

- Konfiguracija fajervol sistema uz upotrebom *iptables* softvera [Pur04],
- analiza rada mreže i saobraćaja uz pomoć alata kao što su *ping*, *traceroute* i *nmap* [KD00]
- i otkrivanje napada na mrežu (eng. *Network Intrusion Detection*) uz pomoć alata *snort*.



Slika 3.6. Prikaz arhitekture V-NetLab sistema

## Einar

Za razliku od ostalih projekata, u ovom slučaju se radi o studentskom projektu koji je imao za cilj kreiranje virtuelne mrežne laboratorije, namenjene prvenstveno za rad sa dinamičkim rutiranjem i učenje tog koncepta. Projekat više nije aktivan (kao ni Internet stranica sa koga se može preuzeti), a distribuiran je u *Live CD* verziji. Razvijen je na Kraljevskom institutu za tehnologiju (eng. *Royal institute of technology* - Švedska) i zasnovan je Knoppix platformi sa Xen softverom za virtuelizaciju i Quagga routing softverom [Ein06].

## VELNET (*Virtual Environment for Learning Networking*)

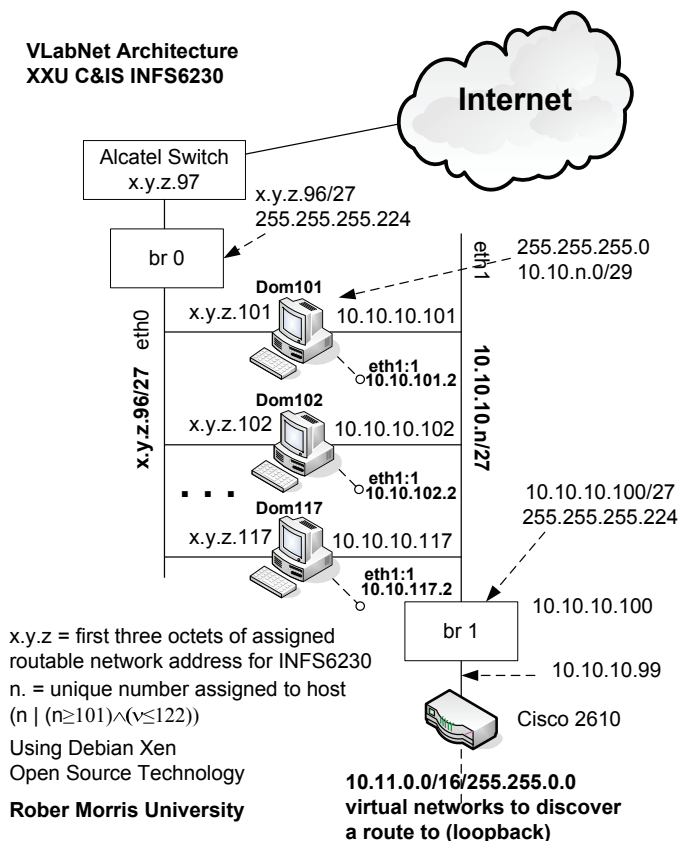
VELNET predstavlja još jedno rešenje namenjeno upotrebi u nastavi računarskih mreža [KB03, KHB04]. Za razliku od većine rešenja koja su razvijena na UML ili Xen platformi, VELNET koristi VMWare Workstation softver za virtuelizaciju sa različitim guest operativnim sistemima (Windows XP, Windows NT Workstation, OpenBSD). U ovom slučaju prevladuje Windows platforma, za razliku od prethodnih rešenja gde dominira Linux okruženje. Razvijen je 2003. god. na Školi za računarstvo i informacione tehnologije (Univerzitet Zapadnoj Sidneja, Australija).

## VLabNet

Projekat je zasnovan na Debian operativnom sistemu, Xen softveru za virtuelizaciju i Quagga Routing Suite softveru za rutiranje. Ovo okruženje predstavlja jedan od prvih pokušaja kombinovanja virtuelnih mašina i fizičke mrežne opreme. U okviru laboratorije je integrisan i fizički ruter Cisco 2610. Ova karakteristika je dala dobar primer kombinovanja virtuelnih i klasičnih rutera u formiranju laboratorije [PDJ07, PTW07].

Ceo sistem se nalazi na serveru sledeće konfiguracije: HP ML370G3 računar sa 2.8 GHz CPU, dva diska 36 GB SCSI konfigurisana kao hardverski RAID-1 *mirror* sistem i 1 GB RAM-a. Ovaj server

pruža podršku za rad do 22 virtualne mašine od kojih svaka ima svojih 48 MB RAM-a i 1 GB hard disk. Kapacitet RAM-a je ograničavajući faktor, zato što i softver za virtuelizaciju Xen koristi za virtuelne mašine RAM koji pripada serveru. Kapacitet memorije od 48 MB pokazalo se kao dovoljna količina memorije za rad virtuelnih mašina. CPU hosta i prostor na disku nisi bili problem u toku rada. Arhitektura sistema prikazana je na slici 3.7.



Slika 3.7. Prikaz infrastrukture VLabNet sistema

### **Manage Large Networks (MLN)**

MLN je open-source paket koji predstavlja alat za definisanje računarskih mreža. Sistem koristi User-Mode Linux (UML) ili Xen kao softver za virtuelizaciju i različite virtuelne mašine bazirane na različitim Linux distribucijama. Razvijen je i MLN jezik koji olakšava kreiranje mrežnih scenarija i konfiguracija. Studentima je omogućeno da koriste ovaj sistem upotrebom skript datoteka pisanim u tom jeziku [GLA08a, GLA08b, Beg06].

Specifikacija konfiguracije čvorova i mrežnih scenarija je u formi tekstualnog dokumenta. Imajući u vidu parametre koji se odnose na konfiguraciju čvorova mreže, opis je veoma detaljan. MLN pruža mogućnost za detaljan opis konfiguracije mrežnog interfejsa, podršku za nasleđivanje i uključivanje drugih dokumenata i dr. Sve to je realizovano u svrhu postizanja lakšeg održavanja velikih mrežnih scenarija (topologija). Slično kao VNUML, MLN parser podržava kreiranje i pokretanje scenarija. Za sada ne postoji MLN grafički interfejs. MLN je jezik koji se može proširiti po potrebi, tj. jezik dozvoljava dodavanje novih elemenata. MLN parser arhitektura je *plug-in* orijentisana, a dodavanje *plug-in* dodataka je moguće na osnovu datog API-ja. Ova karakteristika omogućava dalje usavršavanje parsera, a samim tim i alata.

## **Marionnet**

Marionnet [LS07, LS08] je virtuelna mrežna laboratorija koja omogućava korisnicima definisanje, konfigurisanje i pokretanje kompleksnih mreža sa mrežnim uređajima kao što su komutatori (svičevi, eng. *switch*), koncentratori (hubovi, eng. *hub*), ruteri i kablovi. Marionnet je besplatna aplikacija razvijena za Linux platformu i zasnovana na UML i VDE [Dav05] alatima za virtuelizaciju.

## **vBET**

vBET sistem [JX03] zasnovan je na UML-u. Korisnici opisuju scenario upotrebom tekstualno orijentisanog jezika za specifikaciju. Parser obrađuje specifikaciju i generiše skript, koji se zatim izvršava, što dovodi do stvaranje virtuelnog čvora i virtuelne topologije. Scenario se može poništiti po završetku eksperimenta. vBET jezik je sličan NetML i MLN jezicima jer ne dozvoljava automatizaciju izvršavanja komandi na virtuelnim mašinama.

## **Dynagen**

Dynagen [Anu07] je različit u odnosu na svoje prethodnike. Taj sistem je orijentisan na kreiranje scenarija sačinjenih od emuliranih Cisco rutera koji su dobijeni upotrebom Dynamips tehnologije. Kao neka od ranije opisanih okruženja i ovo okruženje je bazirano na tekstualnom jeziku, koji omogućava specifikaciju broja i karakteristike rutera, kao što su brojevi portova, tehnologija povezivanja itd. Istovremeno, obuhvata i interfejs za startovanje i stopiranje rada rutera.

Prednost Dynagen je visok nivo realnosti u scenarijima. Veliki nedostatak sistema je njegova nemogućnost da emulira ni jednu drugu platformu i tip čvora (server, radna stanica itd.) osim Cisco rutera.

## **Ostale mrežne laboratorije**

Ostale virtuelne mrežne laboratorije / okruženja su Cloonix-Net (baziran na UML i KVM) i GINI razvijen na *Advanced Networking Research Laboratory (McGill University School of Computer Science)*. GINI (GINI is not Internet) je alat za kreiranje virtuelnih mikro Internet okruženja za učenje tema iz oblasti računarskih mreža [MMN09]. Ima lak grafički korisnički interfejs koji se naziva gBuilder, a koristi se za kreiranje mrežnih instanci. Pored navedenih tu su još i NVLab [WN09a, WN09b, WNN11] i mnoge drugi sistemi [Kra04, NO06, ACM07, DW08, FMJ02, GT07].

### **3.3. Softverske komponente virtuelnih mrežnih laboratorija**

U softverske komponente virtuelnih mrežnih laboratorija spadaju:

- softveri za virtuelizaciju
- softveri koji omogućavaju infrastrukturu (serverski i pristupni softver za virtuelnu laboratoriju)
- softveri koji omogućavaju osnovnu funkcionalnost virtuelnih mašina (operativni sistemi, serverski i aplikativni softver)
- softveri za rutiranje (multiprotokol ruting softveri)

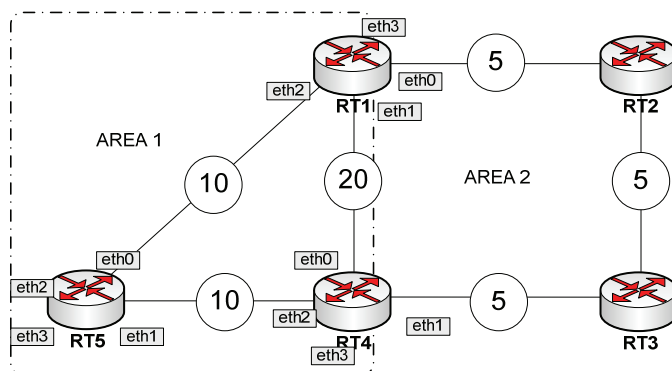


Softveri za virtuelizaciju su opisani u prethodnoj sekciji. Serverski i pristupni softver, tj. softver koji omogućava instalaciju, pokretanje i upravljanje mrežnom laboratorijom, virtuelnim mašinama i pristup toj laboratoriji, biće opisani u narednim poglavljima u delu koji se odnosi na opis arhitekture i softverskih komponenata virtuelne mrežne laboratorije - VNLab. Isto je i sa softverom koji omogućavaju osnovnu funkcionalnost virtuelnih mašina, tj. njihovu ulogu kao radne stanice, veb ili mail servera. U tu grupu softvera spada standardni aplikativni i serverski softver, a korišćeni softveri u ovom okruženju biće navedeni u narednim poglavljima.

U ovom poglavlju će biti opisana samo najvažnija, specifična softverska komponenta virtuelnih mrežnih laboratorija, tj. samo softver koji virtuelnim mašinama omogućava emulaciju rada rutera – multiprotokol rutinng softveri [DO06b]. S obzirom na to da i u ovom slučaju postoji veći broj mogućih rešenja u narednom delu teksta biće opisani najkarakterističniji predstavnici ove vrste softvera. Postoji velikih broj besplatnih paketa koji se mogu koristiti za tu svrhu, a to su:

- GNU Zebra [Zeb11]
- Quagga [Qua11]
- XORP (eXtensible Open Router Platform) [Xor11]
- BIRD (BIRD Internet Routing Daemon) [BIR11]

**GNU Zebra** je besplatan multiprotokol rutinng softver [Ram11]. Predstavlja veoma popularnu platformu. To je softver za dinamičko rutiranje sa podrškom za BGP-4 [RLH06], BGP-4+, OSPFv2 [Moy98], OSPFv3 [CFM08], RIPv1 [Hen98], RIPv2 [Mal98] i RIPng [MM97] protokole u IPv4 [RFC971] i IPv6 [DH98] mrežama. Distribuira se pod GNU General Public License i namenjen je za upotrebu na Linux i drugim UNIX platformama. Besplatna verzija ovog softvera prestala je da se razvija 2005. god. Zaključno sa verzijom 0.95. Naslednik ovog softvera je Quagga.



Slika 3.8. Topologija mreže sa Zebra ruterima i OSPF protokolom

Upotrebljivost ovog softvera je utoliko veća zbog velike sličnosti sa Cisco IOS interfejsom [Cis06, Bon01, Sed01]. U primeru, radi poređenja sa Cisco IOS interfejsom, data je konfiguracija jednog Zebra rutera (Listing 3.2. i Listing 3.3.) za OSPF protokol u NSSA područjima za mrežu prikazanu na slici 3.8.

```
ospfd> enable
ospfd# configure terminal
ospfd(config)# router ospf
ospfd(config-router)# passive-interface eth3
ospfd(config-router)# network 192.168.41.0/24 area 1
ospfd(config-router)# network 172.16.14.0/30 area 1
```

```

ospfd(config-router)# network 172.16.34.0/30 area 0
ospfd(config-router)# network 172.16.45.0/30 area 1
ospfd(config-router)# area 1 stub no-summary
ospfd(config-router)# exit
ospfd(config)# interface eth0
ospfd(config-if)# ip ospf cost 20
ospfd(config-if)# exit
ospfd(config)# interface eth1
ospfd(config-if)# ip ospf cost 5
ospfd(config-if)# exit
ospfd(config)# interface eth2
ospfd(config-if)# ip ospf cost 10
ospfd(config-if)# exit
ospfd(config)# write
Configuration saved to /usr/local/etc/ospfd.conf
ospfd(config)# exit
ospfd# exit

```

### Listing 3.2. Konfiguracija rutera RT5 sa slike

Prikaz konfiguracije Zebra rutera vrši se komandom *show running-config* (Listing 3.4.)

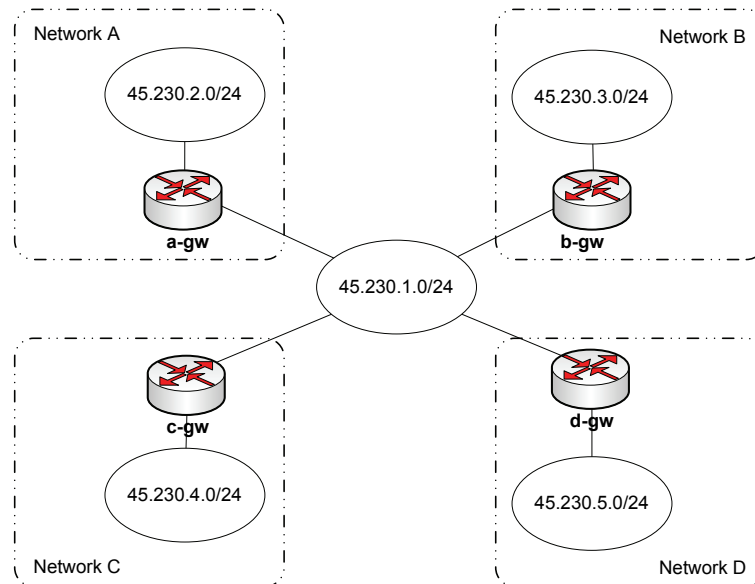
```

ospfd(config)# show running-config
Current configuration:
!
hostname ospfd
password zebra
log stdout
!
!
!
interface lo
!
interface eth0
ip ospf cost 20
!
interface eth1
ip ospf cost 5
!
interface eth2
ip ospf cost 10
!
interface eth3
!
router ospf
passive-interface eth3
network 172.16.14.0/30 area 0.0.0.1
network 172.16.34.0/30 area 0.0.0.0
network 172.16.45.0/30 area 0.0.0.1
network 192.168.41.0/24 area 0.0.0.1
!
line vty
!
End

```

Listing 3.3. Primer Zebra konfiguracije dobijen *show running-config* komandom

**XORP (eXtensible Open Router Platform)** je softver za rutiranje koji je nastao sa osnovnim ciljem da omogući stabilnu platformu za komercijalnu upotrebu i dovoljno fleksibilnu platformu za istraživanja [HHK03]. Podržava veliki broj rutirajućih protokola za IPv4 i IPv6 mreže, uključujući BGP, RIP, PM-SM, IGMP/MLD. Takođe, sistem podržava i prošireni programski API. Nastao je na Međunarodnom institutu za računarne nauke Berkli (eng. *Berkeley, USA*), ali projekat uključuje i veliki broj učesnika projekta iz celog sveta. Distribuirana se pod licencom sličnoj BSD licenci i razvijen je u programskom jeziku C++. Primer konfiguracije (Listing 3.4.) prikazan je za ruter u mreži (Slika 3.9.).



Slika 3.9. Topologija mreže sa XORP ruterima i BGP protokolom

```

[root@gw-corp-b ~]# more xorp.boot
interfaces {
interface eth0 {
  description: "core interface"
  disable: false
  /*default-system-config*/
  vif eth0 {
    disable: false
    address 45.230.1.3 {
      prefix-length: 24
      broadcast: 45.230.1.255
      disable: false
    }
  }
}
interface eth1 {
  description: "LAN interface"
  disable: false
  /*default-system-config*/
  vif eth1 {
    disable: false
    address 45.230.3.1 {
      prefix-length: 24
      broadcast: 45.230.3.255
      disable: false
    }
  }
}
}
fea {
unicast-forwarding4 {

```

```

        disable: false
    }
}
protocols {
    static {
        route4 0.0.0.0/0 {
            next-hop: 45.230.1.1
            metric: 1
        }
    }
}
bgp {
    bgp-id: 45.230.1.3
    local-as: 65003
    peer 45.230.1.2 {
        local-ip: 45.230.1.3
        as: 65002
        next-hop: 45.230.1.2
    }
    peer 45.230.1.4 {
        local-ip: 45.230.1.3
        as: 65004
        next-hop: 45.230.1.4
    }
    peer 45.230.1.5 {
        local-ip: 45.230.1.3
        as: 65005
        next-hop: 45.230.1.5
    }
}
network4 45.230.3.0/24 {
    unicast: true
    next-hop: 45.230.1.3
}
}
}

```

Listing 3.4. Primer XORP konfiguracije

**Quagga** [Qua11] je softver nastao usled razvoja Zebra paketa i ima podršku za slične protokole kao i njegov prethodnik. Kao i prethodni softveri, distribuirao se pod GPL licencom, a razvijen je na programskom jeziku C. Ovaj projekat je u stalnom razvoju, a jedan od njegovih tekućih razvojnih pravaca je implementacija podrške za MPLS [RR06]. Sličan je Zebra konfiguraciji (Listing 3.2).

**BIRD (BIRD Internet Routing Daemon)** je paket razvijen sa sličnim ciljem kao i prethodno opisani paketi. Namenjen je za Unix platforme i ima podršku za IPv4 i IPv6 mreže i BGP, RIP, OSPF protokole u IPv4 okruženju. Softverski paket BIRD podržava i statičko rutiranje, višestruke tabele rutiranja, inter-table protokole itd. Ima veliki broj planiranih pravaca razvoja. Razvijen je kao školski projekat na Fakultetu za matematiku i fiziku u okviru Karlovog univerziteta (Prag, Češka) i distribuirao se pod GNU General Public License [Bir11].

Zbog lakoće instalacije, konfiguracije i korišćenja, zbog pouzdanosti i tačnosti u radu, a najviše zbog sličnosti sa Cisco IOS interfejsom i pozitivnih iskustava u toku rada, za upotrebu u ovom istraživanju odabrana je GNU Zebra. Quagga, novija varijanta ovog softvera, takođe se može koristiti sa istim uspehom i na isti način.

#### **4. Modeliranje i realizacija virtuelne mrežne laboratorije sa lokalnim pristupom**

Ovo poglavlje opisuje početnu fazu u razvoju modela virtuelne mrežne laboratorije. Model laboratorije je razvijen na Tehničkom fakultetu "Mihajlo Pupin" u Zrenjaninu - Univerziteta u Novom Sadu za upotrebu u nastavi na predmetu Računarske mreže. Sam početak razvoja inspirisan je sve većom upotrebom tehnologije za virtuelizaciju u kreiranju laboratorija za računarske mreže. Jedan broj projekata (VNUML, Netkit i dr.) koji su uticali na razvoj ovog modela opisani su u prethodnom poglavlju. Drugi izvor ideja za razvoj modela inspirisan je na osnovu scenarija za učenje mrežnih koncepata koji se koriste u okviru simulacionih softvera za računarske mreže – OPNET IT Guru Academic Edition [Opn11].

U početnoj fazi razvoja modela, postojao je niz zahteva i uslova koji vladaju u okruženju, a koji su bitno uticali na dalje pravce razvoja i modeliranje laboratorije. Prvi zahtev je bio da se laboratorija razvija specifično za predmet Računarske mreže, ali se ostavlja mogućnost za kreiranje fleksibilne laboratorije koja se može koristiti i u drugim predmetima. Uporeba u okviru predmeta Računarske mreže može biti vrlo raznovrsna, pa je odlučeno da se laboratorija fokusira na rad sa TCP/IP mrežama i upoznavanje studenata sa njihovim funkcionisanjem, konceptima kao i mrežnim softverom i servisima. Laboratorija treba da podrži rad sa IPv4 i IPv6 verzijama protokola. Sledeći bitan koncept bilo je statičko i dinamičko rutiranje. U cilju efikasnog i funkcionalnog izučavanja koncepta dinamičkog rutiranja laboratorija je morala da podrži rad sa RIPv2, RIPng, OSPFv2, OSPFv3 i BGP-4 verzijama protokola. Za učenje fajervol (eng. *firewall*) sistema [ZCC00], laboratorija je morala da pokrije oblast filtriranja paketa (eng. *packet filtering*), prosleđivanje portova (eng. *port forwarding*) i NAT (Network Address Translation). Na kraju, ostali su i mrežni servisi i serverski softver koji je potrebno obuhvatiti ovom laboratorijom, pa se tu nametnuo rad sa veb servisom ili HTTP protokolom, FTP protokolom, DNS i mail servisom (prvenstveno sa SMTP ali i eventualno POP3 protokolom).

Drugi zahtev koji je imao važan uticaj na razvoj modela, a kasnije i njegovu realizaciju, bio je u pogledu platforme koja će se koristiti. Virtuelna laboratorija treba da bude instalirana na računarima koji se nalaze u laboratorijama za nastavu iz predmeta Računarske mreže. U tim laboratorijama se vrši nastava i iz drugih predmeta (Operativni sistemi, Računarska grafika, Veštačka inteligencija, i dr.) Zbog toga, laboratorija je trebala da koristi Windows operativni sistem zbog toga što se taj operativni sistem koristi i na navedenim predmetima.

Na kraju, postojao je još jedan zahtev. Zbog nedovoljnih finansijskih fondova bilo je potrebno pronaći rešenje koje je jeftino. Cena je bila primarni faktor koji je definisao okruženje za virtuelnu laboratoriju, kako po pitanju softvera tako i po pitanju kupovine hardvera.

##### **4.1. Modeliranje laboratorije sa lokalnim pristupom**

Sa obzirom na navedene zahteve i ograničenja, u početnoj fazi razvoja, izbor je bio ograničen na VMWare i Microsoft Virtual PC softver za virtuelizaciju. To je bilo uslovljeno prvenstveno Windows platformom koja se nalazila na radnim stanicama. Bez upoređivanja njihovih performansi, odlučeno je da to bude Microsoft Virtual PC 2004, jer je u to vreme pa sve do danas pitanje legalne upotrebe tog softvera na Fakultetu bilo rešeno. Fakultet je pretplatnik MSDN Academic Alliance programa, te može legalno da koristi Microsoft proizvode.

Posle odluke o odgovarajućem alatu za virtuelizaciju, prešlo se na planiranje i odabir guest operativnog sistema (Guest OS). Ovo pitanje je bilo lakše, jer se moglo razmišljati o upotrebi više operativnih sistema radi bolje funkcionalnosti laboratorije i tako vršiti njihovo upoređivanje u

toku same eksploatacije sistema. Svaki od odabranih operativnih sistema instalirao se na svaku virtuelnu mašinu pojedinačno. Korišćeni operativni sistemi su opisani u narednim poglavljima prilikom opisa laboratorije. Ono što je bitno uticalo na odabir guest operativnog sistema bili su sledeći zahtevi:

- operativni sistem mora da zadovolji punu funkcionalnost laboratorije, tj. implementaciju i primenu većeg broja mrežnih koncepata unutar nje,
- dostupnost softvera za taj sistem treba da je velika, a softver po mogućstvu treba da bude open-source tipa zbog izbegavanja upotrebe nelicenciranog softvera. Ovo je dosta važno pitanje jer operativni sistem treba da omogući emulaciju različitih tipova mrežnih uređaja, od servera i radnih stanica do specijalizovanih mrežnih uređaja kao što su ruteri, svičevi i habovi,
- minimalni resursi koje taj sistem zahteva za rad. Ovaj faktor je razumljiv zato što je za funkcionalnu laboratoriju potrebno instalirati što veći broj računara na istoj fizičkoj mašini. Minimalno zauzimanje resursa od svake virtuelne mašine tako postaje osnovni faktor koji utiče na upotrebljivost laboratorije i mogućnost njenog prilagođavanja potrebnim mrežnim konceptima.

Logičan korak je bio upotreba nekog od postojećih distribucija Linux operativnog sistema. Iako postoji veliki broj sistema koji zadovoljavaju potrebne kriterijume, jedan od operativnih sistema koji je testiran među prvima je Trustix Linux Server 2.2, sličan Red Hat distribuciji. To je kompaktan operativni sistem sa veoma malom instalacijom i minimalnim hardverskim zahtevima. Minimalni zahtevi u pogledu memorijskih resursa potrebnih za rad, takođe su jedna od bitnih karakteristika ovog operativnog sistema. Ta prednost će biti izražena u drugoj fazi razvoja virtuelne laboratorije prilagođenje za pristup na daljinu.

Dodatna funkcionalnost platforme je postignuta dostupnošću potrebnog softvera za kreiranje potrebnih mrežnih scenarija. Trustix Linux, kao i druge Linux distribucije imaju veliki izbor raznovrsnog softvera na raspolaganju koji je open-source. Potreban softver za izvođenje laboratorijskih vežbi iz mrežnih servisa i serverskog softvera (veb, FTP, DNS itd.) bio je dodatna opcija za razmatranje. Za navedene servise upotrebljen je standardan softver. Za veb servis instaliran je *apache* server [LL02], za mail servis instaliran je *sendmail* softver [Cos02], za FTP instaliran je *proftpd* [Pro11], a za DNS je instaliran *bind* [LA07]. Svi ovi softveri predstavljaju široko rasprostranjene i često upotrebljavane softvere u realnim okruženjima na Internetu, a lako se mogu preuzeti sa zvaničnih veb sajtova organizacija koje ih razvijaju.

Pored instaliranog serverskog softvera, ne treba zaboraviti potrebu da virtuelne mašine treba da emuliraju i rad rutera. Ruteri su nezaobilazne komponente mreža i bez njih kreiranje bilo kakve laboratorije za računarske mreže ne bi bilo moguće. Linux računari mogu biti efikasni ruteri, a to je omogućeno upotrebom nekog od multiprotokol ruting softvera, koji su opisani u prethodnom poglavlju. I pored velikog izbora (Zebra, Quagga i XORP), posle istraživanja odlučeno je da se u ovu svrhu koristi open-source softver GNU Zebra, koji ima podršku za RIPv2 [Hen98], RIPv6 [MM97], OSPFv2 [Moy98], OSPFv3 i BGP-4 [RLH06] protokole za dinamičko rutiranje.

Na kraju, softver koji je potreba za analizu rada mreže i njenu osnovnu konfiguraciju nalazio se u okviru standardne instalacije Linux operativnog sistema. Prvenstveno se radi o alatima *ping* i *traceroute* za analizu podešenih putanja u TCP/IP mreži; komandama *netconfig* i *ifconfig* za podešavanje IP adresa i njihovu analizu; komande *tcpdump* za analizu saobraćaja; komande *iptables* za podešavanje fajervola; i ostalim komandama kao što su *netstat*, *arp* itd [SWF05].

Formiranje laboratorije vrši se tako što se prvo formira jedna virtuelna mašina (Slika 4.1). Na nju se instalira Trustix operativni sistem, a zatim GNU Zebra i potreban serverski softver.

Da bi se od svake virtuelne mašine napravio ruter potrebno je instalirati GNU Zebra Free Routing softver. Struktura softvera je takva da se njegov rad zasniva na glavnom rutingu serverskom softveru pod nazivom *zebra* (Slika 4.2.). Ostali serverski softveri se pokreću nezavisno. Za svaki od protokola koristi se poseban serverski softver. Tako se za OSPF protokol postoji *ospfd*, a za RIP *ripd*. Konfiguracione datoteke Zebra softvera nalaze se u `/usr/local/etc` direktorijumu. Glavni konfiguraciona datoteka je *zebra.conf*, dok se ostale datoteke nazivaju po serverskom softveru, npr. za OSPF protokol konfiguraciona datoteka se zove *ospfd.conf*.

Po završetku instalacije potrebno je izvršiti izmenu `/etc/services` datoteke, tj. potrebno je dodeliti svakom Zebra servisu odgovarajući port. Izmena datoteke vrši se dodavanjem sledećih vrednosti.

```
zebrasrv      2600/tcp      # zebra service
zebra         2601/tcp      # zebra vty
ripd          2602/tcp      # RIPd vty
ripngd        2603/tcp      # RIPngd vty
ospfd         2604/tcp      # OSPFd vty
bgpd          2605/tcp      # BGPd vty
ospf6d        2606/tcp      # OSPF6d vty
```

Podizanje Zebra servisa vrši se komandom *zebra -d*. Ovo je neophodno da se uradi pre startovanja bilo kog drugog servisa. Tek kada je zebra servis podignut, može se startovati npr. OSPF protokol komandom *ospfd -d*. Posle izvršenih izmena i startovanja servisa moguće je pristupiti konfiguraciji softvera za rutiranje pomoću telnet klijenta sa lokalnog ili udaljenog računara – npr. *telnet localhost 2604*.

```
root@RT1 ~# telnet localhost ospfd
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.

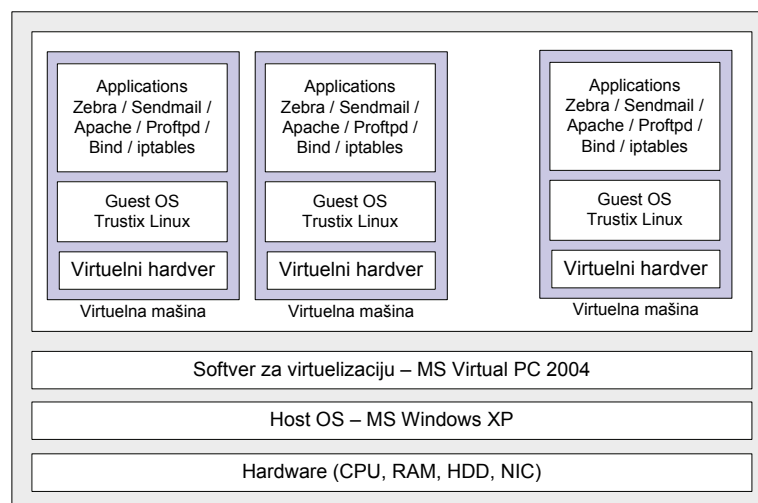
Hello, this is zebra (version 0.94).
Copyright 1996-2002 Kunihiro Ishiguro.
```

```
User Access Verification
```

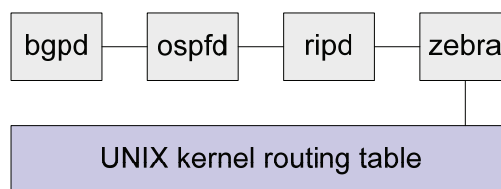
```
Password:
```

Zebra interfejs je skoro isti kao Cisco IOS interfejs, tako da se studentima pruža mogućnost da steknu iskustva u radu sa ovim popularnim sistemom [Cis06, Bon01, Sed01].

Virtuelne mašine se sastoje iz dve datoteke – to su `.vhd` i `.vmc` datoteke. U prvoj datoteci nalazi se virtuelni hard disk mašine, a u drugom njena konfiguracija. Navedena kombinacija softvera koji su instalirani na virtuelne mašine radi dobro u Virtual PC 2004 okruženju. Virtuelne mašine koje su upotrebljene u ovoj laboratoriji imaju datoteku `.vhd` veličine nešto ispod 400 MB, koliko je potrebno za instalaciju Trustix operativnog sistema sa kompletnom podrškom za rad. Virtuelne mašine se podižu brzo, a svaka od njih zahteva samo 16 MB RAM memorije da bi mogla da funkcioniše.



Slika 4.1. Model virtuelne mrežne laboratorije sa lokalnim pristupom



Slika 4.2. Arhitektura GNU Zebra softvera

Ostale virtuelne mašine kreiraju se kopiranjem ove dve datoteke pod drugim imenom i dodavanjem novoformiranih virtuelnih mašina opcijom *New Virtual Machine Wizard* u Virtual PC2004 softveru. Na kraju, potrebno je promeniti odgovarajuće konfiguracione datoteke operativnog sistema na virtuelnim mašinama, tj. potrebno je promeniti IP adrese i sl.

#### 4.2. Modeliranje mrežnih scenarija za virtuelnu mrežnu laboratoriju

Nakon modeliranja laboratorije, za njenu primenu u nastavi potrebno je modelirati mrežna scenarija podesna za učenje predviđenih mrežnih koncepata. Laboratorijske vežbe i rad studenata zasnivaju se na ovim scenarijima. Od scenarija zavisi nivo znanja studenata. Modeliranje mrežnih scenarija i konfiguracija računara u njima tako dobijaju na značaju i imaju uticaj na čitav životni ciklus laboratorije, u fazi implementacije, ali i fazama eksploatacije i izmene modela.

Kao osnova za kreiranje mrežnih scenarija nove laboratorije odabrane su laboratorijske vežbe specijalizovanog softvera za simulaciju računarskih sistema pod nazivom OPNET IT Guru Academic Edition [Opn11, TBL02, TLP03, PTT01, PTR02].

OPNET IT Guru Academic Edition softver pruža mogućnost kreiranja simuliranog okruženja za modelovanje, analizu i predviđanje performansi IKT infrastruktura, uključujući aplikacije, servere, i mrežne tehnologije. Baziran na OPNET IT Guru softveru, Academic Edition je dizajniran za dopunu specifičnih laboratorijskih vežbi koje služe za učenje fundamentalnih mrežnih koncepata. Komercijalna verzija OPNET IT Guru softvera ima šire mogućnosti i dizajnirana je za kompanije i njihovu IT strukturu, dokumentovanje i profesionalnu podršku. OPNET softver se koristi u hiljadama preduzeća i vladinih organizacija širom sveta, i na oko 500 univerziteta u nastavi i istraživanju. OPNET IT Guru – Academic Edition, je program za



simulaciju diskretnih događaja (eng. *discrete event simulator*) u računarskim mrežama i za njega je razvijen veliki broj simulacija i simulacionih scenarija koji se koriste u nastavi na univerzitetima [WLW09]. Veliki broj laboratorijskih vežbi objavljen je kao publikacija i u formi udžbenika. Određeni broj vežbi se može preuzeti i sa OPNET veb sajta (adresa: [http://www.opnet.com/university\\_program/teaching\\_with\\_opnet/textbooks\\_and\\_materials/index.html](http://www.opnet.com/university_program/teaching_with_opnet/textbooks_and_materials/index.html)), a spisak udžbenika i nastavnih materijala dostupnih sa sajta dat je u nastavku:

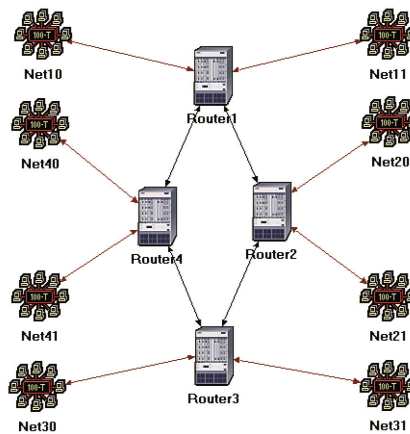
- Business Data Communications – 5 izdanje - Vilijam Stalings (William Stallings) - knjiga i onlajn laboratorijski priručnik [Sta04]
- Business Data Communications and Networking – 9 izdanje – Džeri Ficdžerald i Alan Denis (Jerry Fitzgerald & Alan Dennis) - knjiga i onlajn laboratorijski priručnik [FD09]
- Business Data Networks and Telecommunications – 4. Izdanje – Rejmond R. Panko (Raymond R. Panko) - knjiga i onlajn laboratorijski priručnik [PP11]
- Computer Networks - A Systems Approach - 3. Izdanje – Lari L. Peterson i Brus S. Dejvi (Larry L. Peterson & Bruce S. Davie) - knjiga i onlajn laboratorijski priručnik [PD07]
- Data and Computer Communications – 7. Izdanje - Vilijam Stalings (William Stallings) [Sta05]
- OPNET IT Guru Academic Edition – A Tool for Networking Education – Regis univerzitet, SAD (onlajn laboratorijski priručnik)
- Security Labs in IT Guru Academic Edition – Univerzitet Ramon Ljulj, Španija (*Universitat Ramon Llull*) (onlajn laboratorijski priručnik)
- Simulacao Computacional para Redes de Computadores – prvo izdanje - Paulo Dias de Alekrim (Paulo Dias de Alekrim) – knjiga [Ale09]

#### **4.2.1. Modeliranje scenarija za RIP protokol**

Prva laboratorijska vežba koja je modelirana za virtuelnu mrežnu laboratoriju zasnovana je na vežbi koja obuhvata rad sa RIP protokolom. RIP (Routing Information Protocol) spada u grupu protokola za dinamičko rutiranje koji se zasniva na distance-vector Bellman-Ford algoritmu [Mal98].

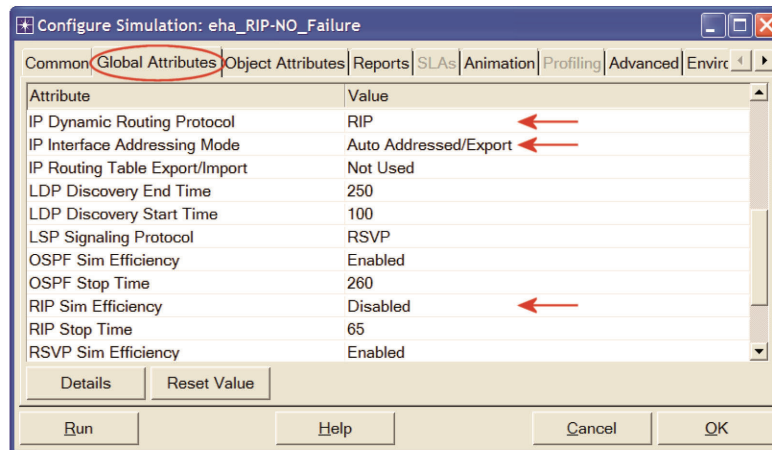
##### ***Vežba za RIP protokol modelirana u OPNET IT Guru Academic Edition softveru***

Mrežni scenario kreiran u OPNET IT Guru Academic Edition softveru izgleda kao što je prikazano na slici (Slika 4.3.). Na slici se vidi 4 rutera koja koriste RIP protokol, a na njih je povezano osam zasebnih IP mreža [Abo03].



Slika 4.3. Mrežni scenario za laboratorijsku vežbu sa RIP protokolom

Konfiguracija simulacije i rutera se vrši posle kreiranja mreže. Definisane parametara simulacije je jednostavno i vrši se kao što je prikazano na slici (Slika 4.4.).



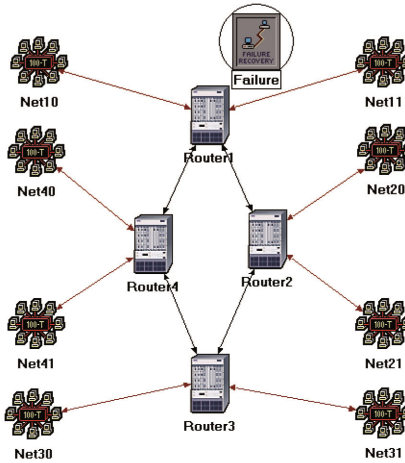
Slika 4.4. Prozor za konfiguraciju RIP parametara

Izgled tabele rutiranja (Slika 4.5.) dostupan je po završenoj simulaciji, kao i analiza pojedinih parametara rada mreže.

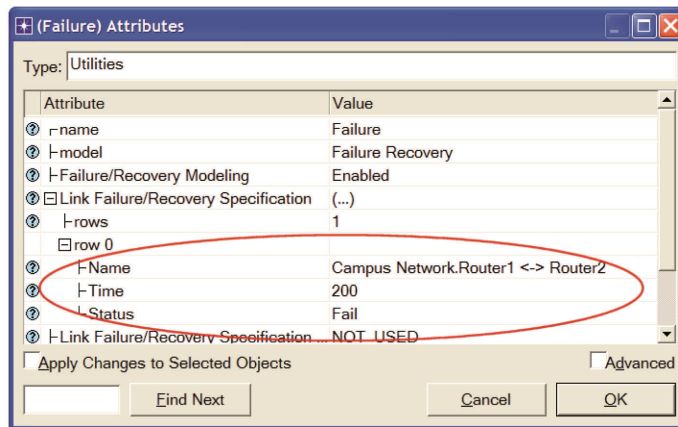
Node Name: Campus Network.Router1	Interface Name	Interface Index	IP Address	Subnet Mask	Connected Link
	IF0	0	192.0.0.1	255.255.255.0	Campus Network.Net11 <-> Router1
	IF1	1	192.0.1.1	255.255.255.0	Campus Network.Net10 <-> Router1
	IF10	10	192.0.2.1	255.255.255.0	Campus Network.Router1 <-> Router2
	IF11	11	192.0.3.1	255.255.255.0	Campus Network.Router4 <-> Router1
	Loopback	12	192.0.4.1	255.255.255.0	Not connected to any link.

Slika 4.5. Prikaz tabele rutiranja dobijene RIP protokolom

Vežba se može proširiti simulacijom prekida pojedinog linka ili kvara rutera. Kreiranje scenarija sa prestankom rada jednog rutera prikazano je na slici (Slika 4.6.), a podešavanje parametara za taj scenario na slici (Slika 4.7.)



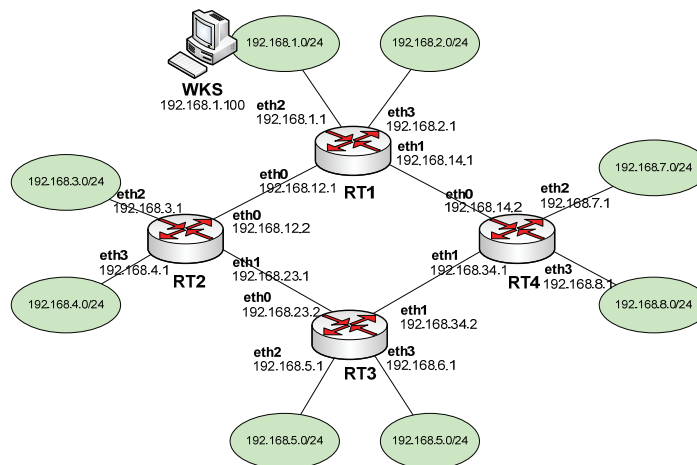
Slika 4.6. Proširenje scenarija sa prekidom rada rutera



Slika 4.7. Podešavanje parametara za scenario sa prekidom rada uređaja

### ***Vežba za RIP protokol kreirana u Virtualnoj laboratoriji sa lokalnim pristupom***

U ovom delu prikazana je adaptacija opisane vežbe iz programa OPNET IT Guru Academic Edition za okruženje virtualne mrežne laboratorije. Mrežni scenario i adresna šema adaptirane vežbe prikazane su na slici (Slika 4.8.). Pre početka vežbe studenti treba da startuju zebra i ripd programe pokretanjem komandi *zebra -d* i *ripd -d*.



Slika 4.8. Scenario za RIP protokol sa 4 rutera i jednom radnom stanicom

Mrežni scenario obuhvata 4 rutera i jednu radnu stanicu. GNU Zebra softver za rutiranje je instaliran na ruterima, dok se na radnoj stanici koristi samo statičko rutiranje, uz pomoć komade `route add -net`. Unete rute na Linux radnim stanicama mogu se prikazati komandom `route -n`, a rezultat ove komande ima sledeći prikaz:

```
root@trustix ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.1.0 0.0.0.0 255.255.255.0U 0 0 0 eth0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
0.0.0.0 192.168.1.1 0.0.0.0 UG 0 0 0 eth0
```

Student je instruisan kako i sa kojim komandama treba da konfiguriše svaki ruter u mreži. Konfiguracija GNU Zebra softvera i RIP protokola za jedan od rutera u mreži prikazana je na listingu 4.1. Konfiguracija svakog rutera čuva se u `rip.conf` datoteci koji se nalazi u `/usr/local/etc/` direktorijumu na Linux file sistemu.

```
root@trustix ~# cat /usr/local/etc/ripd.conf
! -*- rip -*-
!
! RIPd sample configuration file
!
! $Id: ripd.conf.sample,v 1.11 1999/02/19 17:28:42 developer Exp $
!
hostname ripd
password zebra
!
! debug rip events
! debug rip packet
!
router rip
redistribute connected
network eth0
network eth1
network eth2
network eth3
!
log file /var/log/zebra/ripd.log
!
log stdout
```

#### Listing 4.1. Konfiguraciona datoteka Zebra softvera za RT1 ruter u datom scenariju

Student koji trenutno radi na ruteru RT1, može da proveri da li je veza prema mreži `192.168.3.0/24` dostupna sa radne stanice. To se može proveriti upotrebom komande `ping` na interfejs `eth2` rutera RT2 koji ima IP adresu `192.168.3.1`.

```
root@trustix ~# ping 192.168.3.1
PING 192.168.3.1 (192.168.3.1) 56(84) bytes of data.
64 bytes from 192.168.3.1: icmp_seq=1 ttl=63 time=22.3 ms
64 bytes from 192.168.3.1: icmp_seq=2 ttl=63 time=2.66 ms
64 bytes from 192.168.3.1: icmp_seq=3 ttl=63 time=2.51 ms
64 bytes from 192.168.3.1: icmp_seq=4 ttl=63 time=2.49 ms
64 bytes from 192.168.3.1: icmp_seq=5 ttl=63 time=2.62 ms
64 bytes from 192.168.3.1: icmp_seq=6 ttl=63 time=2.93 ms

--- 192.168.3.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5057ms
rtt min/avg/max/mdev = 2.497/5.936/22.380/7.355 ms
```

Putanja prema hostu 192.168.3.1 može se prikazati i upotrebom komande *traceroute*.

```
root@trustix ~# traceroute 192.168.3.1 -n
traceroute to 192.168.3.1 (192.168.3.1), 30 hops max, 40 byte packets
1 192.168.1.1 9.241 ms 2.681 ms 6.512 ms
2 192.168.3.1 9.389 ms 4.566 ms 4.057 ms
```

Tabela rutiranja na ruteru RT2, koji ujedno predstavlja podrazumevani mrežni prolaz (eng. *default gateway*) za radnu stanicu WKS, može se dobiti korišćenjem prikazanih komandi, ili upotrebom komande *show ip rip* u Zebra softveru. Kao što je već rečeno, jedna od prednosti upotrebe ove komande je njena sličnost sa interfejsom i komandnim promptom koji je sličan Cisco CLI (Command Line Interface) okruženju. Cisco CLI i njegove komande predstavljaju najrasprostranjeniju platformu na svetu. Pristup ovoj komandi se može ostvariti pristupom na zebra servis uz pomoć komande *telnet* i odgovarajućeg porta, kao što je prikazano u primeru:

```
telnet localhost 2602
```

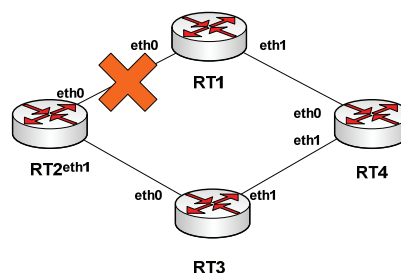
Prikaz ruta posle izvršene komande *show ip rip* u Zebra terminalu dat je u listing 4.2. Rute označene sa C su rute direktno povezane na mrežu, a rute sa prefiksom R su rute koje se dinamički menjaju usled informacija koje razmenjuje RIP protokol.

```
ripd> show ip rip
Codes: R - RIP, C - connected, O - OSPF, B - BGP
(n) - normal, (s) - static, (d) - default, (r) - redistribute, (i) - interface
```

Network	Next Hop	Metric	From	Time
C(i)	192.168.1.0/24	0.0.0.0	1	self
C(i)	192.168.2.0/24	0.0.0.0	1	self
R(n)	192.168.3.0/24	192.168.12.2	2	192.168.12.2 02:42
R(n)	192.168.4.0/24	192.168.12.2	2	192.168.12.2 02:42
R(n)	192.168.5.0/24	192.168.14.2	3	192.168.14.2 02:35
R(n)	192.168.6.0/24	192.168.14.2	3	192.168.14.2 02:35
R(n)	192.168.7.0/24	192.168.14.2	2	192.168.14.2 02:35
R(n)	192.168.8.0/24	192.168.14.2	2	192.168.14.2 02:35
C(i)	192.168.12.0/24	0.0.0.0	1	self
C(i)	192.168.14.0/24	0.0.0.0	1	self
R(n)	192.168.23.0/24	192.168.12.2	2	192.168.12.2 02:42
R(n)	192.168.34.0/24	192.168.14.2	2	192.168.14.2 02:35

Listing 4.2. Tabela rutiranja na RT1 ruteru prikazana u GNU Zebra softveru

Kao što je to urađeno i u OPNET IT Guru laboratorijskoj vežbi, i u ovom slučaju može se izvršiti simulacija prekida linka ili simulacija prestanka rada rutera. U nastavku će biti prikazan primer simulacije prekida linka između rutera RT1 i RT2. Prekid linka se može simulirati komandom *ifconfig eth0 down* na ruteru RT2, koja obara interfejs eth0 a samim tim i link prema ruteru RT1 (Slika 4.9).



Slika 4.9. Simulacija prekida linka

Posle te komande, komadna *ping 192.168.3.1* neće pokazati da postoji veza između rutera RT1 i RT2, a njen odziv će biti kao što je prikazano u nastavku:

```
root@trustix ~# ping 192.168.3.1
PING 192.168.3.1 (192.168.3.1) 56(84) bytes of data.

--- 192.168.3.1 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4017ms
```

Posle određenog vremena, promeniće se tabela rutiranja na posmatranom ruteru RT1, a prema mrežama 192.168.3.0/24 i 192.168.4.0/24 će sada voditi alternativne putanje. Umesto da putanje vode preko rutera RT1 i RT2, mreže će biti dostupne preko rutera RT4 i RT3 i sa drugačijim vrednostima za parametar *metric* kao što je prikazano na listingu 4.3.

```
ripd> show ip rip
Codes: R - RIP, C - connected, O - OSPF, B - BGP
(n) - normal, (s) - static, (d) - default, (r) - redistribute, (i) - interface

Network      Next Hop      Metric From   Time
C(i)  192.168.1.0/24  0.0.0.0      1      self
C(i)  192.168.2.0/24  0.0.0.0      1      self
R(n)  192.168.3.0/24  192.168.14.2 4      192.168.14.2 02:48
R(n)  192.168.4.0/24  192.168.14.2 4      192.168.14.2 02:48
R(n)  192.168.5.0/24  192.168.14.2 3      192.168.14.2 02:37
R(n)  192.168.6.0/24  192.168.14.2 3      192.168.14.2 02:37
R(n)  192.168.7.0/24  192.168.14.2 2      192.168.14.2 02:37
R(n)  192.168.8.0/24  192.168.14.2 2      192.168.14.2 02:37
C(i)  192.168.12.0/24 0.0.0.0      1      self
C(i)  192.168.14.0/24 0.0.0.0      1      self
R(n)  192.168.23.0/24 192.168.14.2 3      192.168.14.2 02:37
R(n)  192.168.34.0/24 192.168.14.2 2      192.168.14.2 02:37
```

Listing 4.3. Automatski izmenjena tabela rutiranja na ruteru RT1

Različita putanja prema navedenim mrežama se može prikazati i komandom *traceroute 192.168.3.1 -n* na radnoj stanici WKS.

```
root@trustix ~# traceroute 192.168.3.1 -n
traceroute to 192.168.3.1 (192.168.3.1), 30 hops max, 40 byte packets
 1 192.168.1.1 2.645 ms 1.762 ms 1.364 ms
 2 192.168.14.2 9.176 ms 8.379 ms 7.677 ms
 3 192.168.34.2 17.314 ms 15.075 ms 7.336 ms
 4 192.168.3.1 7.507 ms 7.588 ms 9.465 ms
```

Laboratorijska vežba se može nastaviti restauracijom prekinutog linka, što se može izvršiti komandom *ifconfig eth0 up*. Ubrzo posle primene komande, doći će do izmene tabele rutiranja i povratka na prvobitno stanje.

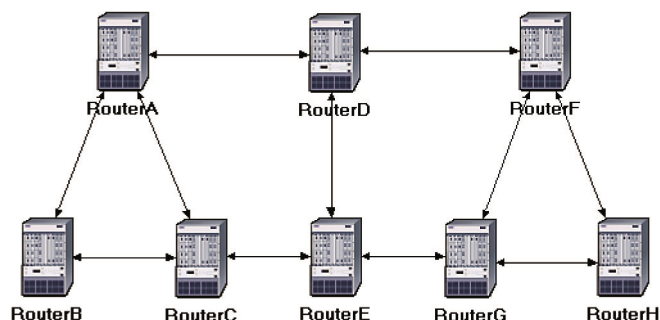
#### 4.2.2. Modeliranje scenarija za OSPF protokol

Cilj ove vežbe je upoznavanje studenata sa protokolom za dinamičko rutiranje - OSPF (Open Shortest Path First) [DO06c] i njegovim osobinama. OSPF, kao i RIP protokol, spada u unutrašnje protokole rutiranja (Interior Gateway Protocol - IGP) i koristi se za rutiranje unutar AS (autonomnog sistema). OSPF je najrasprostranjeniji protokol za rutiranje u unutrašnjim mrežama i spada u klasu *link-state* rutiranja protokola (protokola sa stanjem linkova). Zasniva se na Dijkstra algoritmu za proračunavanje najkraćih putanja. Zamišljen je kao naslednik RIP protokola, a blizak je IS-IS protokolu [Ora90]. Protokol postoji u dve verzije, jedna je OSPFv2 namenjena za IPv4 mreže, a druga je OSPFv3 namenjena za IPv6 mreže. OSPFv2 je definisan

dokumentom RFC 2328 [Moy98], a OSPFv3 dokumentom RFC 5340 [CFM08].

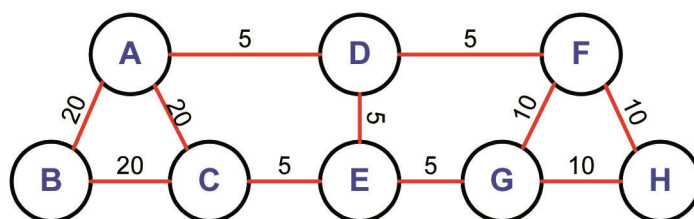
### ***Vežba za OSPF protokol kreirana u programu IT Guru Academic Edition***

Mrežni scenario kreiran u OPNET IT Guru Academic Edition softveru izgleda kao što je prikazano na slici (Slika 4.10.). Na slici se vidi 7 rutera koja koriste OSPF protokol [Abo03].



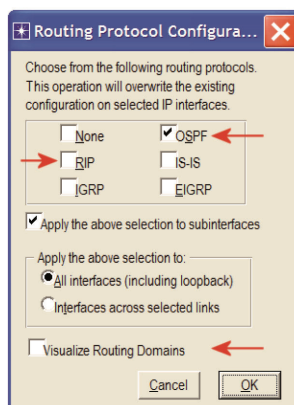
Slika 4.10. Topologija mreže za laboratorijsku vežbu za OSPF protokol

Kada se koristi OSPF, ruter proverava stanje linkova i njihovu cenu (*eng. path cost*). Ruta se postavlja preko najkraće putanje, tj. putanje sa najmanjom cenom. Kada se veza uspostavi ili prekine, generiše se objava o stanju veze (*eng. link-state advertisement - LSA*). Raspodela cena putanja između rutera prikazana je na slici 4.11.

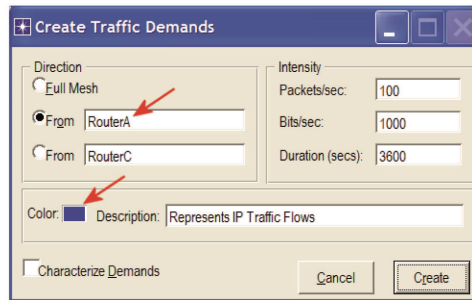


Slika 4.11. Cene putanja (path cost) u scenariju

U vežbi se može izvršiti uključivanje OSPF protokola na ruterima u simulaciji (Slika 4.12.), podešavanje područja, metrike interfejsa i definisanje parametara saobraćaja (Slika 4.13.).



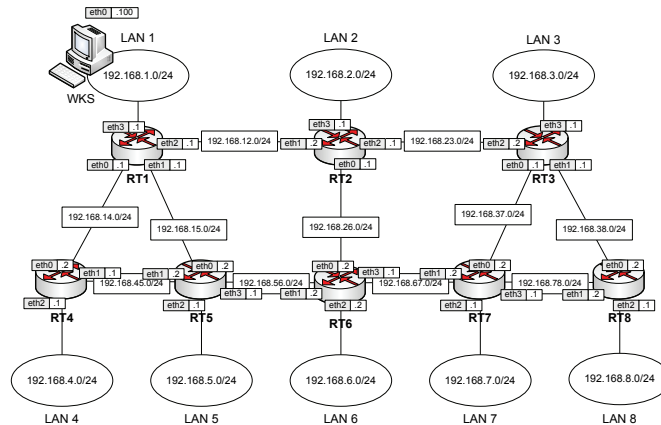
Slika 4.12. Uključivanje OSPF protokola



Slika 4.13. Podešavanje parametara saobraćaja u mreži

### Vežba za OSPF protokol modelirana u Virtuelnoj laboratoriji sa lokalnim pristupom

Adaptacija pokazane vežbe iz programa OPNET IT Guru Academic Edition za OSPF protokol u okruženju virtuelne mrežne laboratorije, mrežni scenario i adresna šema prikazane su na slici (Slika 4.14.). Pre početka vežbe studenti treba da startuju zebra i ospfd programe pokretanjem komandi *zebra -d* i *ospfd -d*. Definisane cene linka se vrši sa vrednostima koje su prikazane za vežbu u programu IT Guru Academic Edition (Slika 4.11.).



Slika 4.14. Mrežni scenario za virtuelnu mrežnu laboratoriju baziran na OSPF protokolu

Da bi se navedena virtuelna laboratorija mogla koristiti za simulaciju OSPF protokola na svim virtuelnim mašinama predviđenim za rutere potrebno je izvršiti konfiguraciju GNU Zebra softvera. Konfiguracija rutera RT1 može da se izvrši pomoću komandi prikazanih u listingu 4.4.

```
ospfd> enable
ospfd# configure terminal
ospfd(config)# router ospf
ospfd(config-router)# passive-interface eth3
ospfd(config-router)# network 192.168.14.0/24 area 0
ospfd(config-router)# network 192.168.15.0/24 area 0
ospfd(config-router)# network 192.168.12.0/24 area 0
ospfd(config-router)# network 192.168.1.0/24 area 0
ospfd(config-router)# exit
ospfd(config)# interface eth0
ospfd(config-if)# ip ospf cost 20
ospfd(config-if)# exit
ospfd(config)# interface eth1
ospfd(config-if)# ip ospf cost 20
ospfd(config-if)# exit
```



```
ospfd(config)# interface eth2
ospfd(config-if)# ip ospf cost 5
ospfd(config-if)# exit
ospfd(config)# write
Configuration saved to /usr/local/etc/ospfd.conf
ospfd(config)# exit
ospfd# exit
```

#### Listing 4.4. Komande za konfigurisanje rutera RT1

U toku konfiguracije vrši se asocijacija mreža koje se oglašavaju i njihova pripadnost odgovarajućem području (npr. *network 192.168.14.0/24 area 0*), definicija cene svakog linka (*ip ospf cost 20*) i označava se interfejs preko koga se ne oglašavaju rute (*passive-interface eth3*). Na sličan način potrebno je konfigurisati sve rutere u mreži sa odgovarajućim parametrima. Izvršena konfiguracija sačuvana je u */usr/local/etc/ospfd.conf* datoteci, a može se videti i komandom *show running-config* u ospfd terminalu kome se pristupa komandom *telnet localhost 2603*.

```
ospfd# show running-config

Current configuration:
!
hostname ospfd
password zebra
log stdout
!
!
!
interface lo
!
interface eth0
    ip ospf cost 20
!
interface eth1
ip ospf cost 20
!
interface eth2
ip ospf cost 5
!
interface eth3
!
router ospf
    passive-interface eth3
    network 192.168.1.0/24 area 0.0.0.0
    network 192.168.12.0/24 area 0.0.0.0

network 192.168.14.0/24 area 0.0.0.0
    network 192.168.15.0/24 area 0.0.0.0
!
line vty
!
end
```

#### Listing 4.5. Konfiguracija rutera RT1 prikazana komandom *show running-config*

Posle startovanja svih virtuelnih mašina koje predstavljaju čvorove u virtuelnoj laboratoriji može se početi sa radom na scenariju. Tabela rutiranja na ruteru RT1 se može videti pomoću Linux komande *route -n*.

```
root@RT1 ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.7.0 192.168.12.2 255.255.255.0UG 25 0 0 eth2
192.168.23.0 192.168.12.2 255.255.255.0UG 10 0 0 eth2
192.168.38.0 192.168.12.2 255.255.255.0UG 20 0 0 eth2
192.168.6.0 192.168.12.2 255.255.255.0UG 20 0 0 eth2
192.168.5.0 192.168.12.2 255.255.255.0UG 25 0 0 eth2
192.168.4.0 192.168.14.2 255.255.255.0UG 30 0 0 eth0
192.168.37.0 192.168.12.2 255.255.255.0UG 20 0 0 eth2
192.168.3.0 192.168.12.2 255.255.255.0UG 20 0 0 eth2
192.168.2.0 192.168.12.2 255.255.255.0UG 15 0 0 eth2
192.168.1.0 0.0.0.0 255.255.255.0U 0 0 0 eth3
192.168.67.0 192.168.12.2 255.255.255.0UG 15 0 0 eth2
192.168.15.0 0.0.0.0 255.255.255.0U 0 0 0 eth1
192.168.14.0 0.0.0.0 255.255.255.0U 0 0 0 eth0
192.168.12.0 0.0.0.0 255.255.255.0U 0 0 0 eth2
192.168.45.0 192.168.12.2 255.255.255.0UG 35 0 0 eth2
192.168.78.0 192.168.12.2 255.255.255.0UG 25 0 0 eth2
192.168.26.0 192.168.12.2 255.255.255.0UG 10 0 0 eth2
192.168.56.0 192.168.12.2 255.255.255.0UG 15 0 0 eth2
192.168.8.0 192.168.12.2 255.255.255.0UG 30 0 0 eth2
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```

Tabela rutiranja se može prikazati i u GNU Zebra terminalu komandom *show ip ospf route* (Listing 4.6.). Ostale informacije u vezi sa OSPF protokolom mogu se dobiti komandama *show ip ospf database*, *show ip ospf neighbor* i *show ip ospf interface*.

```
ospfd> show ip ospf route
===== OSPF network routing table =====
N 192.168.1.0/24 [10] area: 0.0.0.0
    directly attached to eth3
N 192.168.2.0/24 [15] area: 0.0.0.0
    via 192.168.12.2, eth2
N 192.168.3.0/24 [20] area: 0.0.0.0
    via 192.168.12.2, eth2
N 192.168.4.0/24 [30] area: 0.0.0.0
    via 192.168.14.2, eth0
N 192.168.5.0/24 [25] area: 0.0.0.0
    via 192.168.12.2, eth2
N 192.168.6.0/24 [20] area: 0.0.0.0
    via 192.168.12.2, eth2
N 192.168.7.0/24 [25] area: 0.0.0.0
    via 192.168.12.2, eth2
N 192.168.8.0/24 [30] area: 0.0.0.0
    via 192.168.12.2, eth2
N 192.168.12.0/24 [5] area: 0.0.0.0
    directly attached to eth2
N 192.168.14.0/24 [20] area: 0.0.0.0
    directly attached to eth0
N 192.168.15.0/24 [20] area: 0.0.0.0
    directly attached to eth1
N 192.168.23.0/24 [10] area: 0.0.0.0
    via 192.168.12.2, eth2
```

```

N      192.168.26.0/24      [10]  area: 0.0.0.0
        via 192.168.12.2, eth2
N      192.168.37.0/24      [20]  area: 0.0.0.0
        via 192.168.12.2, eth2
N      192.168.38.0/24      [20]  area: 0.0.0.0
        via 192.168.12.2, eth2
N      192.168.45.0/24      [35]  area: 0.0.0.0
        via 192.168.12.2, eth2
N      192.168.56.0/24      [15]  area: 0.0.0.0
        via 192.168.12.2, eth2
N      192.168.67.0/24      [15]  area: 0.0.0.0
        via 192.168.12.2, eth2
N      192.168.78.0/24      [25]  area: 0.0.0.0
        via 192.168.12.2, eth2

===== OSPF router routing table =====
===== OSPF external routing table =====

```

Listing 4.6. Tabela rutiranja na ruteru RT1 dobijena pomoću komande `show ip ospf route`

Za razliku od rutera na kojima se tabela rutiranja kreira dinamički, na radnoj stanici WKS konfigurirano je statičko rutiranje. Dodata je samo podrazumevana (*default*) ruta preko interfejsa eth3 rutera RT1 sa IP adresom 192.168.1.1. Tabela rutiranja za WKS je data u nastavku.

```

root@WKS ~# route -n
Kernel IP routing table
Destination Gateway      Genmask         Flags Metric Ref    Use    Iface
192.168.1.0 0.0.0.0      255.255.255.0  U         0      0        0    eth0
127.0.0.0   0.0.0.0      255.0.0.0      U         0      0        0    lo
0.0.0.0     192.168.1.1 0.0.0.0        UG         0      0        0    eth0

```

Rute na mreži se mogu proveriti tako što će se sa radne stanice WKS proveriti putanja do nekog od hostova iz mreže 192.168.5.0/24. Provera putanje se može izvršiti pomoću *traceroute* komande do IP adrese 192.168.5.1.

```

root@WKS ~# traceroute 192.168.5.1
traceroute to 192.168.5.1 (192.168.5.1), 30 hops max, 40 byte packets
 1  RT1  (192.168.1.1) 11.440 ms 10.161 ms 1.878 ms
 2  RT2  (192.168.12.2) 13.743 ms 10.891 ms 10.272 ms
 3  RT6  (192.168.26.2) 18.694 ms 15.397 ms 12.174 ms
 4  RT5  (192.168.5.1) 12.890 ms 15.996 ms 15.689 ms

```

U nastavku vežbe može se simulirati prekid na nekom od linkova, npr. na linku između rutera RT1 i RT2. Simulacija prekida linka se izvršava Linux komandom *ifconfig eth1 down* na ruteru RT2. Posle ove komande *traceroute* komanda bi dala sledeći rezultat, tj. tada bi ruta do mreže 192.168.5.0/24 bila direktno preko rutera RT2 i RT5.

```

root@WKS ~# traceroute 192.168.5.1
traceroute to 192.168.5.1 (192.168.5.1), 30 hops max, 40 byte packets
 1  RT1 (192.168.1.1) 2.417 ms 6.309 ms 2.547 ms
 2  RT5 (192.168.5.1) 4.642 ms 3.544 ms 7.178 ms

```

U toku vežbe je moguće analizirati rad OSPF protokola, prikazom poruka koje razmenjuju ruteri. Prikaz poruka se dobija komandom *tcpdump*.

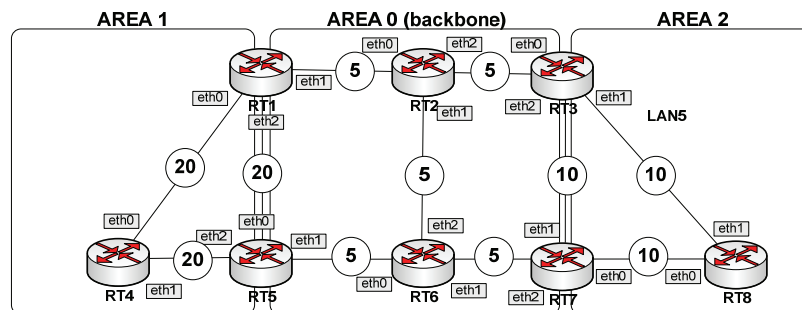
```

root@RT1 ~# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
20:01:19.820605 IP 192.168.23.1 > 192.168.23.2: OSPFv2, LS-Update (4), length: 192
20:01:20.419790 IP 192.168.26.1 > 224.0.0.5: OSPFv2, Hello (1), length: 48
20:01:20.719788 IP 192.168.15.2 > 224.0.0.5: OSPFv2, Hello (1), length: 48
20:01:22.700798 IP 192.168.67.1 > 192.168.67.2: OSPFv2, LS-Update (4), length: 192
.....
20:02:38.980931 IP 192.168.38.2 > 224.0.0.5: OSPFv2, LS-Ack (5), length: 44
20:02:39.120559 IP 192.168.23.1 > 192.168.23.2: OSPFv2, LS-Update (4), length: 120

```

### Modeliranje scenarija za OSPF protokol u mreži podeljenoj na područja

Jedna od prednosti virtualne laboratorije je relativno laka prekonfiguracija mreže. Simulacija OSPF protokola se dalje može vršiti u mreži podeljenoj na tri područja (Slika 4.15.). Ruteri RT1 i RT5 su definisani kao ABR ruteri (granični ruteri područja, eng. *Area Border Router*) za područje 1, a RT3 i RT7 kao ABR ruteri za područje 2.



Slika 4.15. Podela mreže na područja

Da bi se na ruterima u području 1 izvršila prekonfiguracija potrebno je prebaciti interfejs eth0 na ruteru RT1, eth2 na ruteru RT5 i sve interfejse na ruteru RT4 u odgovarajuće područje. To radi GNU Zebra komandom `network 192.168.14.0/24 area 1` za eth0 na RT1. Definisanje tipa područja vrši se komandama `area 1 stub [no-summary]`, za stub područja. Za rutere u području 2, potrebno je izvršiti slične prekonfiguracije. Posle izvršenih izmena, izgled tabele rutiranja na ruteru RT4 koji se nalazi u području 1 prikazan je na listingu (Listing 4.7.). IA je oznaka za OSPF *inter area route* (tj. rute prema mreži koja se nalazi u drugom području), a ABR za granični ruter područja.

```

ospfd> show ip ospf route
===== OSPF network routing table =====
N      IA      0.0.0.0/0          [21]   area: 0.0.0.1
                via 192.168.14.1, eth0
                via 192.168.45.2, eth1
N      IA      192.168.1.0/24     [30]   area: 0.0.0.1
                via 192.168.14.1, eth0
N      IA      192.168.2.0/24     [35]   area: 0.0.0.1
                via 192.168.14.1, eth0
N      IA      192.168.3.0/24     [40]   area: 0.0.0.1
                via 192.168.14.1, eth0
N      IA      192.168.4.0/24     [10]   area: 0.0.0.1
                directly attached to eth2
N      IA      192.168.5.0/24     [30]   area: 0.0.0.1
                via 192.168.45.2, eth1
N      IA      192.168.6.0/24     [35]   area: 0.0.0.1
                via 192.168.45.2, eth1

```

```

N      IA      192.168.7.0/24 [40]   area: 0.0.0.1
                        via 192.168.45.2, eth1
N      IA      192.168.8.0/24 [50]   area: 0.0.0.1
                        via 192.168.14.1, eth0
                        via 192.168.45.2, eth1
N      IA      192.168.12.0/24 [25]   area: 0.0.0.1
                        via 192.168.14.1, eth0
N      192.168.14.0/24 [20]   area: 0.0.0.1
                        directly attached to eth0
N      IA      192.168.15.0/24 [40]   area: 0.0.0.1
                        via 192.168.14.1, eth0
                        via 192.168.45.2, eth1
N      IA      192.168.23.0/24 [30]   area: 0.0.0.1
                        via 192.168.14.1, eth0
N      IA      192.168.26.0/24 [30]   area: 0.0.0.1
                        via 192.168.14.1, eth0
                        via 192.168.45.2, eth1
N      IA      192.168.37.0/24 [40]   area: 0.0.0.1
                        via 192.168.14.1, eth0
                        via 192.168.45.2, eth1
N      IA      192.168.38.0/24 [40]   area: 0.0.0.1
                        via 192.168.14.1, eth0
N      192.168.45.0/24 [20]   area: 0.0.0.1
                        directly attached to eth1
N      IA      192.168.56.0/24 [25]   area: 0.0.0.1
                        via 192.168.45.2, eth1
N      IA      192.168.67.0/24 [30]   area: 0.0.0.1
                        via 192.168.45.2, eth1
N      IA      192.168.78.0/24 [40]   area: 0.0.0.1
                        via 192.168.45.2, eth1
===== OSPF router routing table =====
R      192.168.15.1 [20]   area: 0.0.0.1, ABR
                        via 192.168.14.1, eth0
R      192.168.56.1 [20]   area: 0.0.0.1, ABR
                        via 192.168.45.2, eth1
===== OSPF external routing table =====

```

Listing 4.7. Tabela rutiranja na ruteru RT4 dobijena pomoću komande show ip ospf route u mreži podjeljenoj na područja.

### 4.3. Modeliranje ostalih scenarija za virtuelnu mrežnu laboratoriju

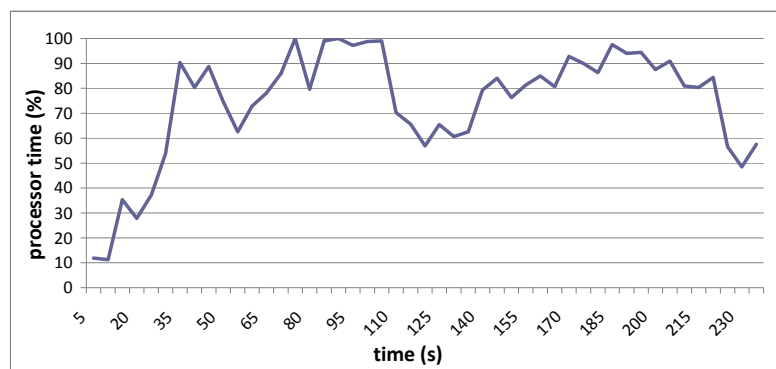
Da bi se virtuelna mrežna laboratorija mogla koristiti u nastavi bilo je potrebno modelirati više laboratorijskih vežbi. U početku je kreirano 7 laboratorijskih vežbi. Sve vežbe su kreirane po uzoru na scenarija korišćena u prikazanim vežbama za RIP i OSPF protokole. Broj potrebnih virtuelnih mašina/rutera i njihova konfiguracija zavise od teme laboratorijske vežbe. Najveći broj virtuelnih mašina u jednom scenariju iznosi 7. Pregled mrežnih scenarija i broj virtuelnih mašina za svaki scenario dat je u tabeli 4.1. Na primer, mreža koja se koristi u vežbi 5 ima 5 rutera (RT1, RT2, RT3, RT4 i RT5), 1 radnu stanicu (WS1) i 1 server (SRV1) po scenariju koji je predstavljen [DO06a].

TABELA 4.1 - STRUKTURA VEŽBI U VIRTUELNOJ MREŽNOJ LABORATORIJI

LAB	TEMA	VIRTUELNE MAŠINE	SOFTVER
1	IP adresiranje	RT1	ifconfig, netconfig komande
2	Statičko rutiranje	WS1, WS2, RT1, RT2, SRV1	route komanda
3	RIP protokol	WS1, RT1, RT2, RT3, RT4, SRV1	zebra routing softver
4	OSPF protokol, jedno područje	WS1, RT1, RT2, RT3, RT4, RT5, SRV1	zebra routing softver
5	OSPF protokol, stub područja	WS1, RT1, RT2, RT3, RT4, RT5, SRV1	zebra routing softver
6	Veb servis FTP servis	WS1, RT1, RT2, SRV1	apache veb server, proftpd server
7	firewall koncept	WS1, WS2 RT1, RT2, SRV1	iptables komanda

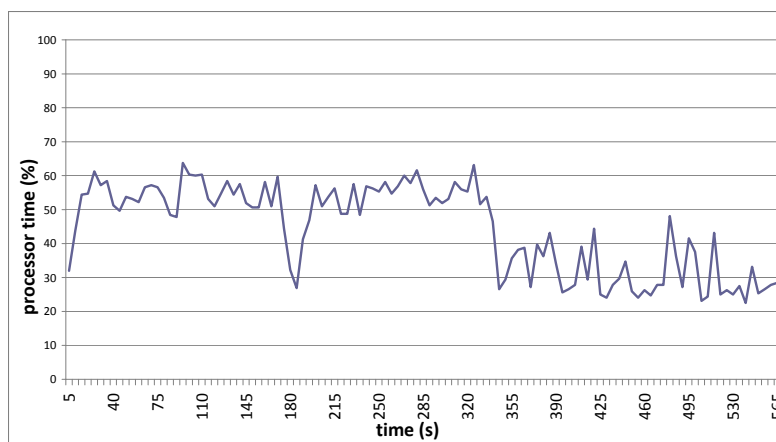
#### 4.4 Testiranje performansi virtuelne laboratorije

Virtuelna mrežna laboratorija sa lokalnim pristupom je testirana na Windows XP radnoj stanici sa Intel Celeron 2,66GHz CPU i 512MB RAM-a i Windows XP radnoj stanici sa Intel IV 1,5GHz CPU i 512MB RAM-a. Količina slobodnog prostora na disku zavisila je od broja virtuelnih mašina. Za svaku virtuelnu mašinu dovoljno je bilo odvojiti 32 MB RAM memorije. Svaka virtuelna mašina / ruter sa Linux operativnim sistemom konfigurisana je sa 4 virtuelna mrežna interfejsa (to je ujedno i maksimalan broj interfejsa dozvoljen od strane MS Virtual PC 2004 softvera). Iskorišćenost hardverskih resursa (procesorsko vreme) za vreme startovanja mašina iz laboratorije prikazano je na slici 4.16.



Slika 4.16. Procesorsko vreme za vreme startovanja virtuelne mrežne laboratorije

Iskorišćenost hardverskih resursa u toku vežbe za RIPv2 protokol prikazan je na slici 4.17.



Slika 4.17. Procesorsko vreme za vreme rada u RIPv2 scenariju

Prikaz iskorišćenja hardverskih resursa prikazan je u tabeli 4.2 za svaku vežbu posebno. Procesorsko vreme je variralo od 45% do 100% zauzeća u procesu startovanja laboratorije i od 22% do 65% zauzeća za vreme izvođenja laboratorijskih vežbi. Vreme potrebno za startovanje 7 virtuelnih mašina bilo je oko 5 minuta sa sličnim procesorskim iskorišćenjem. Scenario označen sa 7 XP sadrži u sebi radnu stanicu sa operativnim sistemom Microsoft Windows XP. Windows XP virtuelnoj mašini potrebno je mnogo više prostora na disku (2,58 GB) u poređenju sa 400 MB koliko je potrebno za Trustix Linux virtuelnu mašinu. Takođe, upotreba fizičke memorije je znatno veća, minimum 128 MB RAM-a. Rad sa XP virtuelnom radnom stanicom bio je znatno sporiji, ali ipak moguć.

Test je urađen upotrebom Microsoft ® Management Console © – System Monitor and

Performance Logs and Alerts softvera.

Test je pokazao da je glavno hardversko ograničenje laboratorije količina operativne memorije računara na kome se ona nalazi. Tuxix Linux virtuelne mašine zahtevaju 32MB RAM virtuelne memorije i ta količina je potrebna za svaku virtuelnu mašinu posebno. Virtual PC proces takođe zahteva određenu količinu fizičke memorije, a ona varira od 6MB kada se radi sa jednom, preko 11 Mb za rad sa šest, do oko 25Mb potrebnih za rad sa 8 virtuelnih mašina. Maksimalan broj virtuelnih mašina na računaru sa 512 Mb je 8 [OD06].

TABELA 4.2

TEST PERFORMANSI RAČUNARA

C	NVM	ST (min)	PU (%)	PMU (MB)	HDU (GB)
C1	1	~1	7-38	329	0.39
C1	5	~2	15-73	199	1.92
C1	6	~4	16-76	168	2.31
C1	7	~5	31-88	137	2.70
C1	7 XP	~6	47-100	91	5.33
C2	1	~1	7-48	331	0.39
C2	5	~3	56-95	225	1.92
C2	6	~5	58-97	194	2.31
C2	7	~6	72-100	169	2.70
C2	7 XP	~7	79-100	68	5.33

C = Konfiguracija računara; NVM = Broj virtuelnih mašina; ST = Vreme podizanja sistema (min); PU = upotreba procesora za vreme vežbe (u procentima); PMU = slobodna fizička memorija računara (MB); HDU = upotreba hard diska (GB).

C1 = Windows XP radna stanica sa Intel Celeron 2,66GHz CPU i 512MB RAM memorije; C2 = Windows XP radna stanica sa Intel IV 1,5GHz CPU i 512MB RAM memorije.

#### 4.5 Anketiranje studenata o radu sa virtuelnom laboratorijom

Posle primene laboratorijskih vežbi u nastavi sprovedena je anketa među studentima koji su koristili laboratoriju. Ukupno je anketirano 79 studenata.

Studenti su trebali da ocene upotrebljivost softvera za virtuelizaciju Microsoft Virtual PC 2004 u toku nastave i laboratorijskih vežbi iz predmeta Računarske mreže. Ocenjivanje je vršeno sa ocenama na skali od 1 do 5, gde je 1 najlošija, a 5 najbolja ocena (pitanje Q1 - Tabela 4.3.).

Sledeće pitanje u anketi se odnosilo na složenost upotrebe Virtual PC softvera u laboratorijskim vežbama. Postojala su dva odgovora:

- Da – za slučaj kada student smatra da je upotreba ovog softvera previše kompleksna za laboratorijsku upotrebu i
- Ne – za one studente koji misle da potreba ovog softvera nije složena.

Na kraju, studenti su anketirani da odaberu pogodniji softver za upotrebu u nastavi predmeta Računarske mreže. Postojale su tri opcije:

- Microsoft Virtual PC 2004,
- softver za simulaciju računarskih mreža OPNet IT Guru Academic Edition i

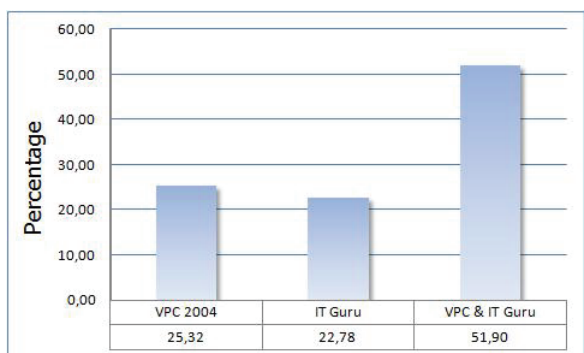
- oba softvera podjednako.

Studenti su bili upoznati sa radom oba softvera budući da je i OPNet IT Guru Academic Edition korišćen u nastavi u okviru istog predmeta. Takođe, studenti su sa IT Guru softverom imali više iskustva nego u radu sa Microsoft Virtual PC 2004.

Velika većina studenata (83,54%) smatra da upotrebljeni metod nije previše kompleksan za upotrebu u nastavi. To je bio iznenađujući rezultat, zato što se radilo o laboratoriji koja nije bila integrisana kao jedinstven sistem i studenti su morali da manipulišu sa svakom virtuelnom mašinom posebno.

Na kraju, rezultat ankete o pogodnijem alatu za upotrebu u laboratoriji pokazao je da čak 51.90% studenata nije moglo da odabere bolji alat i oba alata su smatrala podjednako dobrim. Ostalih 25.32% studenata je odabralo Virtual PC, a 22.78% IT Guru Academic Edition kao pogodniji softver za nastavu. (Slika 4.18).

Zadnji rezultat je pokazao da je Virtual PC neznatno popularniji softver od IT Guru Academic Edition-a, iako su studenti proveli manje vremena u radu sa njim.



Slika 4.18. Rezultati ankete o mogućnosti upotrebe pojedinih softvera u nastavi

U nastavku ankete postavljena su još dva pitanja. Studenti su trebali da ocene koliko je pristup u učenju mrežnih koncepata uz pomoć softvera za virtuelizaciju koristan po njihovom mišljenju (pitanje Q2). Drugo, studenti su anketirani da ocene koliko ovakav način učenja može da se koristi za pripremu za rad sa stvarnim sistemima po njihovom mišljenju (pitanje Q3). Ocene su bile od 1 do 5, sa ocenom 5 kao najvećom. Rezultati su dati u tabeli (Tabela 4.3.).

Studenti su bili veoma zadovoljni sa softverom za virtuelizaciju Virtual PC 2004. Prosečna ocena je bila 4.19, što je značilo da su studenti brzo prihvatili Virtual PC softver kao alat za učenje mrežnih koncepata. Skoro 40% studenata (njih 31) dalo je najvišu ocenu 5, a 43% ocenu 4 (34).

Studenti su bili manje uvereni da im ovakav pristup može biti koristan u učenju mrežnih koncepata, budući da je prosečna ocena 3.95. Takođe, još manje su bili uvereni da ih ova laboratorija može pripremiti za rad sa realnim sistemima (prosečna ocena 3.56), ali je u oba slučaja prosečna ocena i dalje visoka.

TABELA 4.3.  
REZULTATI UPITNIKA

Q	AG	N5	N4	N3	N2	N1
Q1	4.19	31	34	12	2	0
Q2	3.95	22	35	18	4	0
Q3	3.56	12	34	23	6	4



Q – Pitanje; AG – Srednja ocena; N5 – broj dobijenih ocena 5; N4 - broj dobijenih ocena 4; N3 - broj dobijenih ocena 3; N2 - broj dobijenih ocena 2; N1 - broj dobijenih ocena 1.

Q1 – Ocena studenta o korisnosti upotrebe Virtual PC 2004 softvera u nastavi; Q2 – Ocena studenta o mogućnosti da se kroz upotrebu Virtual PC softvera nauče mrežni koncepti; Q3 – Ocena studenta o mogućnosti da će ih upotreba softvera Virtual PC pripremiti za rad sa realnim mrežnim sistemima?

#### **4.6. Analiza rada virtuelne mrežne laboratorije sa lokalnim pristupom**

U ovom poglavlju prikazano je modeliranje i realizacija virtuelne laboratorije za računarske mreže sa lokalnim pristupom i njena upotreba u nastavi za učenje rutiranja i drugih mrežnih koncepata. Primeri za simulaciju rada RIP i OSPF protokola u takvoj laboratoriji su takođe prikazani. Virtuelna laboratorija za računarske mreže pokazala se kao efikasno okruženje upotrebljivo u nastavi zbog svoje izuzetno niske cene, velike fleksibilnosti i dostupnosti velikom broju studenata, kako u terminima predviđenim za nastavu, tako i van nastavnim terminima i kod kuće.

Da bi funkcionisala, laboratorija se mora instalirati na svakom računaru posebno. U praksi to se realizovalo tako što je administrator/predavač instalirao samo Virtual PC na svakom računaru. Studenti su, prema uputstvima pre svake vežbe, sa servera preuzimali (kopirali) i instalirali potrebne virtuelne mašine. Taj proces je bio deo pripreme, ali je trajao srazmerno kratko i nije bitno uticao na zastoje u radu. Najduže vreme bilo je potrebno za samo preuzimanje datoteka koje sadrže virtuelne mašine sa servera (eng. *download*), jer je prosečna veličina datoteka za prenos bila oko 2,7 Gb (Tabela 4.2.). Drugi način distribucije ove laboratorije je bio putem DVD diskova sa narezanim datotekama virtuelnih mašina. Ni takav način nije znatnije pojednostavio proces pripreme laboratorije za vežbu.

Osnovni nedostatak ovakve primene, osim navedenog sporog procesa pokretanja laboratorije, je njena administracija. Sa obzirom na to da se laboratorija instalira na svakom računaru posebno, njeno održavanje i unapređenje je veoma komplikovano. Dodatno ograničenje, predstavljala je limitiranost u scenarijima laboratorije. Uzrok tome su bili mali resursi radnih stanica koji su dozvoljavali rad samo ograničenom broju virtuelnih mašina. Prvi limitirajući resurs bila je operativna memorija računara, a drugi je bio prostor na disku.

Zbog svih ovih limitirajućih faktora započet je rad na modeliranju virtuelne laboratorije sa udaljenim pristupom [Sto03, YH04, Slo02, PRB01, Jov04, LJD08] uz eventualnu upotrebu veb tehnologije. Takav koncept, omogućio bi upotrebu znatno većih resursa, a samim tim i znatno kompleksnijih mreža. To bi se postiglo smeštajem virtuelnih mašina na server sa većim kapacitetima, prvenstveno u pogledu operativne memorije i prostora na hard disku. Tako bi se omogućila upotreba raznovrsnijih scenarija, pristup većem broju studenata u isto vreme, a upotrebljivost laboratorije bi se mogla proširiti na sve prostorije Fakulteta i na pristup preko Interneta. Hostovanje laboratorije na posvećenom serveru poboljšao bi i olakšalo proces njene administracije i rekonfiguracije.

U toku planiranja ovakve laboratorije bilo je potrebno sagledati raspoloživu tehnologiju koja omogućava udaljeni pristup. Nakon odabira i implementacije tehnologije, bilo je potrebno kreirati i podesna mrežna scenarija za ovakvo novo okruženje. Migracija na novi onlajn sistem, stvaranje modela i njegova arhitektura prikazani su u sledećem poglavlju.

## 5. Modeliranje i realizacija virtuelne mrežne laboratorije sa pristupom na daljinu

### 5.1. Modeliranje virtuelne mrežne laboratorije sa pristupom na daljinu

Virtuelna mrežna laboratorija sa daljinskim pristupom nazvana VNLab (*Virtual Network Laboratory*) nastala je kao rezultat istraživanja u pravcu migracije postojećeg sistema na jedno drugo, pouzdanije i fleksibilnije okruženje koje će biti efikasnije za rad i upotrebu u nastavi za učenje mrežnih koncepata i TCP/IP mreža. VNLab je onlajn edukativni sistem namenjen za obrazovanje inženjera računarstva, informacionih tehnologija i informatike koje distribuira materijal i sredstva za učenje posredstvom mrežnih tehnologija. VNLab je ujedno i emulirana računarska mreža koja se sastoji od virtuelnih mašina. Pristup ovom sistemu omogućen je kroz TCP/IP mrežu, tj. kroz postojeću infrastrukturu ustanove. U ovom poglavlju je opisano kreiranje modela koji je omogućio realizaciju virtuelne mrežne laboratorije VNLab.

VNLab je hardversko-softverski model virtuelne mrežne laboratorije koji je struktuiran na sledeći način [DSO11]. Softverske komponente modela sastoje se od nekoliko elemenata koje omogućavaju funkcionisanje sistema. U hardverske komponente modela, spadaju server, radne stanice i mrežna infrastruktura koja omogućava upotrebu laboratorije i pristup VNLab serveru.

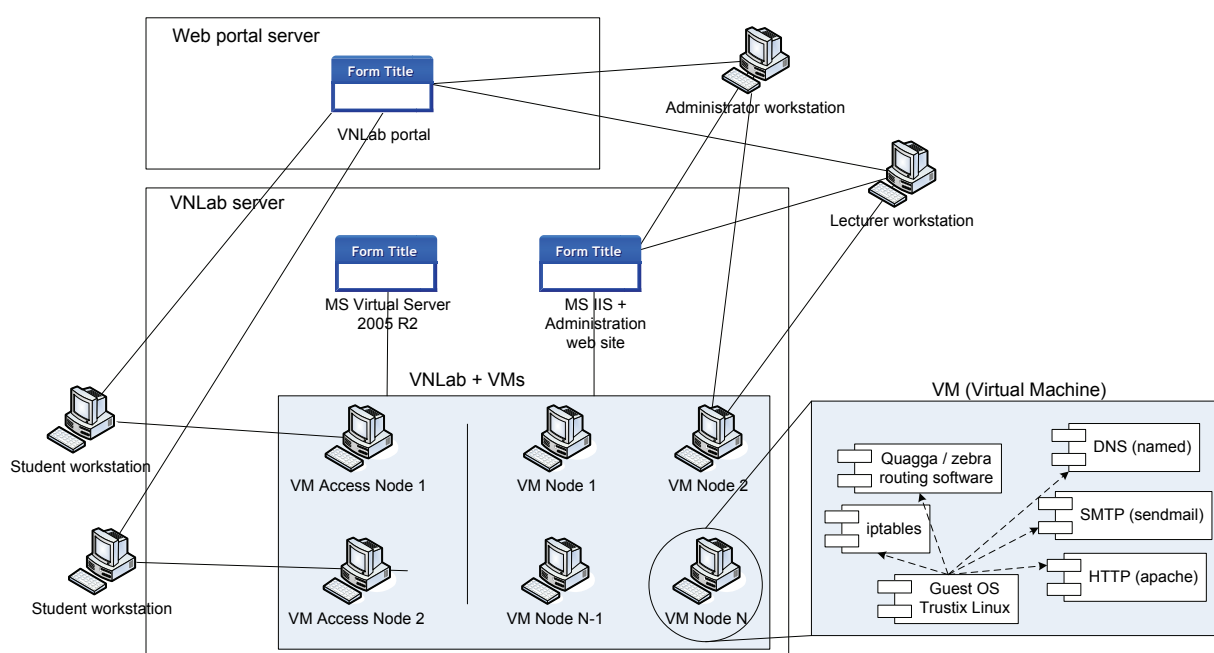
VNLab je baziran na Microsoft Virtual Server 2005 R2 [LC07] softveru za virtuelizaciju. To je ujedno i glavna razlika u poređenju sa laboratorijom sa lokalnim pristupom baziranom na softveru Microsoft Virtual PC 2004. VNLab se hostuje na računaru sa host operativnim sistemom MS Windows 2003 i veb serverom Microsoft IIS 6.0 (Internet Information Server) [Tul03].

Arhitektura VNLab modela prikazana je na slici 5.1. Microsoft Virtual Server 2005 R2 sa IIS 6.0 veb serverom čini osnovu za stvaranje virtuelne mreže i omogućava efikasnu infrastrukturu za pokretanje, rad i administraciju virtuelne laboratorije. Virtual Server 2005 R2, pored kreiranja i pokretanja virtuelnih mašina, omogućava i emulaciju virtuelne mreže podrškom za virtuelne mrežne konekcije između virtuelnih mašina. Ovaj postupak je opisan detaljnije u sekciji koja je posvećena konfiguraciji i administraciji VNLab-a.

Emulirana mreža se sastoji od velikog broja virtuelnih mašina. Broj od 40 virtuelnih mašina je dovoljan za kreiranje mrežnih scenarija i za izvođenje laboratorijskih vežbi. Virtuelne mašine emuliraju hardver i softver fizičkih računara zajedno sa njihovim mrežnim interfejsima (mrežnim karticama). Svaka virtuelna mašina može da emulira do 4 mrežne kartice (eng. *Network Interface Card* - NIC), jer je to maksimalan broj emuliranih virtuelnih mrežnih kartica koje podržava Virtual Server 2005.

Emulirana mreža se sastoji od dva tipa virtuelnih mašina. Prvi tip virtuelnih mašina ima ulogu računara i rutera koji čine jegro mreže. To su prethodno konfigurisani čvorovi koji omogućavaju postojanje funkcionalne mreže koja u potpunosti odgovara realnim sistemima. Ti čvorovi su prikazani na slici 5.1. kao VM Node 1, VM Node 2, itd. Emulirani čvorovi konfigurisani su tako da omoguće stvaranje jezgra virtuelne mreže koja pruža mogućnost studentima da primenjuju stečenja znanja i unapređuju svoje veštine praktičnim radom u sistemima koji su slični realnim okruženjima. Nivo realnosti mrežnih scenarija i funkcionisanje svih studentskih vežbi direktno zavisi od konfiguracije ovih čvorova. Ako ti čvorovi nisu dobro konfigurisani, ni okruženje za učenje mrežnog koncepta (mrežni scenario) neće biti dobro konfigurisano i prilagođeno potrebama nastave.

Drugi tip računara su oni koji imaju ulogu mrežnih čvorova koje je potrebno konfigurisati, a to je deo posla koji obavljaju studenti u cilju praktične primene naučenog i sticanja novih veština.



Slika 5.1. Arhitektura modela virtuelne mrežne laboratorije sa pristupom na daljinu

Ti čvorovi su prikazani na slici 5.1. kao VM AccessNode1, VM AccessNode2 itd., i oni se dodeljuju studentima koji ih koriste za vreme nastave i pomoću kojih uče mrežne koncepte konfigurisanjem virtuelnih mašina sa ciljem da svaku virtuelnu mašinu učine funkcionalnim delom postojeće mreže. Jezgro mreže se mora konfigurisati tako da se svim studentima omogući ravnopravan pristup i nivo funkcionalnosti za vreme upotrebe pristupnih čvorova (VM AccessNode1, VM AccessNode2 itd.).

Udaljeni pristup na pristupne čvorove je omogućen upotrebom Microsoft software Virtual Machine Remote Console (VMRC), softvera koji ide u paketu sa Virtual Server 2005 R2. Ovaj softver koristi VMRC protokol za udaljeni pristup postojećoj virtuelnoj mašini.

Konfiguracija svake virtuelne mašine je slična. Kao i u slučaju koji je opisan u prethodnom poglavlju, instaliran je Trustix Linux 2.2 kao Guest OS, zajedno sa drugim neophodnim softverom za scenarija koja se koriste u nastavi. Struktura instaliranog softvera je definisana multifunkcionalom karakteristikom svake virtuelne mašine, tj. instalirani softver mora da podrži sve potrebne funkcije za odgovarajući scenario. Ako je potrebno da virtuelna mašina ima ulogu veb ili mail servera, softver potreban za to treba da je instaliran. Ako virtuelna mašina treba da ima ulogu OSPF rutera, odgovarajući ruting softver je takođe potreban. Svaka virtuelna mašina može da se ponaša kao ruter, server ili radna stanica u zavisnosti od toga kakvu ulogu ima u odgovarajućem scenariju.

Da bi se postigao takav nivo fleksibilnosti potrebno je da su na svaku virtuelnu mašinu instalirani serverski softveri za multiprotokol rutiranje, fajervol, veb, mail, FTP i DNS. Veliki broj open-source softvera može se koristiti u okviru VNetLab-a: *apache* kao veb server [LL02], *proftpd* kao ftp server, *iptables* za fajervol [Pur04], *sendmail* kao mail server [Cos02], GNU Zebra kao multiprotokol softver za rutiranje itd.

Svi navedeni softveri su se koristili u staroj verziji virtuelne mrežne laboratorije i zbog pozitivnog iskustva se koriste i u novom okruženju. Do sada, u okviru VNetLab-a, nije bilo potrebe

za promenom softvera koji se koristi. Iako taj softver postepeno zastareva, to ne utiče na cilj VNLab-a, a to je učenje studenata osnovnim mrežnim konceptima.

Ako se u nekom trenutku ukaže potreba za upotrebom novijeg softvera, bilo koje drugo open-source rešenje se može koristiti kao i druge verzije operativnih sistema.

Uniformna softverska konfiguracija virtuelnih mašina nije neophodna, ali je poželjna zbog inicijalne instalacije laboratorije. Budući da su diskovi svih instaliranih virtuelnih mašina organizovani kao jedna imidž datoteka, moguće je instalirati sav potreban softver na prvoj virtuelnoj mašini. Posle toga, može se napraviti druga kopija imidž datoteke odgovarajuće virtuelne mašine i samo preimenovati tako da odgovara imenu novokreirane virtuelne mašine. Na taj način se samo kopira celokupan disk i dodeljuje novoj mašini, onoliko puta koliko je to potrebno da bi se kreirali svi potrebni računari u scenariju. Tako se može kreirati veliki broj čvorova u VNLab-u u veoma kratkom periodu.

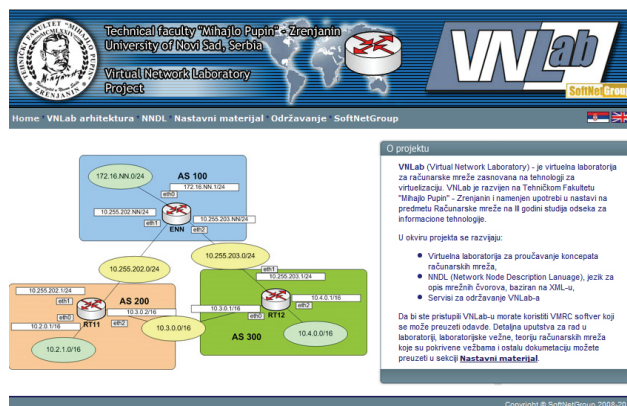
Veoma važan faktor kod modeliranja i realizacije ove virtuelne laboratorije je nizak nivo hardverskih performansi koji je potreban za pokretanje 40 virtuelnih mašina, koliko je potrebno za najveći broj mrežnih scenarija. Sve virtuelne mašine treba da se startuju i rade u približno istom vremenu. Njihovoj broj zavisi od scenarija, ali i broja studenata koji ih koriste.

## 5.2. VNLab Portal

Važna softverska komponenta modela je VNLab veb portal (Slika 5.2). Portal je dizajniran da olakša studentima korišćenje VNLab okruženja. Portal pruža studentima pristup VNLab materijalima kao što je softver za udaljeni pristup virtuelnim mašinama, uputstva za rad sa laboratorijom, laboratorijske vežbe i prateći materijali [SD10].

Osnovna struktura portala je sledeća:

- Home page – ulazak u VNLab okruženje,
- Opis VNLab arhitekture,
- Network Node Description Language (NNDL) – sekcija za opis NNDL jezika
- Laboratorijske vežbe, materijali i softverski alati,
- Sekcija za održavanje.



Slika 5.2. VNLab portal

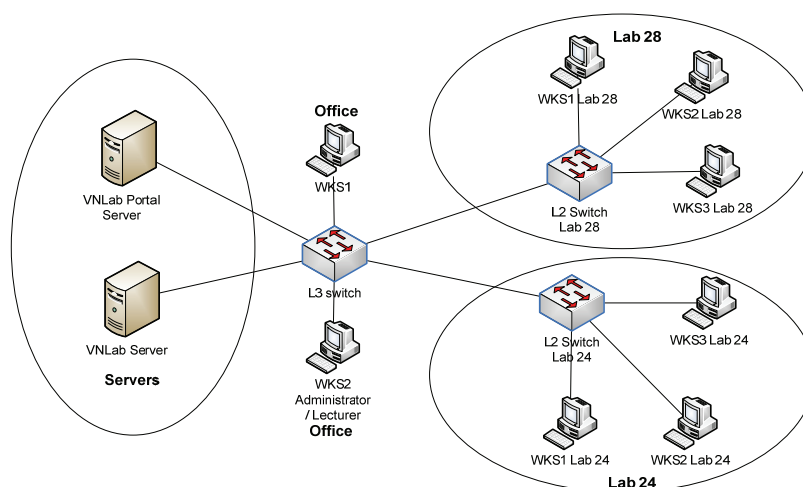
VNLab Home page predstavlja ulaznu tačku u VNLab okruženje. Ono pomaže studentima koji učestvuju u vežbama i koji pohađaju predmet Računarske mreže da shvate strukturu okruženja. Takođe, iz ovog dela portala sve ostale sekcije su dostupne. Deo portala o arhitekturi pruža tehničke informacije o VNLab okruženju i pomaže studentima da detaljnije upoznaju okruženje u kojem rade. Sledeća sekcija je posvećena NNDL-u (Network Node Description Language), XML baziranom jeziku koji je dizajniran za definisanje mrežnih scenarija u laboratorijskim vežbama [DSO08, DSD09].

Četvrta sekcija omogućava pristup važnim komponentama koje su potrebne studentima da urade zadatak predviđen laboratorijskim vežbama. Prvo, ova sekcija omogućava pristup laboratorijskim materijalima u pdf formatu. Do danas ovi materijali su dostupni samo na srpskom jeziku kao i ostatak sajta. Sledeći uputstva u laboratorijskim vežbama, studenti su u mogućnosti da konfigurišu virtuelne mašine – mrežne čvorove, tj. rutere, servere i radne stanice. U virtuelnim mrežnim scenarijima uče se određeni mrežni koncepti koji su opisani u vežbi. Struktura laboratorijskih vežbi će se detaljnije opisati u nastavku. Drugo, ovaj deo sajta omogućava pristup softveru potrebnom za upotrebu VNLab-a, kao što je VMRC (Virtual Machine Remote Control) softver koji se koristi za udaljeni pristup virtuelnim mašinama, kao i drugom pomoćnom softveru namenjen za razumevanje konfiguracije virtuelnih čvorova i scenarija, (npr. ScenarioBuilder i NodeConfigurator)[DSD09].

Na kraju, sekcija za održavanje je još u razvoju, i omogućava servis za prijavu bagova i problema sa kojima se studenti susreću u toku upotrebe laboratorije [SD10].

### 5.3. Hardverske komponente modela

Pored opisanih softverskih, model ima i hardverske komponente. Te komponente su predstavljene na slici 5.3. Na slici je prezentovan segment LAN mreže koji omogućava udaljeni pristup laboratoriji. Sistem ima dva servera: VNLab server, na kome je instaliran MS Virtual Server 2005 i IIS veb server za administraciju VNLab-a i sve virtuelne mašine; i VNLab portal server - koji služi za hosting VNLab portala. Radne stanice su povezane preko LAN mreže Fakulteta i koriste se za pristup studenata i nastavnika/administratora laboratoriji. Nastavnici, administratori i studenti koriste radne stanice koje se nalaze u dve fizičke laboratorije (Lab 28 i Lab 24). VNLab server je postojeći računar, koji je u okviru fakultetske mreže imao svoju funkciju. Računar je Dell™ PowerEdge™ 800 server sledeće konfiguracije: CPU Intel Pentium 4 3,2 GHz, 1GB RAM-a, 150GB HDD sa operativnim sistemom Microsoft Windows Server 2003.

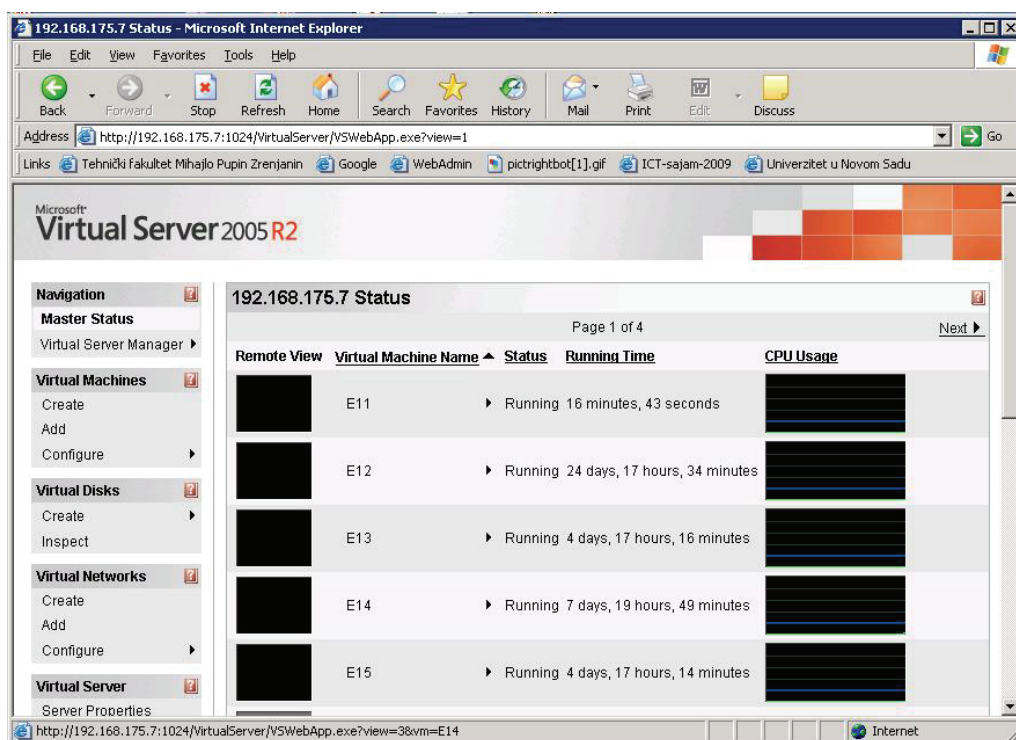


Slika 5.3. Hardverske komponente VNLab-a

## 5.4. Konfiguracija i administracija VNLab-a

Dizajn, konfiguracija i administracija VNLab-a podržane su u okviru MS Virtual Server 2005 R2 softvera i njegovih opcija. Pristup stranicama za administraciju koje se nalaze na VNLab serveru moguć je upotrebom veb brausera navođenjem domenskog imena servera i pristupom na port 1024, npr. <http://budimir:1024>. Stranica za administraciju (Slika 5.4.) omogućava veliki broj komandi klasifikovanih u pet grupa.

*Master status* komande omogućavaju monitoring virtuelnih mašina i osnovne operacije nad njima. Upotrebom ovih opcija VNLab administrator može da startuje i obori određenu virtuelnu mašinu. Dostupna je opcija za brz pregled stanja virtuelne mašine. Ekran svake mašine prikazan je na veb stranici kao minimizirani prozor sa trenutnim sadržajem. Administrator može da pristupi i da se loguje na bilo koju virtuelnu mašinu klikom na *preview* prozor.



Slika 5.4. Microsoft Virtual Server 2005 R2 stranica za administraciju

Sledeća grupa komandi su komande povezane sa kreiranjem i konfiguracijom virtuelnih mašina. Kreiranje virtuelnih mašina obuhvata sledeće akcije:

- dodelu imena
- dodelu količine virtuelne RAM memorije
- podešavanje virtuelnog hard diska
- podešavanje virtuelne mrežne kartice

Prilikom konfiguracije virtuelnih mašina parametri se mogu menjati u toku kreiranja, ali i tokom korišćenja virtuelne mašine. Dodatna podešavanja hardvera su omogućena za COM ili LPT portove, dodavanje novih mrežnih interfejsa (do četiri na jednu fizičku mrežnu karticu), flopi/CD/DVD podešavanja itd. Treća grupa opcija omogućava dodavanje postojećih i prethodno

konfigurisanih virtuelnih mašina. Ova opcija se često koristi u VNLab konfiguraciji. To se dešava u slučajevima kada se dodaje nova virtuelna mašina, na osnovu prethodno kreiranih.

Raspodela RAM memorije u VNLab-u se vrši tako što se svakoj virtuelnoj mašini dodeli po 16MB. Ova mala količina memorije, dovoljna je za funkcionisanje virtuelnih mašina, a predstavlja značajnu karakteristiku VNLab-a. Ukupna raspoloživa količina RAM memorije direktno utiče na maksimalan broj virtuelnih mašina u scenariju i predstavlja glavni limitirajući faktor sistema. RAM memorija je najvitalniji deo sistema i kritičniji u smislu potrošnje od prostora na disku.

Treća grupa opcija služi za kreiranje imidž datoteka za virtuelne diskove, koji se nalaze u dokumentima sa ekstenzijom *.vhd*. Slično, *virtual network* opcija služi za kreiranje ili dodavanje postojećih virtuelnih mreža u okruženje. Svaka postojeća virtuelna mreža se može asociirati sa fizičkim mrežnim adapterom. Navedena podešavanja se nalaze uskladištena u *.vnc* dokumentima. Virtuelne mašine i njihovi mrežni adapteri se mogu povezati ili raskonektovati sa mreže po potrebi. Ova komanda je veoma važna za kreiranje mrežnih scenarija, a za potrebe VNLab-a samo je jedna virtuelna mreža kreirana.

Poslednja grupa komandi je povezana sa administracijom. One sadrže:

- *server property manager*,
- *website property manager*,
- *resource allocation* i
- *event viewer*

*Event viewer* pomaže u rešavanju problema vezanih za funkcionisanje VNLab okruženja. Sve greške propraćene su odgovarajućim porukama koje se čuvaju u log datotekama. Pregledom poruka, administrator ima kontrolu nad događajima i lako može da primeti problem koji se dogodio za vreme izvođenja laboratorijskih vežbi. Pored grešaka, ovako se može kontrolisati i situacija kada se VNLab koristi neovlašćeno, npr. u slučajevima kada neko od studenata (korisnika VNLab-a) pristupa mašinama koje su dodeljene za rad drugim studentima. Sve konekcije na virtuelne mašine se mogu pratiti u log datotekama sa zapisanim IP adresama sa kojih se pristupa i vremenom konekcije.

Opcija koja dozvoljava raspodelu resursa je *resource allocation*. Ovaj koristan alat služi za optimizaciju VNLab okruženja. Upotrebom ove opcije administrator može limitirati ili rezervisati upotrebu CPU jedinice u procentima za svaku virtuelnu mašinu posebno. U sadašnjem podešavanju VNLab okruženja, CPU upotreba je ograničena na 20% po studentskim radnim stanicama (VM AccessNode1, VM AccessNode2, itd.), a na 5% po čvorovima jezgra mreže (VM Node1, VM Node2 itd.)

Sekcija *server website property* omogućava administratoru da doda, ukloni i podesi naloge za korisnika i grupu za pristup VNLab-u. Takođe, omogućava konfiguraciju i startovanje VMRC servera na predloženom portu (podrazumevana vrednost je 5900). VMRC server omogućava udaljeni pristup na virtuelne mašine posredstvom VMRC klijentskog sotvera. Administrator može da odabere rezoluciju ekrana, obezbedi upotrebu i konfiguracije SSL 3.0/TLS 1.0 enkripcije i sertifikata kao i da podesi *idle timeout*. Ostale komande se odnose na konfiguraciju skriptova (izvršnih programa komandne linije koji se mogu pokretati kada se neki događaj aktivira) i za podešavanje putanje za pretragu (*path*). Na kraju, *web site property* odnosi se na veb sajt za administraciju i njegova podešavanja.

## 5.5. Modeliranje laboratorijskih scenarija u VNLab okruženju

VNLab okruženje se koristi za kurs iz predmeta Računarske mreže na Tehničkom fakultetu „Mihajlo Pupin“ - Zrenjanin. Ova laboratorija se u opisanoj formi koristi od 2006. god. sa dobrim rezultatima. Za vreme kursa, veći broj laboratorijskih vežbi je kreiran i testiran u nastavi. VNLab se primarno koristi za rad sa TCP/IP mrežama. Lekcije koje su razvijene u okviru VNLab-a su IPv4 i IPv6 koncepti, adresiranje i statičko rutiranje. Sledeća grupa vežbi odnosi se na protokole za dinamičko rutiranje RIP, OSPF i BGP. Slede fajervol koncept, filtriranje paketa (eng. *packet filtering*), NAT (adresna translacija) i prosleđivanje portova (eng. *port forwarding*). Na kraju, VNLab pokriva i teme kao što su konfiguracija i funkcionisanje mrežnih servisa kao što su veb, ftp, mail i DNS, sa instalacijom i podešavanjem potrebnog softvera.

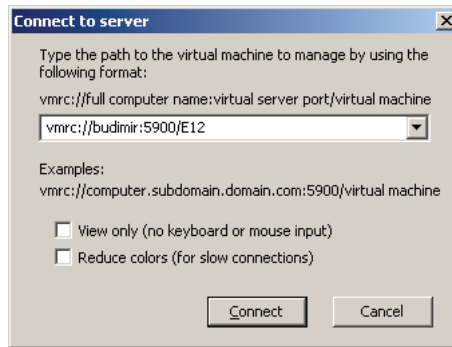
Sve navedene vežbe su realizovane kao programirane vežbe za studente. Studenti treba da se loguju na virtuelnu mrežu prema uputstvima, upotrebom VMRC softvera. Logovanje se vrši na određenu virtuelnu mašinu koja je dodeljena svakom studentu ili paru studenata (grupi). Dodela virtuelnih mašina se vrši usmeno od strane predavača. Posle dobijanja instrukcija o virtuelnoj mašini koja im je dodeljena, studenti slede instrukcije u skladu sa laboratorijskim materijalom koji se može preuzeti sa VNLab portala ili sa mrežnog diska.

Laboratorijske vežbe su struktuirane na sledeći način. Na početku vežbe, postoji kratka teorijska osnova za koncept koji se trenutno uči, tj. koji je obuhvaćen ovom vežbom. Cilj teorijskog dela je da u kratkim crtama objasni koncept koji se radi u vežbi i studentima prezentuje informacije potrebne za razumevanje vežbe. Posle uvoda, u laboratorijskom materijalu, dat je kratak opis mrežne topologije koja se koristi u vežbi, zajedno sa problemom koji student treba da reši. Studenti treba da izvrše nekoliko aktivnosti da bi mogli da reše problem. Rešenje problema vodi do učenja i razumevanja određenih koncepata. Da bi se studentim pružila detaljna uputstva, laboratorijski materijali daju prikaz konfiguracije korak po korak. Studenti dobijaju instrukcije kako da se loguju na virtuelnu mašinu (mrežni čvor), kako da konfiguriraju važne parametre potrebne za određeni mrežni scenario i kako da testiraju konfiguraciju koju su napravili.

Važan deo procesa učenja jeste vrednovanje studentskog znanja stečenog u toku vežbe. Da bi se to postiglo, na kraju vežbe je sekcija sa pitanjima i zadacima. Dva ili tri pitanja koja su povezana sa urađenom vežbom, namenjena su studentima za rešavanje. Studenti rešavaju te zadatke i odgovaraju na pitanja odmah po završetku praktičnog dela vežbe. Njihovi odgovori se proveravaju i vrednuju. Analizirani odgovori mogu da pomognu nastavniku na dva načina: (1) da dobije uvid u efikasnost sistema, (2) da analizira i unapredi teže razumljive delove vežbe ili delove na kojima studenti imaju najlošiji učinak.

Na početku vežbe, studenti se loguju na njima dodeljeni ENN router. NN predstavlja broj dodeljen studentu. Tako, ako je dodeljen broj 11, student se loguje na ruter E11. Logovanje se vrši na ruter uz pomoć VMRC klijentskog softvera koji je preuzet sa VNLab portala ili mrežnog diska. Na primer, student pristupa E11 ruteru navođenjem adrese vmrc://budimir:5900/E11 gde *budimir* predstavlja domensko ime VNLab servera, ali se može koristiti i IP adresa (Slika 5.5.).





Slika 5.5. Pristup udaljenom virtualnom ruteru

Posle uspostavljanja konekcije na ENN ruter, student mora da se prijavi za rad na VNLab serveru (Slika 5.6.) sa ličnom lozinkom (za sve studente se koristi isto korisničko ime i ista lozinka). Posle uspešne prijave na VNLab server, sledi prijava na ENN ruter sa VRMC softverom. Proces prijave na ruter ENN je proces prijave na Linux operativni sistem. Studenti koriste *root* korisnički nalog (administrator sistema) i lozinku *123456*.



Slika 5.6. Prijava na VNLab server

VNLab vežbe će biti opisane detaljnije u nastavku da bi se stekao bolji uvid u njihovu strukturu. Detaljan prikaz vežbi pomoći će da se prikaže:

- koja specifična znanja dobijaju studenti
- na koji način se to postiže
- realnost okruženja u kome studenti rade i njegova sličnost sa stvarnim sistemima i situacijama
- način provere znanja studenata

Do sada je u okviru VNLab-a kreirano 11 vežbi. Njihov prikaz dat je u tabeli 5.1.

Tabela 5.1. – VNLab vežbe

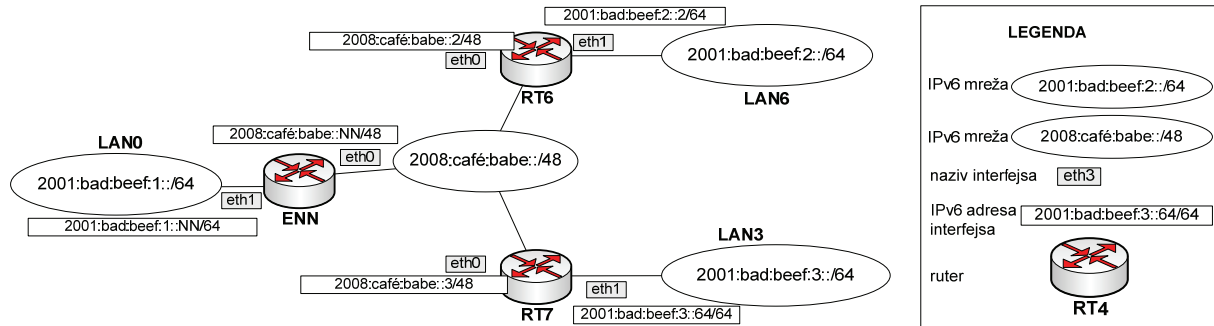
<b>Vežba</b>	<b>Naziv i vm koje se koriste</b>	<b>Koncepti koji se uče</b>	<b>Napomena</b>
VNLab01	IPv4  RT1, RT2, RT3, RT4, ENN	IPv4 struktura adrese, konfiguracija interfejsa, provera konfiguracije interfejsa, adresa mreže, adresa interfejsa,	

		statičko rutiranje	
VNLab02	IPv4 RT1,RT2,RT3,ENN	IPv4 struktura adrese, konfiguracija interfejsa, provera konfiguracije interfejsa, adresa mreže, adresa interfejsa, statičko rutiranje, tabela rutiranja, pojam gateway-a, default ruta, brisanje ruta	Dodatak A
VNLab03	IPv6 RT7,RT8,ENN	IPv6 struktura adrese, konfiguracija interfejsa, provera konfiguracije interfejsa, adresa mreže, adresa interfejsa, statičko rutiranje, IPv6/IPv4 tuneliranje	Poglavlje 5.5.
VNLab04	RIP RT1,RT2,RT3,RT4,RT5, WS1,ENN	Dinamičko rutiranje, RIP i distance vector protokoli, GNU Zebra, konfiguracija RIP protokola, simulacija prekida linka	Dodatak B
VNLab05	OSPF – jedno područje RT1,RT2,RT3,RT4,RT5,WS1	Dinamičko rutiranje, OSPF i shortest path protokoli, konfiguracija OSPF protokola, područja, cene linka simulacija prekida linka	Poglavlje 5.6.
VNLab06	OSPF – više područja RT1,RT2,RT3,RT4,RT5,WS1	konfiguracija OSPF protokola, područja, cene linka, tipovi područja NSSA, SA i TSA simulacija prekida linka	Poglavlje 5.7.
VNLab07	BGP-4 RT11,RT12	Dinamičko rutiranje, autonomni sistemi, BGP-4 protokol, konfiguracija BGP-4 protokola, susedi, simulacija prekida linka	Poglavlje 5.8.
VNLab08	Firewall ENN	koncepti firewall-a, filtriranje paketa, NAT, prosleđivanje portova, komanda IPTABLES, unos komandi	Poglavlje 5.9.
VNLab09	Firewall – II ENN	unos komandi, zaštita sistema	
VNLab10	DNS ENN	Upoznavanje sa funkcionisanjem DNS servisa, pregled konfiguracionih datoteka, izmena konfiguracije prema specifikaciji, kreiranje primarnog DNS servera	Dodatak C
VNLab11	Web, mail ENN	konfiguracione datoteke, izmena konfiguracionih datoteka, startovanje serverskih programa	

## 5.6. VNLab vežba - IPv6

Vežba koja će biti opisana u ovom poglavlju odnosi se na IPv6 protokol. Na slici 5.7. je prikazana topologija mreže koja se koristi u tom scenariju. U vežbi je potrebno izvršiti konfigurisanje rutera dodelom odgovarajućih IP adresa njegovim interfejsima i unosom potrebnih statičkih

ruta. Pre početka vežbe u materijalu koji se daje studentima nalazi se i instrukcija kako će se te vežbe izvršiti. Dat je kratak uvod u IPv6 protokol, razlog za pojavu takve vrste adresiranja, kao i format IPv6 adresa. Osnovni koncepti statičkog rutiranja nisu prikazani jer se statičko rutiranje vrši na isti način kao u vežbi sa IPv4 statičkm rutiranjem, koja se radi pre IPv6 vežbe. Ta vežba neće biti prikazana u ovom poglavlju zbog sličnosti, ali će biti u priložena u celosti u Dodatku A.



Slika 5.7. Mrežni scenario za IPv6 vežbu

Pre početka rada potrebno je učitati modul za podršku IPv6 protokolu. Na Trustix 2.2. sistemima koristi se kernel v2.4.31 koji podržava IPv6 protokol, ali samo kao spoljašnji modul. Učitavanje modula vrši se komandom:

```
root@RT1 ~# modprobe ipv6
```

Sledeći korak je dodela odgovarajućih IP adresa mrežnim interfejsima. IPv6 adrese na Linux sistemima dodeljuju se komandom `ip -6 addr add <ipv6address>/ <prefixlength> dev <interface>`, npr:

```
root@RT1 ~# ip -6 addr add 2008:café:babe::NN/48 dev eth0
root@RT1 ~# ip -6 addr add 2001:bad:beef:1::NN/64 dev eth1
```

Komandama `ifconfig eth0` i `ifconfig eth1` može da se proveriti da li su IP adrese ispravno dodeljene. Komanda `ifconfig` zajedno sa nazivom interfejsa (npr. `eth0`) daje prikaz podataka o interfejsu kao što su IP adresa (inet addr), MAC adresa (Hwaddr), mrežna maska (Mask), broadcast adresa (Bcast) i statistiku ukupno prenešenih podataka.

```
[root@zpupin root]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:11:09:8D:28:3D
          inet addr:172.16.1.30  Bcast:172.16.1.255  Mask:255.255.255.0
          inet6 addr: 2008:café:babe::1/48  Scope:Global
          inet6 addr: fe80::211:9ff:fe8d:283d/64  Scope:Link
```

```
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:52244677 errors:0 dropped:0 overruns:0 frame:0
TX packets:41424761 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:704605347 (671.9 Mb)  TX bytes:89394363 (85.2 Mb)
Interrupt:11 Base address:0x8c00
```

```
[root@zpupin root]# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:02:44:8F:6F:EC
          inet addr:172.16.2.30  Bcast:172.16.2.255  Mask:255.255.255.0
          inet6 addr: 2001:bad:beef:1::111/64  Scope:Global
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:37793481 errors:1 dropped:2 overruns:1 frame:0
TX packets:55776157 errors:0 dropped:0 overruns:20 carrier:0
```

```
collisions:0 txqueuelen:1000
RX bytes:4231953583 (4035.9 Mb) TX bytes:356804493 (340.2 Mb)
Interrupt:10 Base address:0xb800
```

Prikaz podešenih IP adresa se može dobiti i komandom:

```
root@RT1 ~# ip -6 addr show
```

Prikaz parametara pojedinačnog interfejsa se dobija komandom:

```
root@RT1 ~# ip -6 addr show dev eth0
```

Nakon unosa odgovarajućih IPv6 adresa, može se proveriti tabela rutiranja. To se radi komandom *ip -6 route show*. Zbog toga što nisu unešene statičke rute, tabela rutiranja na ruteru ENN ima samo rute koje se postavljaju automatski ka mrežama koje su direktno povezane na njega, tj. ka mrežama kojima pripadaju njegova dva interefejsa.

```
root@RT1 ~# ip -6 route show
2008:cafe:babe::/48 dev eth0 metric 256 mtu 15000 advmss 1440
2001:bad:beef:1::/64 dev eth1 metric 256 mtu 15000 advmss 1440
fe80::/64 dev eth0 metric 256 mtu 1500 advmss 1440
fe80::/64 dev eth1 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth0 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth1 metric 256 mtu 1500 advmss 1440
default dev eth0 proto kernel metric 256 mtu 1500 advmss 1440
default dev eth1 proto kernel metric 256 mtu 1500 advmss 1440
unreachable default dev lo proto none metric -1 error -101
```

Budući da ne postoji ruta ka mreži 2001:bad:beef:3::/64, rutu je potrebno uneti. Rute se unose komandom *ip -6 route add <ipv6network>/<prefixlength> via <ipv6address>*, gde je <ipv6address> adresa gejtveja. U ovom slučaju je to 2008:cafe:babe::3 ili adresa eth0 interfejsa na ruteru RT7 (Slika 5.7.).

```
root@RT1 ~# ip -6 route add 2001:bad:beef:3::/64 via 2008:cafe:babe::3
```

Nova tabela rutiranja sada treba da izgleda ovako:

```
root@RT1 ~# ip -6 route show
2008:cafe:babe::/48 dev eth0 metric 256 mtu 15000 advmss 1440
2001:bad:beef:1::/64 dev eth1 metric 256 mtu 15000 advmss 1440
2001:bad:beef:3::/64 via 2008:cafe:babe::3 dev eth0 metric 1024 mtu 15000 advmss 1440
fe80::/64 dev eth0 metric 256 mtu 1500 advmss 1440
fe80::/64 dev eth1 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth0 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth1 metric 256 mtu 1500 advmss 1440
default dev eth0 proto kernel metric 256 mtu 1500 advmss 1440
default dev eth1 proto kernel metric 256 mtu 1500 advmss 1440
unreachable default dev lo proto none metric -1 error -101
```

Komunikacija sa tom mrežom sada postoji. To se proverava komandom *ping6*.

```
root@RT ~# ping6 2001:bad:beef:3::64
```

```
64 bytes from 2001:bad:beef:3::64: icmp_seq=1 ttl=62 time=32.9 ms
64 bytes from 2001:bad:beef:3::64: icmp_seq=2 ttl=62 time=20.0 ms
64 bytes from 2001:bad:beef:3::64: icmp_seq=3 ttl=62 time=15.0 ms
64 bytes from 2001:bad:beef:3::64: icmp_seq=4 ttl=62 time=9.97 ms
```

Tasterom Ctr+C prekida se rad komande *ping*.

Za analizu rada komande *ping*, studenti mogu da pokrenu jednu od varijanti ove komande i da analiziraju razliku u izvršavanju. Posle pokretanja komande koja sledi, razlika će se ogledati u

ograničenom broju ping paketa (ukupno 5) i većoj veličini ping paketa (20000b).

```
root@RT ~# ping6 2001:bad:beef:3::64 -c 5 -s 20000
```

Putanja do mreža prema kojima je postavljena ruta se može proveriti komandama *traceroute6* i *tracpath6*. Na primer:

```
root@RT ~# tracpath6 2001:bad:beef:3::64
```

Ako je to potrebno, prethodno uneta ruta se briše komandom:

```
ip -6 route del 2001:bad:beef:3::/64 via 2008:cafe:babe::3
```

Ruta se može postaviti i sledećom komandom:

```
root@RT ~# route -A inet6 add 2001:bad:beef:3::/64 gw 2008:cafe:babe::3
```

### ***Neighbor Discovery Protocol***

Sledeći korak je upoznavanje se ND protokolom (Neighbor discovery protocol). ND protokol je u IPv6 okruženju ekvivalent ARP protokola u IPv4 okruženju. On mapira IP adrese u MAC adrese. Prikaz tabele suseda (*neighbour table*) se vrši komandom.

```
root@RT ~# ip -6 neigh show
```

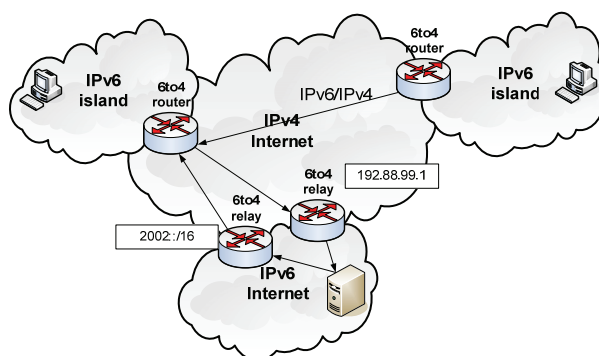
NDP sam popunjava sadržaj tabele. Sledeća komanda demonstrira slučaj kada se sadržaj tabele unosi ručno. Komanda asocira IPv6 adresu hosta 2001:bad:beef:3::64 sa njegovom fizičkom MAC adresom.

```
root@RT ~# ip -6 neigh add 2001:bad:beef:3::64 lladdr AA:01:C2:D3:74:05 dev eth0
```

### ***Tuneliranje IPv4/IPv6***

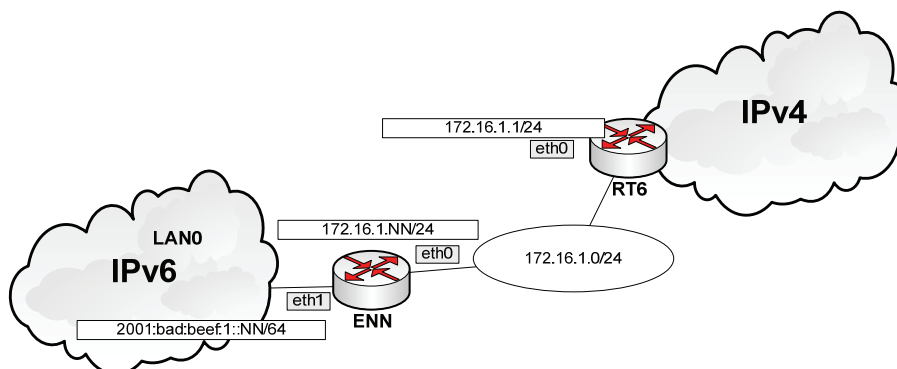
Sledeći koncept koji se uči je tuneliranje u IPv6 okruženju. Tuneliranje predstavlja proces u računarskim mrežama kada se mrežni protokol (protokol isporuke) koristi da bi enkapsulirao različite protokole koji prenose podatke. Tuneliranje u IPv4/IPv6 mrežama se može vršiti na više načina. Jedan od tih načina je 6to4.

6to4 (RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds) je Internet mehanizam tranzicije namenjen migraciji sa IPv4 na IPv6 mreže [CM01]. Taj sistem omogućava IPv6 paketima da se prenesu preko IPv4 mreža (uglavnom se misli na IPv4 Internet mreže). 6to4 se može koristiti za individualne hostove ili cele IPv6 mreže (Slika 5.8.).



Slika 5.8. Tuneliranje u realnim TCP/IP mrežama

6to4 tuneliranje koristi jednostavan mehanizam za kreiranje automatskih tunela. Linux operativni sistem podržava kreiranje konfigurisanih tunela sa sit tunnel-interfejsom.



Slika 5.9. Tuneliranje u VNLab-u

Prvo se vrši provera postojećih tunela komandom `ip -6 tunnel show`.

```
root@RT ~# ip -6 tunnel show
sit0: ipv6/ip remote any local any ttl 64 nopmtudisc
```

Dodavanje i podešavanje tunela se sastoji iz nekoliko faza. Prvo se vrši kreiranje interfejsa koji se označava sa `sit1` i podešavanje IPv4 adrese komandom `ip tunnel add sit1 mode sit remote <router-ipv4> local <host-ipv4>`.

Gde je `<router-ipv4>` IPv4 adresa udaljenog rutera koji predstavlja krajnju tačku veze – tzv. *tunnel endpoint*, a `<host-ipv4>` lokalna IPv4 adresa rutera koja povezuje lokalni ruter sa udaljenim. Ako se upotrebi parametar *any* tada se koristi bilo koja lokalna adresa rutera (tj. sve adrese postojećih interfejsa na ruteru).

```
root@RT ~# ip tunnel add sit1 mode sit ttl 64 remote 172.16.1.1 local any
```

Link se podiže sa komandom:

```
root@RT ~# ip link set sit1 up
```

Kreirani tunel može videti komandom:

```
root@RT ~# ip -6 tunnel show
sit0: ipv6/ip remote any local any ttl 64 nopmtudisc
sit1: ipv6/ip remote 172.16.1.1 local any ttl 64
```

Podešavanje default IPv6 rute preko kreiranog tunela izvršava se komandom:

```
root@RT ~# ip -6 route add ::/0 dev sit1
```

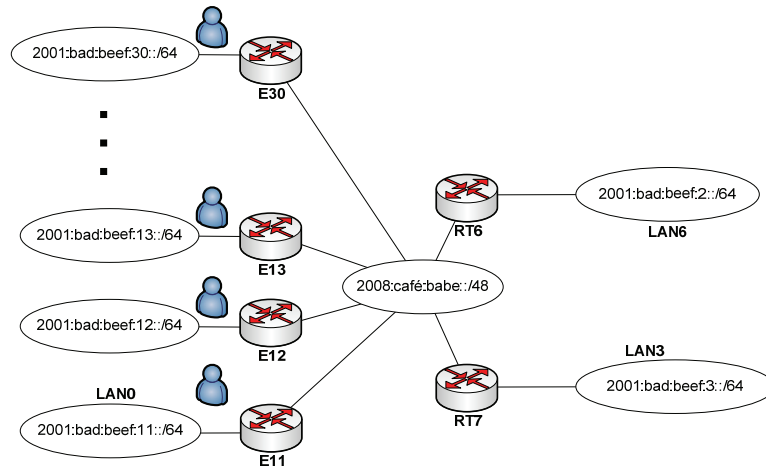
Na taj način se sav odlazni saobraćaj sa rutera koji konfiguriramo usmerava na tunel i dalje na IPv4 mrežu koja se nalazi iza hosta 172.16.1.1 (Slika 5.9.)

### **Provera stečenih znanja**

Provera stečenih znanja vrši se zadatkom koji studenti rešavaju na papiru. U zadatku je potrebno napisati komande za postavljanje rute na ruteru RT6 za mrežu 2001:bad:beef.1::/64 koja je prikazana na slici 5.9., tj. na istom scenariju.

## Rad studenata u okruženju

U toku rada studenti se nalaze u istom okruženju, ali za njih rad izgleda kao da su sami u scenariju. Paralelni rad studenata, bitna karakteristika ovog modela laboratorije, prikazan je na slici 5.10.

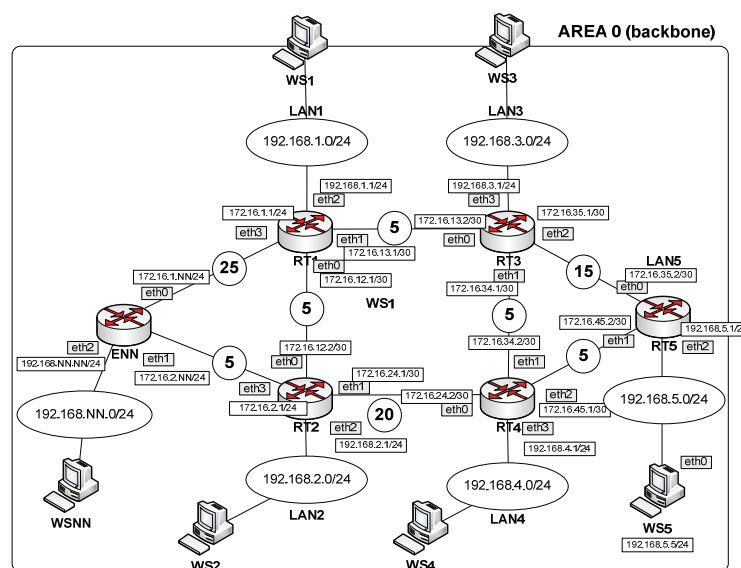


Slika 5.10. Uporedni rad studenata na istom scenariju.

## 5.7. VNLab vežba - OSPF (Open Shortest Path First) sa jednim područjem

Tema vežbe koja će biti prikazana u nastavku je OSPF (*Open Shortest Path First*) protokol za dinamičko rutiranje. Vežba sa istim protokolom je opisana u prethodnom poglavlju u okviru opisa scenarija za mrežnu laboratoriju sa lokalnim pristupom. Ovde je prezentovana verzija u scenariju prilagođenom za pristup na daljinu.

U ovom scenariju studenti konfigurišu jedan ruter ENN (Slika 5.11.) povezan na OSPF mrežu sa pet rutera RT1, RT2, RT3, RT4 i RT5. Mreža je podeljena u jedno područje – area 0 (backbone area).



Slika 5.11. Topologija VNLab OSPF scenarija sa jednim područjem

Za vreme vežbe, studenti uče o karakteristikama OSPF protokola za rutiranje kao i OSPF terminima kao što su područje (eng. *area*), cena putanje (eng. *path cost*), zatim o specifičnim ulogama rutera kao što su BR (*backbone router*), ABR (*area border router*), IR (*internal router*), ASBR (*Autonomous System Boundary Router*), LSA (*Link-State Advertisement*), NSSA (*Not-So-Stubby Area*) i dr. Svaki student sa svog radnog mesta, treba da se loguje na njemu dodeljeni ruter. Studentima se dodeljuje odgovarajući broj, tako da se studenti loguju na E11, E12 i druge rutere kao i u prethodnim vežbama.

Posvećeni ruter je ruter koji studenti konfiguriraju sami za sebe da bi ga povezali na mrežu i učinili funkcionalnim delom iste mreže. Konfiguracija rutera podrazumeva adresiranje mrežnih interfejsa rutera odgovarajućim IP adresama (u ovom slučaju IPv4). Adresiranje se vrši u okviru IP mreža definisanim scenarijom (Slika 5.11.). Za adresiranje mrežnih interfejsa koriste se Linux komande *ifconfig* i *netconfig*. Student treba da pokrene i GNU Zebra ruting servis ručno i OSPF rutiranje na ruteru koristeći isti softverski paket. Logovanjem na GNU Zebra konfiguracioni terminal i unošenjem konfiguracionih komandi studenti mogu da izvrše potrebnu konfiguraciju.

IP adrese interfejsa rutera, njihova logička imena i područja kojima oni pripadaju dati su u tabeli 5.2.

Tabela 5.2. Adresiranje interfejsa na ruterima

Čvor	IP adresa	Interfejs	Područja
RT1	172.16.12.1/30	eth0	0
RT1	172.16.13.1/30	eth1	0
RT1	192.168.1.1/24	eth2	0
RT1	172.16.1.1/24	eth3	0
RT2	172.16.12.2/30	eth0	0
RT2	172.16.24.1/30	eth1	0
RT2	192.168.2.1/24	eth2	0
RT2	172.16.2.1/24	eth3	0
RT3	172.16.13.2/30	eth0	0
RT3	172.16.34.1/30	eth1	0
RT3	172.16.35.1/30	eth2	0
RT3	192.168.3.1/24	eth3	0
RT4	172.16.24.2/30	eth0	0
RT4	172.16.34.2/30	eth1	0
RT4	172.16.45.1/30	eth2	0
RT4	192.168.4.1/24	eth3	0
RT5	172.16.35.2/30	eth0	0
RT5	172.16.45.2/30	eth1	0
RT5	192.168.5.1/24	eth2	0

Pre početka rada potrebno je proveriti da li su dodeljene odgovarajuće IP adrese interfejsima na ruteru. Provera IP adresa vrši se komandom *ifconfig / more*. Ako adrese nisu ispravno dodeljene mogu se promeniti komandom *netconfig*, npr.

```
root@RT1 ~# netconfig --ip=172.16.1.NN --netmask=255.255.255.0 -d eth0
root@RT1 ~# netconfig --ip=172.16.2.NN --netmask=255.255.255.0 -d eth1
```

Primena unetih parametara i dodela IP adresa interfejsima se inicira komandom *service network restart* koja prvo obori interfejse, pa ih kasnije podigne sa novim parametrima.

```
root@RT1 ~# service network restart
Shutting down interface eth0:          [ OK ]
Shutting down interface eth1:          [ OK ]
Shutting down loopback interface:      [ OK ]
Setting network parameters:            [ OK ]
Bringing up loopback interface:        [ OK ]
Bringing up interface eth0:            [ OK ]
Bringing up interface eth1:            [ OK ]
```



Na ruteru ENN potrebno je startovati softver za rutiranje, tj. programe zebra i ospfd. To se vrši komandama.

```
root@RT ~# zebra -d
```

```
root@RT ~# ospfd -d
```

Pristup rutinng softveru se vrši komandom *telnet localhost 2604* ili *telnet localhost ospfd*. Password je **zebra**.

```
root@RT# telnet localhost 2604
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.

Hello, this is zebra (version 0.94).
Copyright 1996-2002 Kunihiro Ishiguro.

User Access Verification

Password:
```

Konfigurisanje OSPF protokola vrši se sledećim komandama, a prilikom konfigurisanja potrebno je uneti sve mreže koje ruter treba da oglašava tj. sve mreže koje su povezane direktno na ruter (172.16.1.0/24, 172.16.2.0/24 i 192.168.NN.NN).

```
ospfd > enable
ospfd # configure terminal
ospfd(config)# router ospf
ospfd(config)# router-id 0.0.0.NN
ospfd(config)# router ospf
ospfd(config-router)# passive-interface eth2
ospfd(config-router)# network 172.16.1.0/24 area 0
ospfd(config-router)# network 172.16.2.0/24 area 0
ospfd(config-router)# network 192.168.NN.0/24 area 0
ospfd(config-router)# exit
ospfd(config)# interface eth0
ospfd(config-if)# ip ospf cost 25
ospfd(config-if)# exit
ospfd(config)# interface eth1
ospfd(config-if)# ip ospf cost 5
ospfd(config-if)# exit
ospfd(config)# write
Configuration saved to /usr/local/etc/ospfd.conf
ospfd (config)# exit
```

#### Listing 5.1. Komande za konfigurisanje studentskog OSPF rutera

Konfiguracija se može pogledati komandom *show running-config* (dok se studenti nalaze u modu za konfigurisanje - enable) u rutinng softveru, tj. pre izlaska iz zebra softvera.

```
ospfd(config)# show running-config
Current configuration:
!
hostname ospfd
password zebra
log stdout
!
!
!
interface lo
!
interface eth0
 ip ospf cost 25
!
interface eth1
 ip ospf cost 5
!
interface eth2
```

```

!
interface eth3
!
router ospf
passive-interface eth2
 network 172.16.1.0/24 area 0.0.0.1
 network 172.16.2.0/24 area 0.0.0.1
!
line vty
!
end

```

## Listing 5.2. Konfiguracija studentskog rutera

Izlazak is zebra softvera se vrši komandom:

```

ripd# exit
Connection closed by foreign host.
root@RT #

```

Tabela rutiranja koja se dobija komandom `route -n` treba da ima sledeći izgled:

```

root@RT ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
172.16.12.0 172.16.2.1 255.255.255.252 UG 10 0 0 eth1
172.16.45.0 172.16.2.1 255.255.255.252 UG 25 0 0 eth1
172.16.13.0 172.16.2.1 255.255.255.252 UG 15 0 0 eth1
172.16.35.0 172.16.2.1 255.255.255.252 UG 30 0 0 eth1
172.16.34.0 172.16.2.1 255.255.255.252 UG 20 0 0 eth1
172.16.24.0 172.16.2.1 255.255.255.252 UG 25 0 0 eth1
192.168.5.0 172.16.2.1 255.255.255.0 UG 35 0 0 eth1
192.168.4.0 172.16.2.1 255.255.255.0 UG 30 0 0 eth1
172.16.2.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1
192.168.3.0 172.16.2.1 255.255.255.0 UG 25 0 0 eth1
192.168.2.0 172.16.2.1 255.255.255.0 UG 15 0 0 eth1
192.168.1.0 172.16.2.1 255.255.255.0 UG 20 0 0 eth1
172.16.1.0 0.0.0.0 255.255.255.0 U 30 0 0 eth1
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo

```

Sada je potrebno proveriti da li postoji komunikacija između rutera ENN i radne stanice WS1. To se proverava komandom `ping 192.168.5.5` sa rutera ENN.

```

root@RT ~# ping 192.168.5.5
PING 192.168.5.5 (192.168.41.2) 56(84) bytes of data.
64 bytes from 192.168.5.5: icmp_seq=1 ttl=62 time=32.9 ms
64 bytes from 192.168.5.5: icmp_seq=2 ttl=62 time=20.0 ms
64 bytes from 192.168.5.5: icmp_seq=3 ttl=62 time=15.0 ms
64 bytes from 192.168.5.5: icmp_seq=4 ttl=62 time=9.97 ms
64 bytes from 192.168.5.5: icmp_seq=5 ttl=62 time=20.6 ms

--- 192.168.5.200 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4042ms
rtt min/avg/max/mdev = 9.979/19.702/32.917/7.643 ms

```

Komandom `traceroute 192.168.5.5` pokazuje se putanja ka WS1, i ona ide preko rutera RT2 (172.16.2.1), RT1 (172.16.12.1), RT3 (172.16.13.2), RT4 (172.16.34.2) i RT5 (172.16.45.2) do WS2 (192.168.5.5), tj. putanjom sa najmanjom cenom (Slika 5.12.).

```

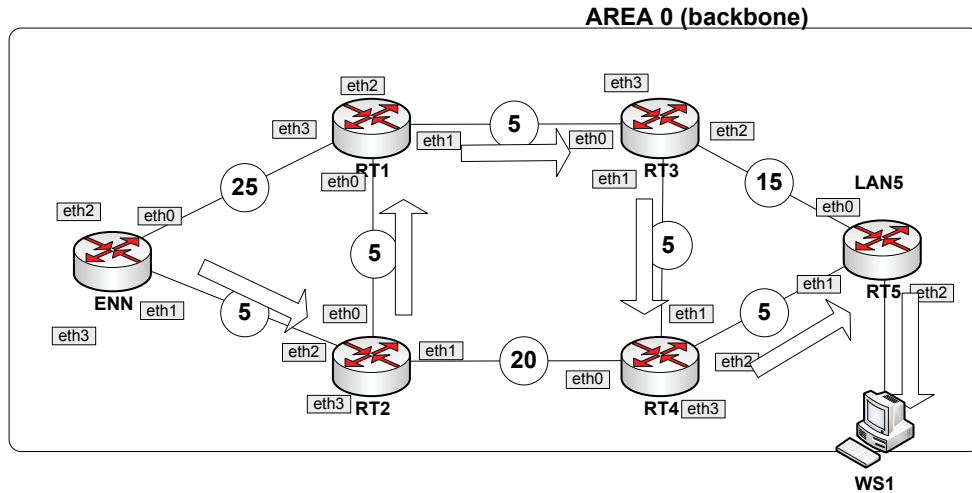
root@WS ~# traceroute 192.168.5.5 -n
traceroute to 192.168.5.200 (192.168.41.2), 30 hops max, 40 byte packets
 1 172.16.2.1 5.320 ms 6.044 ms 2.892 ms
 2 172.16.12.1 12.812 ms 18.654 ms 8.078 ms
 3 172.16.13.2 19.266 ms 23.663 ms 17.321 ms

```

```

4 172.16.34.2 59.849 ms 39.848 ms 19.685 ms
5 172.16.45.2 18.981 ms 12.016 ms 66.694 ms
6 192.168.5.5 135.574 ms 179.866 ms 149.808 ms

```



Slika 5.12. Putanja dobijena na osnovu OSPF protokola

Simulacija prekida linka pokazuje studentima prednosti dinamičkog rutiranja. Ova simulacija može lako da se izvede obaranjem *eth1* interfejsa. U početnoj konfiguraciji sve rute vode preko tog interfejsa zbog manje cene putanje, te je stoga interfejs *eth1* taj koji treba da bude oboren. Posle komande:

```
root@WS ~# interface eth1 down
```

Tabela rutiranja ima sledeći izgled. Vidi se da sada sve rute idu preko izlaznog interfejsa *eth0*.

```

root@RT ~# route -n
Kernel IP routing table
Destination  Gateway      Genmask      Flags  Metric  Ref    Use    Iface
172.16.12.0  172.16.1.1  255.255.255.252  UG    30      0      0      eth0
172.16.45.0  172.16.1.1  255.255.255.252  UG    40      0      0      eth0
172.16.13.0  172.16.1.1  255.255.255.252  UG    30      0      0      eth0
172.16.35.0  172.16.1.1  255.255.255.252  UG    45      0      0      eth0
172.16.34.0  172.16.1.1  255.255.255.252  UG    35      0      0      eth0
172.16.24.0  172.16.1.1  255.255.255.252  UG    50      0      0      eth0
192.168.5.0  172.16.1.1  255.255.255.0    UG    50      0      0      eth0
192.168.4.0  172.16.1.1  255.255.255.0    UG    45      0      0      eth0
192.168.3.0  172.16.1.1  255.255.255.0    UG    40      0      0      eth0
192.168.2.0  172.16.1.1  255.255.255.0    UG    40      0      0      eth0
192.168.1.0  172.16.1.1  255.255.255.0    UG    35      0      0      eth0
172.16.1.0   0.0.0.0     255.255.255.0    U      0        0      0      eth0
127.0.0.0   0.0.0.0     255.0.0.0        U      0        0      0      lo

```

Sada i *traceroute* prikazuje drugu putanju. Studenti na ovaj način imaju praktičan prikaz rada protokola dinamičkog rutiranja i mogu da shvate važnost razmene podataka između rutera.

```

root@RT ~# traceroute 192.168.5.5 -n
traceroute to 192.168.5.5 (192.168.41.2), 30 hops max, 40 byte packets
 1  172.16.1.1  8.332 ms  0.922 ms  0.716 ms
 2  172.16.13.2  4.338 ms  4.030 ms  7.759 ms
 3  172.16.34.2  21.596 ms  16.054 ms  9.819 ms
 4  172.16.45.2  18.402 ms  10.082 ms  4.666 ms
 5  192.168.5.5  12.137 ms  4.722 ms  7.445 ms

```

Analiza OSPF podataka se može izvršiti koristeći naredbe prikazane u nastavku. Tako se može proveriti niz podataka koji su vezani za OSPF protokol. Prikaz podataka o OSPF rutama se vrši na sledeći način:

```
ospfd> show ip ospf route
===== OSPF network routing table =====
N IA 0.0.0.0/0          [11] area: 0.0.0.1
                        via 172.16.15.1, eth0
                        via 172.16.45.1, eth1
N   172.16.14.0/30     [30] area: 0.0.0.1
                        via 172.16.15.1, eth0
                        via 172.16.45.1, eth1
N   172.16.15.0/30     [10] area: 0.0.0.1
                        directly attached to eth0
N   172.16.45.0/30     [10] area: 0.0.0.1
                        directly attached to eth1
N   192.168.11.0/24    [20] area: 0.0.0.1
                        via 172.16.15.1, eth0
N   192.168.41.0/24    [20] area: 0.0.0.1
                        via 172.16.45.1, eth1
N   192.168.51.0/24    [10] area: 0.0.0.1
                        directly attached to eth2
N   192.168.52.0/24    [10] area: 0.0.0.1
                        directly attached to eth3

===== OSPF router routing table =====
R   172.16.15.1        [10] area: 0.0.0.1, ABR
                        via 172.16.15.1, eth0
                        via 172.16.15.1, eth0
R   172.16.45.1        [10] area: 0.0.0.1, ABR
                        via 172.16.45.1, eth1

===== OSPF external routing table =====
```

### Listing 5.3. Prikaz komande show ip ospf route

Prikaz podataka o OSPF mrežnim inerfejsima se vrši komandom:

```
ospfd> show ip ospf interface
lo is up, line protocol is up
  OSPF not enabled on this interface
eth0 is up, line protocol is up
  Internet Address 172.16.15.2/30, Area 0.0.0.1 [Stub]
  Router ID 172.16.45.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 172.16.15.1, Interface Address 172.16.15.1
  Backup Designated Router (ID) 172.16.45.2, Interface Address 172.16.15.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
  Neighbor Count is 1, Adjacent neighbor count is 1
eth1 is up, line protocol is up
  Internet Address 172.16.45.2/30, Area 0.0.0.1 [Stub]
  Router ID 172.16.45.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 172.16.45.1, Interface Address 172.16.45.1
  Backup Designated Router (ID) 172.16.45.2, Interface Address 172.16.45.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:07
  Neighbor Count is 1, Adjacent neighbor count is 1
eth2 is up, line protocol is up
  Internet Address 192.168.51.1/24, Area 0.0.0.1 [Stub]
  Router ID 172.16.45.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 172.16.45.2, Interface Address 192.168.51.1
```

```

No backup designated router on this network
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  No Hellos (Passive interface)
Neighbor Count is 0, Adjacent neighbor count is 0
eth3 is up, line protocol is up
  Internet Address 192.168.52.1/24, Area 0.0.0.1 [Stub]
  Router ID 172.16.45.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 172.16.45.2, Interface Address 192.168.52.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  No Hellos (Passive interface)
  Neighbor Count is 0, Adjacent neighbor count is 0

```

#### Listing 5.4. Prikaz komande show ip ospf interface

A prikaz podataka o OSPF bazi na sledeći način:

```

ospfd> show ip ospf database

      OSPF Router with ID (172.16.45.2)

      Router Link States (Area 0.0.0.1 [Stub])

Link ID        ADV Router    Age         Seq#          CkSum         Link count
172.16.15.1    172.16.15.1   209        0x80000008   0x4384        3
172.16.34.2    172.16.34.2   1177       0x80000001   0x68a9        2
172.16.45.1    172.16.45.1   218        0x8000000d   0x74b6        3
172.16.45.2    172.16.45.2   183        0x80000004   0x9add        4
192.168.52.1   192.168.52.1  274        0x8000000a   0xa566        4

      Net Link States (Area 0.0.0.1 [Stub])

Link ID        ADV Router    Age         Seq#          CkSum
172.16.14.1    172.16.15.1   270        0x80000001   0xb15d
172.16.15.1    172.16.15.1   209        0x80000001   0xb458
172.16.45.1    172.16.45.1   222        0x80000001   0xa50d

      Summary Link States (Area 0.0.0.1 [Stub])

Link ID        ADV Router    Age         Seq#          CkSum         Route
0.0.0.0        172.16.15.1   1454       0x80000001   0x91ff        0.0.0.0/0
0.0.0.0        172.16.45.1   276        0x80000001   0xbeb4        0.0.0.0/0

```

#### Listing 5.5. Prikaz komande show ip ospf database

Prikaz podataka o susednim ruterima je dat u nastavku:

```

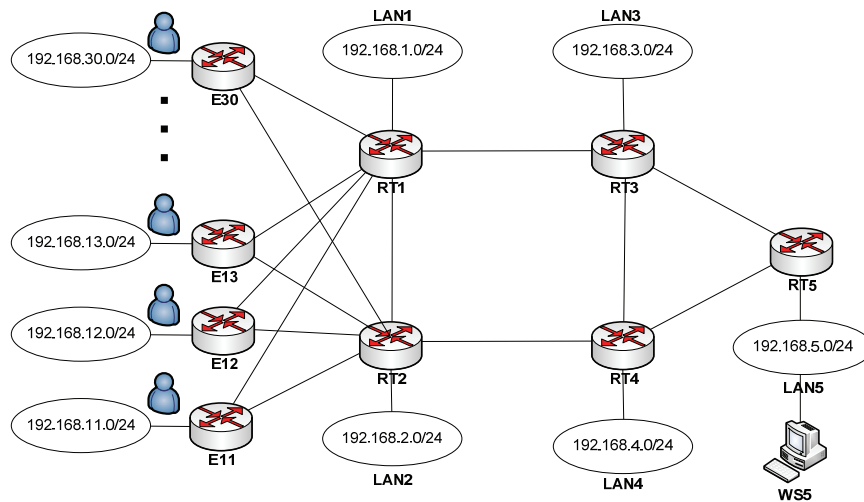
ospfd> show ip ospf neighbor

Neighbor ID    Pri   State   Dead Time   Address          Interface          RXmtL  RqstL  DBsmL
172.16.15.1    1     Full/DR 00:00:39   172.16.15.1    eth0:172.16.1.2   0      0      0
172.16.45.1    1     Full/DR 00:00:34   172.16.45.1    eth1:172.16.4.2   0      0      0

```

#### Listing 5.6. Prikaz komande show ip ospf neighbor

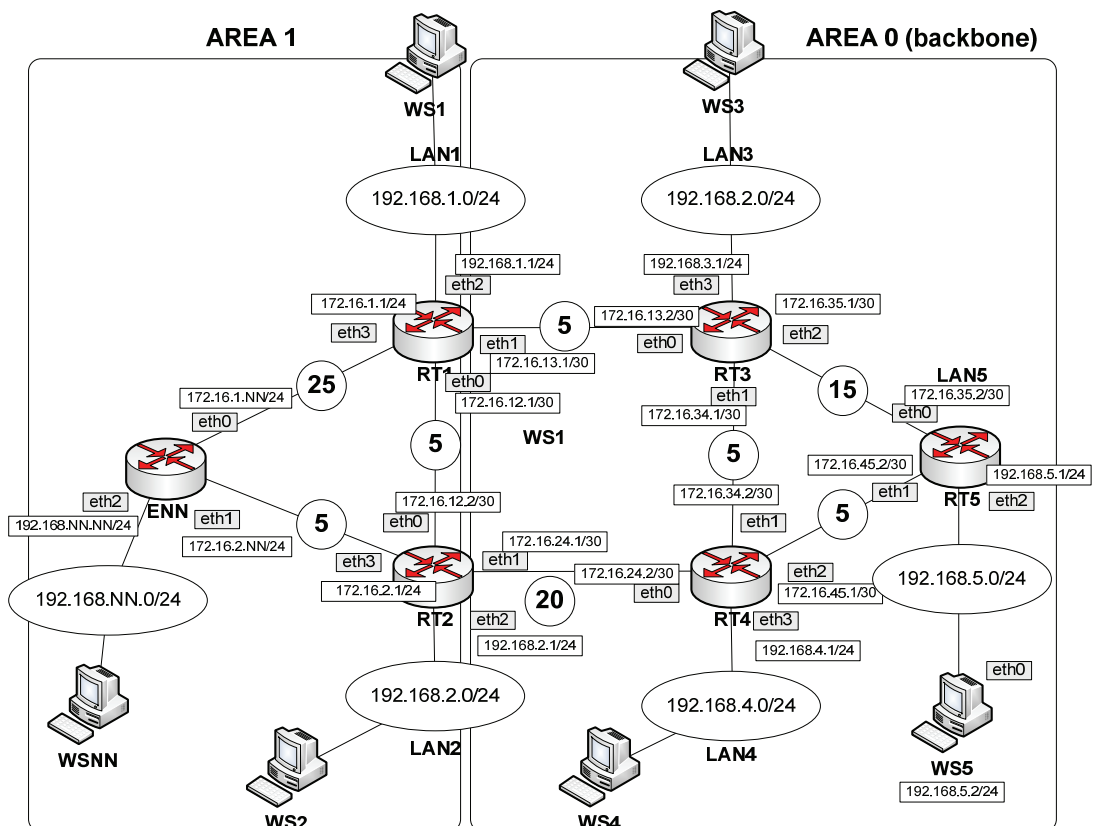
Prikaz rada studenata u toku laboratorijske vežbe i njihov istovremeni pristup na virtuelnu mrežu dat je na slici 5.13.



Slika. 5.13. Paralelni pristup studenata na virtuelnu mrežu sa OSPF protokolom

### 5.8. VNetLab vežba - OSPF (Open Shortest Path First) sa više područja

U sledećem scenariju studenti konfiguriraju jedan ruter ENN (Slika 5.14.) povezan na OSPF mrežu sa pet rutera RT1, RT2, RT3, RT4 i RT5 koja je organizovana u dva područja – area 0 (backbone area) i area 1. Druga varijanta scenarija uključuje poddelu u tri područja, dodavanje područja (area) 2 sa ruterima RT2, RT4 i RT5 u njoj. Zbog testiranja je potrebno da u scenariju postoji najmanje jedna radna stanica - WS5. Po potrebi, taj broj se može povećati sa WS2, WS3 itd. IP adresiranje, podešavanje područja i drugi važni mrežni parametri su prikazani na slici 5.14. koja se nalazi u materijalu koji studentima daje sve informacije o mreži koje su im potrebne.



Slika 5.14. Topologija VNetLab OSPF scenarija sa dva područja

Prvi korak je inicijalno podešavanje IP adresa koje se vrši na isti način i sa istim IP adresama kao u prethodnom primeru. Nakon toga sledi konfiguracija GNU Zebra rutera. Pristup Zebra softveru radi se isto kao u pethodnom primeru. Spisak komandi potreban za konfiguraciju ovog scenarija dat je u Listingu 5.7.

```
ospfd > enable
ospfd # configure terminal
ospfd(config)# router ospf
ospfd(config-router)# passive-interface eth2
ospfd(config-router)# network 172.16.1.0/24 area 1
ospfd(config-router)# network 172.16.2.0/24 area 1
ospfd(config-router)# network 192.168.NN.0/24 area 1
ospfd(config-router)# area 1 stub no-summary
ospfd(config-router)# exit
ospfd(config)# interface eth0
ospfd(config-if)# ip ospf cost 25
ospfd(config-if)# exit
ospfd(config)# interface eth1
ospfd(config-if)# ip ospf cost 5
ospfd(config-if)# exit
ospfd(config)# write
Configuration saved to /usr/local/etc/ospfd.conf
ospfd (config)# exit
```

**Listing 5.7.** Komande za konfiguraciju ENN rutera u scenariju sa više područja

Provera konfiguracije se može izvršiti komandom `show running-config` (Listing 5.8.):

```
ospfd(config)# show running-config
Current configuration:
!
hostname ospfd
password zebra
log stdout
!
!
!
interface lo
!
interface eth0
 ip ospf cost 25
!
interface eth1
 ip ospf cost 5
!
interface eth2
!
interface eth3
!
router ospf
passive-interface eth2
passive-interface eth3
network 172.16.1.0/24 area 0.0.0.1
network 172.16.2.0/24 area 0.0.0.1
network 192.168.NN.0/24 area 0.0.0.1

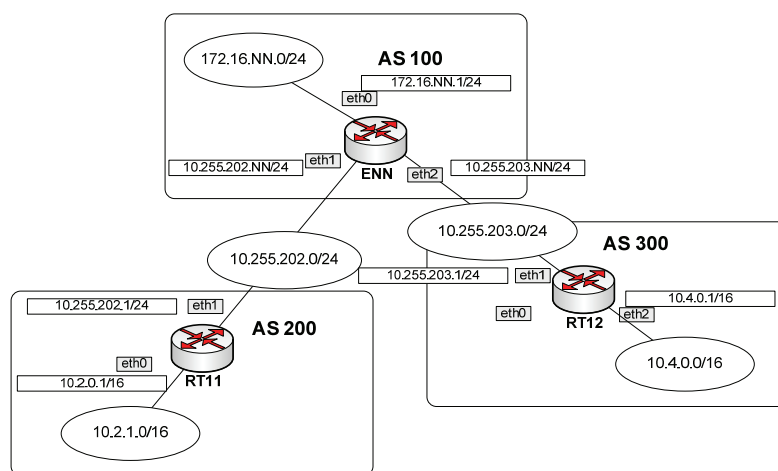
area 1 stub no-summary
!
line vty
!
end
```

**Listing 5.8.** Provera ENN konfiguracije za scenario sa više područja

Na kraju vežbe, studentima su postavljena pitanja iz domena naučenih karakteristika OSPF protokola. Od studenta se traži da objasne vrste OSPF rutera (ABR, ASBR, BR i IR) kao i da izvrše konfiguraciju drugih rutera u istom scenariju, npr. rutera RT4. Jedna od varijanti zadatka je i konfiguracija rutera RT4, ako on pripada području 2 zajedno sa ruterima RT3 i RT5.

## 5.9. VNLab vežba - BGP-4

Sledeća vežba odnosi se na BGP-4 protokol. Vežba i mrežni scenario zasnovani su na realnom scenariju koji se koristi za Cisco sertifikaciju i ispite [SC08]. Za razliku od OSPF protokola, BGP-4 spada u protokole spoljašnjeg rutiranja - EGP (eng. *Exterior Gateway Protocol*). Na slici je prikazana topologija mreže sa 3 rutera koji se nalaze u tri autonomna sistema (ENN, RT11 i RT12) i 5 mreža koje su međusobno povezane preko njih. U vežbi je potrebno izvršiti konfigurisanje rutera za protokol BGP-4, da bi se tabele rutiranja dinamički ažurirale (Slika 5.15).



Slika 5.15. Topologija VNLab BGP-4 scenarija

Student treba da proveri IP adrese koje su dodeljene ruterima komandom *netconfig*. Ako adrese nisu ispravno dodeljene mogu se promeniti komandama.

```
root@RT1 ~# netconfig --ip=10.NN.0.1 --netmask=255.255.0.0 -d eth0
root@RT1 ~# netconfig --ip=10.255.202.NN --netmask=255.255.255.0 -d eth1
root@RT1 ~# netconfig --ip=10.255.203.NN --netmask=255.255.255.0 -d eth2
```

Na ruteru ENN potrebno je startovati softver za rutiranje, tj. programe zebra i bgpd. To se vrši komandama:

```
root@RT ~# zebra -d
root@RT ~# bgpd -d
```

Pristup rutingu softveru se vrši komandom *telnet localhost 2605* ili *telnet localhost bgpd*. Password je zebra.

```
root@RT# telnet localhost 2605
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.

Hello, this is zebra (version 0.94).
Copyright 1996-2002 Kunihiro Ishiguro.
```



```
User Access Verification
```

```
Password:
```

Konfigurisanje se vrši sledećim komandama, a prilikom konfigurisanja potrebno je uneti sve mreže koje ruter treba da oglašava tj. sve mreže koje su povezane direktno na ruter (10.NN.0.0/16, 10.255.202.0/24 i 10.255.203.0/24):

```
bgpd > enable
bgpd # configure terminal
bgpd(config)# router bgp 100
bgpd(config-router)# bgp router-id 10.NN.0.1
bgpd(config-router)# network 10.NN.0.0/16
bgpd(config-router)# network 10.255.202.0/24
bgpd(config-router)# network 10.255.203.0/24
bgpd(config-router)# neighbor 10.255.202.1 remote-as 200
bgpd(config-router)# neighbor 10.255.203.1 remote-as 300
bgpd(config-router)# write
Configuration saved to /usr/local/etc/bgpd.conf

bgpd(config-router)# exit
bgpd (config)# exit
```

#### Listing 5.9. Komande za konfiguraciju rutera ENN za dati scenario

Konfiguracija se može pogledati komandom *show running-config* (dok se nalazite u modu za konfigurisanje - enable) u rutingu softveru, tj. pre izlaska iz zebra softvera.

Izgled tabele rutiranja posle komande *route -n* ima sledeći izgled.

```
root@RT ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
10.255.202.0 0.0.0.0 255.255.255.0U 0 0 0 eth1
10.255.203.0 0.0.0.0 255.255.255.0U 0 0 0 eth2
10.2.0.0 10.255.202.1 255.255.0.0 UG 0 0 0 eth1
10.3.0.0 10.255.202.1 255.255.0.0 UG 0 0 0 eth1
10.4.0.0 10.255.202.1 255.255.0.0 UG 0 0 0 eth1
10.28.0.0 0.0.0.0 255.255.0.0 U 0 0 0 eth0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```

Analiza BGP podataka se vrši komandama *show ip bgp* (Listing 5.10.) i *show ip bgp neighbors* (Listing 5.11.):

```
bgp> show ip bgp
BGP table version is 0, local router ID is 10.28.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
* 10.2.0.0/16	10.255.203.1			0 300	200 i
*>	10.255.202.1	0		0 200	i
*> 10.3.0.0/16	10.255.203.1	0		0 300	i
*	10.255.202.1			0 200	300 i
*> 10.4.0.0/16	10.255.203.1	0		0 300	i
*	10.255.202.1			0 200	300 i
*> 10.28.0.0/16	0.0.0.0	0	32768		i
*> 10.255.202.0/24	0.0.0.0	0	32768		i
* 10.255.203.0/24	10.255.203.1	0		0 300	?
*>	0.0.0.0	0	32768		i
*> 10.255.255.208/30					

```

          10.255.203.1          0          0 300 ?
*         10.255.202.1          0          0 200 300 ?

Total number of prefixes 7

```

### Listing 5.10. Prikaz tabele rutiranja u GNU Zebra softveru

```

bgp> show ip bgp neighbors
BGP neighbor is 10.255.202.1, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 10.2.0.1
  BGP state = Established, up for 00:22:47
  Last read 00:00:47, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and receive (old and new)
    Address family IPv4 Unicast: advertised and received
  Received 28 messages, 0 notifications, 0 in queue
  Sent 28 messages, 0 notifications, 0 in queue
  Router refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds

For address family: IPv4 Unicast
  Community attribute sent to this neighbor (both)
  4 accepted prefixes

  Connections established 1; dropped 0
  Last reset never
Local host: 10.255.202.28, Local port: 3073
Foreign host: 10.255.202.1, Foreign port: 179
Nextthop: 10.255.202.28
Read thread: on Write thread: off

BGP neighbor is 10.255.203.1, remote AS 300, local AS 100, external link
  BGP version 4, remote router ID 10.3.0.1
  BGP state = Established, up for 00:22:38
  Last read 00:00:39, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and receive (old and new)
    Address family IPv4 Unicast: advertised and received
  Received 24 messages, 0 notifications, 0 in queue
  Sent 24 messages, 0 notifications, 0 in queue
  Router refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds

For address family: IPv4 Unicast
  Community attribute sent to this neighbor (both)
  5 accepted prefixes

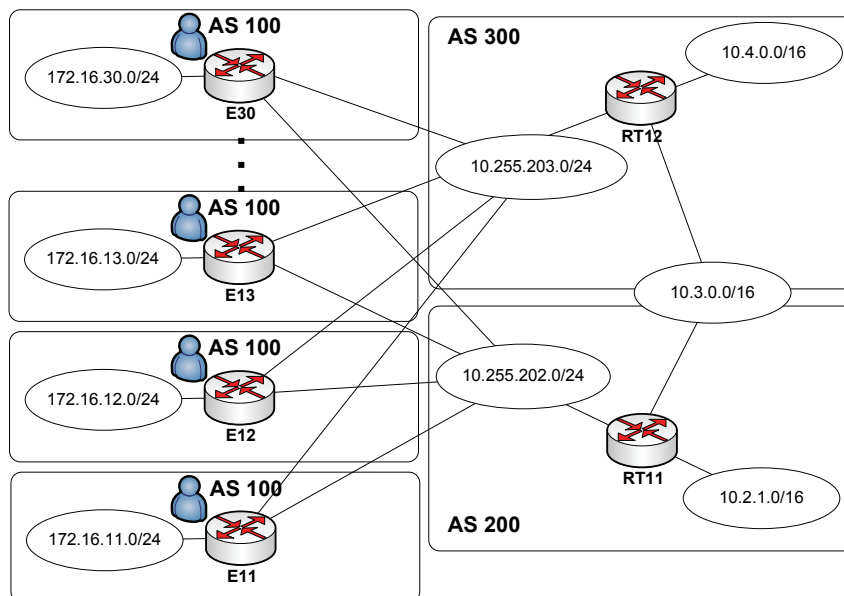
  Connections established 1; dropped 0
  Last reset never
Local host: 10.255.203.28, Local port: 3073
Foreign host: 10.255.203.1, Foreign port: 179
Nextthop: 10.255.203.28
Read thread: on Write thread: off

```

### Listing 5.11. Prikaz BGP podataka u GNU Zebra softveru

Zbog provjere znanja studenti treba da odgovore u pismenom obliku kojim komandama GNU Zebra rutera bi se trebao konfigurirati ruter RT11 na slici da bi mogao da funkcioniše ispravno.

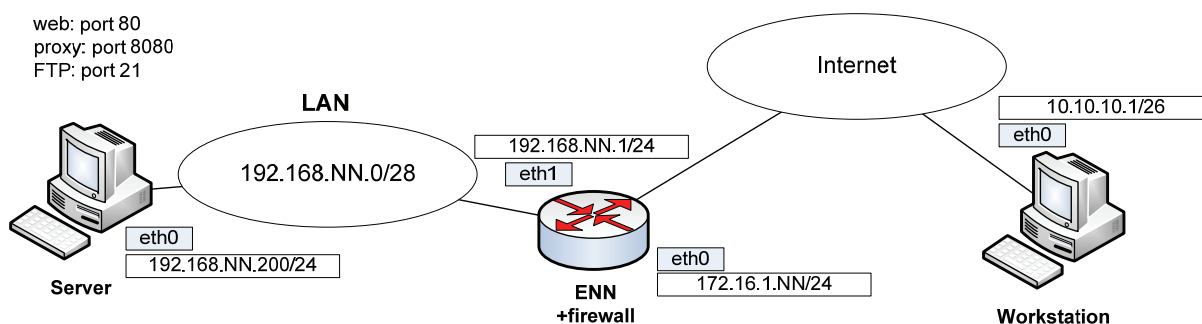
Prikaz rada studenata i njihove konekcije na VNlab u radu sa BGP protokolom dat je na slici (Slika 5.16.).



Slika 5.16. Prikaz rada studenata u vežbi sa BGP protokolom

### 5.10. VNLab vežba - Firewall

Scenario za učenje fajervol koncepta prikazan je na slici 5.17. Dodela IP adresa se vrši u skladu sa konfiguracijom mreže koja je prikazana. U toj mreži, ruter ENN štiti mrežu 192.168.NN.0/24 i server 192.168.NN.200 od neovlašćenog pristupa sa spoljašnje mreže (Internet). Zaštita mreže vrši se komandom *iptables*. Zaštita mreže se vrši filtriranjem paketa na osnovu unetih pravila.



Slika 5.17. Scenario za učenje fajervol koncepta

Sadržaj *iptables* tabele pravila može se videti komandom:

```
iptables -L -n
```

Na početku vežbe, da bi se izbegla pojava grešaka, studenti treba da obrišu sva prethodno uneta pravila. To se radi komandom:

```
iptables -F
```

Da bi se dozvolio pristup sa radne stanice (na slici 5.17. to je *workstation*) sa adresom 10.10..10.1 na FTP server na računaru 192.168.NN.200 (FTP server je na portu 21) potrebno je uneti komandu:

```
iptables -A FORWARD -p tcp -s 10.10.10.1/32 -d 192.168.NN.200/32 --dport 21 -j ACCEPT
```

Pravilo se odnosi na pakete koji dolaze sa spoljašnje mreže (Internet), prolaze kroz ENN ruter (fajervol) i odlaze u mrežu 192.168.NN.0/24. Zbog svega toga se koristi FORWARD lanac uz komandu *iptables*.

Da bi se dozvolio pristup sa radne stanice sa adresom 10.10.10.1 ka *proxy* serveru koji se nalazi na računaru 192.168.NN.200 (proxy server je na portu 8080) potrebno je otkucati komandu:

```
iptables -A FORWARD -p tcp -s 10.10.10.1/32 -d 192.168.NN.200/32 --dport 8080 -j ACCEPT
```

Da bi se zabranio pristup svim ostalim računarima sa spoljašnje mreže ka serveru na računaru 192.168.NN.200, potrebno je otkucati komandu:

```
iptables -A FORWARD -s 0.0.0.0/0 -d 192.168.NN.200/32 -j REJECT
```

U ovom slučaju nije potrebno uneti opciju `-p tcp`, jer se zabranjuje saobraćaj za sve protokole (tcp, udp i icmp, a ne samo tcp). Takođe, u ovom slučaju nije potrebno uneti odredišni port, npr. `-dport 8080` budući da je potrebno zabraniti ulaz na sve portove, a ne samo na port 8080.

Adresa 0.0.0.0/0 menja sve računare, tj. sve moguće adrese. Ako se source (-s) ili destination (-d) adresa izostavi, npr. `iptables -A FORWARD -d 192.168.NN.200/32 -j REJECT`, izostavljene adrese će se zameniti sa 0.0.0.0/0.

U slučajevima kada treba da se zabrani pristup ENN ruteru sa svih ostalih računara, tada se koristi lanac INPUT, pa je potrebno otkucati komandu:

```
iptables -A INPUT -s 0.0.0.0/0 -d 172.16.1.NN/32 -j REJECT
```

Kada treba da se ukine pristup proxy serveru sa spoljne mreže, potrebno je obrisati uneto pravilo. Brisanje se vrši komandom:

```
iptables -D FORWARD -p tcp -s 10.10.10.1/32 -d 192.168.NN.0/24 --dport 8080 -j ACCEPT
```

Lista unetih pravila se dobija komandom:

```
iptables -L -n
```

```
Chain INPUT (policy ACCEPT)
target     prot opt source                destination           reject-with icmp-port-unreachable
REJECT     all  --  0.0.0.0/0             172.16.1.11

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination           tcp dpt:21 reject-with icmp-port-unreachable
ACCEPT     tcp  --  10.10.10.1            192.168.11.200
REJECT     all  --  0.0.0.0/0             192.168.11.200

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

### **Adresna translacija**

Važan koncept u računarskim mrežama je adresna translacija ili NAT (*Network Address Translation*). Studenti uče ovaj koncept na primeru mrežne konfiguracije prikazane na slici 5.17. ENN ruter je potrebno konfigurisati tako da se omogući adresna translacija, tj. dodela IP adrese rutera (172.16.1.NN) hostovima iz unutrašnje mreže (192.168.NN.0/24).

U ovom slučaju potrebno je koristiti nat tabelu (`-t nat`), POSTROUTING lanac i izlazni interfejs

eth0 (-o eth0):

```
iptables -A POSTROUTING -t nat -s 192.168.NN.0/24 -o eth0 -j SNAT --to-source 172.16.1.NN
```

### ***Preusmeravanje na unutrašnju mrežu celokupnog saobraćaja***

U okviru istog scenarija predstavljena je situacija u kojoj je veb server postavljen u unutrašnju LAN mrežu. Veb serveru je dodeljena privatna IP adresa. Ovo je učinjeno radi dodatne sigurnosti servera. Problem je što spoljni računari ne mogu da pristupe veb serveru zbog privatne IP adrese. Da bi se omogućio pristup spoljnim računarima ka veb serveru potrebno je izvršiti preusmeravanje veb zahteva, koji dolaze sa spoljašnje mreže na ruter sa „javnom“ IP adresom 172.16.1.NN<sup>1</sup>, na privatnu IP adresu veb servera (192.168.NN.200).

Da bi se to postiglo koristi se nat tabela (-t nat) i PREROUTING lanac. Celokupan saobraćaj se preusmerava komandom:

```
iptables -A PREROUTING -t nat -d 172.16.1.NN -j DNAT --to-destination 192.168.NN.200
```

Samo veb saobraćaj se preusmerava komandom:

```
iptables -A PREROUTING -p tcp -t nat -d 172.16.1.NN --dport 80 -j DNAT --to-destination 192.168.NN.200
```

Lista unetih pravila se dobija komandom

```
iptables -L -n
```

```
Chain PREROUTING (policy ACCEPT)
target     prot opt source                destination
DNAT       tcp  --  172.16.1.11           0.0.0.0/0            tcp dpt:80 to:192.168.11.200

Chain POSTROUTING (policy ACCEPT)
target     prot opt source                destination
SNAT       all  --  192.168.11.0/24      0.0.0.0/0            to:172.16.1.11

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Kada studenti završe programirani deo vežbe, sledi provera znanja studenata. Za proveru znanja studenata koriste se sledeći zadaci. Na primer, za mrežnu konfiguraciju prikazanu na slici 5.18. potrebno je izvršiti konfiguraciju fajervola tako da se:

a) dozvoli pristup sa računara 212.200.200.200 na SMTP server 192.168.200.3.

b) zabrani pristup za sve računare sa Interneta na mrežu 192.168.200.0/24

c) zabrani pristup za sve računare sa Interneta na ENN ruter + fajervol

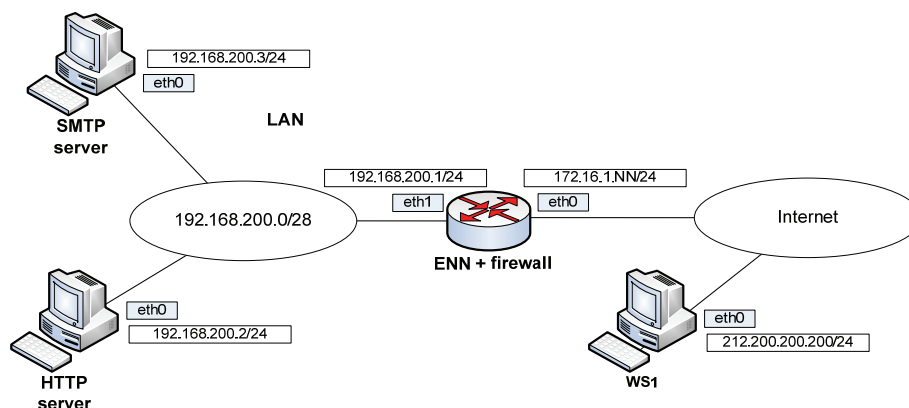
d) izvrši preusmeravanje SMTP zahteva koji dolaze sa spoljašnje mreže na ruter sa javnom IP adresom (212.200.200.1), na privatnu IP adresu SMTP servera (192.168.200.3).

---

<sup>1</sup> Adresa 172.16.1.NN je takođe privatna, ali se u ovoj vežbi koristi za demonstraciju preusmeravanja zbog kompatibilnosti sa ostalim scenarijima.

e) izvrši preusmeravanje HTTP zahteva koji dolaze sa spoljašnje mreže na ruter sa javnom IP adresom (212.200.200.1), na privatnu IP adresu SMTP servera (192.168.200.2).

f) omogući izlazak na Internet mrežu hostovima 192.168.200.2 i 192.168.200.3 (tj. za mrežu 192.168.200.0/24) pomoću adresne translacije.



Slika 5.18. Scenario za proveru znanja studentima iz fajervol koncepta

## 5.11. Iskustva u radu sa VNLab-om

Virtuelna mrežna laboratorija VNLab je okruženje nastalo na osnovu prikazanog modela. Ovaj model se razlikuje od drugih po tome što su njegova scenarija koncipirana tako da se studenti konektuju na virtuelne mašine koje konfigurišu (VM Access Node), a te virtuelne mašine su deo jedne zajedničke funkcionalne mreže koju čine druge, ranije konfigurisane virtuelne mašine (VM Node). Tako se stvara okruženje u kome je moguće vršiti nastavu na predmetu Računarske mreže bez upotrebe velikog broja virtuelnih mašina. Ova karakteristika omogućuje da se virtuelna mrežna laboratorija nalazi na mrežnom serveru slabijih performansi, bez potrebe za kompleksnim sistemima. Model je realizovan upotrebom softvera koji do sada nije korišćen u sličnim okruženjima (MS Virtual Server 2005 R2).

VNLab se posle upotrebe u nastavi pokazao kao pouzdan i stabilan sistem. U dugom periodu eksploatacije ove laboratorije nije bilo situacija koje bi ometale nastavu. Prosečan broj grupa studenata koji su imali vežbe u toku semestra kretao se između 4 i 5. Časovi su trajali 90 minuta, a u zavisnosti od broja kompletiranih ili urađenih vežbi, nastava je trajala različiti broj nedelja. U poslednje 4 godine upotrebe ove laboratorije broj vežbi se kretao od 7 do 11. Ako se uzme u obzir i uvodni čas, koji je služio da se studentima demonstrira rad sa laboratorijom, može se reći da je laboratorija korišćena u proseku od 8 do 9 nedelja i to 4 do 5 puta nedeljno u trajanju od 90 minuta po času. Stabilnost i pouzdanost laboratorije je potvrđena na osnovu činjenice da je laboratorija za svaki od časova bila spremna za rad (osim u slučajevima kada se uzmu u obzir spoljašnji faktori, kao što je nestanak struje). Pred početak svakog časa preporučeno je restartovanje svih virtuelnih mašina kako bi se obrisala konfiguracija koju su uneli studenti iz prethodnih grupa. Posle višemesečnog rada servera, preporučljivo je i njegov povremeni restart.

Kao što je prikazano u opisu arhitekture modela laboratorije (Poglavlje 5.3.), VNLab server nalazi se odvojen u centralnoj prostoriji gde su i ostali serveri sistema. Studentima je omogućen pristup upotrebom klijentskog softvera razvijenom od strane Microsoft-a na bazi VMRC protokola, i to iz učionica koje su odvojene od lokacije servera. Sav potreban softver je bio

dostupan na VNLab portalu ili serveru datoteka, tako da za svaku laboratoriju nije bila potrebna posebna priprema.

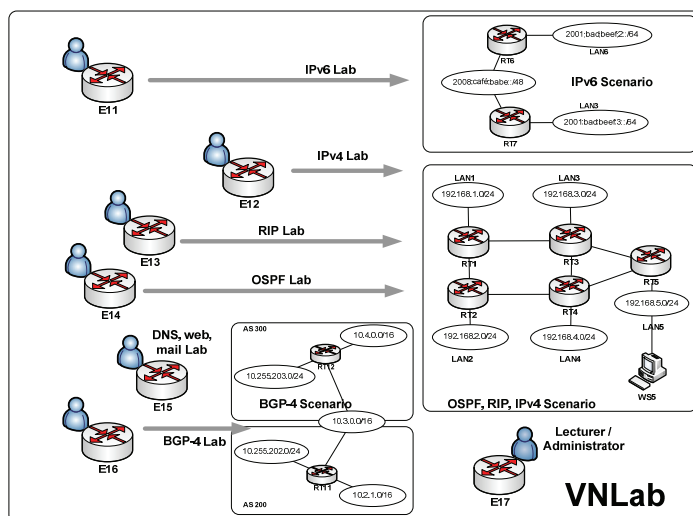
Veza ka serveru je bila pouzdana i nije se prekidala. Jedini „problem“, koji je delovao zbunjujuće na studente, bio je prekid veze usled određenog perioda neaktivnosti. U istom slučaju je dolazilo i do aktiviranja *screen saver*-a, koji je takođe odavao utisak prekida veze. Kao prepreka u radu pokazao se i pritisak na *Windows* taster na tastaturi, koji je Linux virtuelnu mašinu prebacivao na drugi korisnički terminal, onemogućujući studentima unos preko tastature. Takva situacija je u većini slučajeva zahtevala intervenciju predavača. Svaki prekid konekcije usled neaktivnosti, nije imao negativnih posledica na rad studenata jer je ponovna konekcija na istu mašinu pružala mogućnost za nastavak rada, jer se virtuelna mašina, zajedno sa konfiguracijom koju su studenti imali u toku rada, nalazila na serveru.

Funkcionalnost virtuelnih mašina se takođe pokazala zadovoljavajućom. Povremeno su se pojavljivali problemi koji značajnije nisu remetili rad na vežbama. Jedan od najprisutnijih problema je prouzrokovan nemogućnošću da student u toku vežbe dodeli potrebnu IP adresu, npr. 192.168.1.11. Ovaj problem se rešava tako što student tom interfejsu dodeljuje bilo koju drugu slobodnu adresu, npr. 192.168.1.111. i to najčešće uz intervenciju predavača.

Još jedan nedostatak, koji je sprečavao efikasniju upotrebu mrežnih scenarija je mogućnost da se student greškom ili sa namerom konektuje na virtuelnu mašinu namenjenu drugom studentu i tako ometa njegov rad. Višestruka konekcija na istu mašinu nije sprečena, prvenstveno zbog potrebe da se u nekim prilikama omogući predavaču da se konektuje na istu virtuelnu mašinu da bi pomogao studentu u radu ili otklonio neki nedostatak. Neovlašćeni pristup ostaje zapisan u log datoteci.

Međutim, svi navedeni nedostaci nisu imali značajnog uticaja na smanjenje efikasnosti VNLab-a. Svako pogrešno korišćenje virtuelnih mašina od strane studenata koje prouzrokuje manje ili veće nepravilnosti u radu virtuelne mašine, može se otkloniti ili direktnom intervencijom predavača, uz korišćenje standardnih komandi, ili kopiranjem rezervnih kopija imidž datoteka virtuelnih mašina koje se čuvaju sa originalnom konfiguracijom. Kopiranje traje kratko, a posle toga virtuelna mašina se startuje i radi kao i prethodna mašina.

Na kraju, može se dati uporedni prikaz strukture modela virtuelne mrežne laboratorije i načina kako se studenti konektuju na taj sistem u toku rada na različitim scenarijima (Slika 5.19.).



Slika 5.19. Uporedni prikaz rada studenata na različitim scenarijima u okruženju VNLab-a

## **6. Vrednovanje modela virtuelne mrežne laboratorije sa pristupom na daljinu**

Do sada je, u poglavlju 3.2., prikazan veliki broj virtuelnih mrežnih laboratorija. Te virtuelne mrežne laboratorije se odlikuju različitim arhitekturama i tehnologijama na kojima se zasnivaju, a koriste se kako u nastavi tako i za eksperimentisanje, razvoj i testiranje softvera. Veliki broj radova objavljen je u vezi sa ovim laboratorijama. Teme tih radova bile su tehnologija na kojoj su laboratorije zasnovane, načini njene upotrebe i performanse okruženja. Istraživanja o upotrebljivosti ovakvih okruženja u nastavi manje su zastupljena. Nedostaju i istraživanja o njihovoj efikasnosti i upotrebljivosti u nastavi računarskih predmeta na visokoškolskim ustanovama. U ovom poglavlju prikazana je provera efikasnosti VNLab-a.

Prvi pristup u vrednovanju laboratorije je inovativan. Zasniva se na analizi laboratorijskih vežbi koja je bazirana na relaciji nerazberivosti i teoriji grubih skupova. Teorija grubih skupova se koristi da bi se stvorio model zasnovan na if-then pravilima. Počinje se sa tabelarno organizovanim podacima, a koristi se 19 parametara merenja. Ovaj model pomaže da se identifikuju najvažniji delovi laboratorijskih vežbi i ukazuje na glavne razlike u znanju koje poseduju studenti. Izdvajanje najznačajnijih delova vežbi pomaže da se te vežbe unaprede boljim objašnjenjem problema ili modifikacijom sadržaja vežbe. U okviru objašnjenja metoda opisana je i relacija nerazberivosti i teorija grubih skupova, redukcija podataka, sinteza if-then pravila i opis eksperimenta. Analiza pravila i njeni rezultati dati su na kraju sekcije.

Drugi pristup zasniva se na primeni kontrolnih testova. Održana su dva kontrolna testa koja su omogućila analizu nivoa znanja koje su studenti dobili nakon korišćenja VNLab-a. Treći pristup vrednovanju VNLab-a zasniva se na formativnoj proceni znanja studenata. Formativna procena je korišćena da bi se kontinuirano pratio rad studenata, nivo postignutog znanja i da bi se vršila korekcija i adaptacija materijala za laboratorijske vežbe. Sve je to vršeno u cilju prilagođavanja nastave studentima i povećanju efikasnosti čitavog modela, od mrežnih scenarija do pratećeg laboratorijskog materijala. U nastavku su opisana sva tri pristupa.

Na kraju, dat je prikaz upotrebljivosti ove laboratorije upoređivanjem sa preporučenim nastavnim programima za različite obrazovne profile iz oblasti računarskih nauka.

### **6.1. Postojeći primeri vrednovanja virtuelnih laboratorija**

U dosadašnjim istraživanjima, koja su za predmet izučavanja imala virtuelne mrežne laboratorije i njihovu upotrebu u nastavi, akcenat je bio na statističkoj sumarizaciji studentskih rezultata koji su postignuti u radu sa virtuelnom laboratorijom i na osnovu anketa. Proširena istraživanja su odrađena na Ročesterskom institutu za tehnologiju (eng. Rochester Institute of Technology) [LS06]. Pre početka semestra, odabrano je 22 studenta po slučajnom izboru, između približno 80 studenata nižih i viših godina studija koji su pohađali kurs Administracije mreža. Oni su podeljeni u dve grupe: jedna grupa je bila eksperimentalna, a druga kontrolna. Virtuelna laboratorija je bazirana na Windows XP Professional operativnom sistemu sa Remote Desktop Protocol (RDP) klijentima i VMWare Workstation softverom za virtuelizaciju. Neformalni pregled kvalitativnih i kvantitativnih podataka i statistička analiza su korišćenje za evaluaciju rezultata studije. Laboratorijske ocene i ocene sa kursa su analizirane uz pomoć Independent (Samples T-Test) testa i upotrebom SPSS 12.0 softvera za statističku analizu. Da bi se dobile informacije o zadovoljstvu studenata, izvršene su ankete korišćenjem Likertove skale.



Pristup u testiranju relativne efikasnosti laboratorije za mašinstvo i simulaciju sa daljinskim pristupom [NCE07], pristup evaluaciji računarskih adaptivnih testova na mobilnim uređajima [TGE08] i metod evaluacije univerzitetskog softvera za edukaciju [IPP97] i dr. [TIY10] mogu se takođe primeniti u procesu vrednovanja virtuelne mrežne laboratorije.

Prezentovani evaluacioni metodi su obimni, omogućavaju duboku analizu performansi virtuelne laboratorije i znanja koje su studenti stekli. Nasuprot tome, oni nisu pogodni za analizu virtuelnih mrežnih scenarija i njihovog uticaja na znanje studenata. Ovi metodi nisu pogodni ni za identifikaciju ključnih delova prezentovanog gradiva i vežbi koje ga primenjuju, kao i važnosti pojedinih delova u laboratorijskim vežbama.

## 6.2. Vrednovanje modela upotrebom metoda zasnovanog na teoriji grubih skupova

Studenti koji su uključeni u ovaj eksperiment pohađaju nastavu na obrazovnom profilu Informatika [DBB09]. Oni su za vreme redovne nastave kroz laboratorijske vežbe iz predmeta Računarske mreže koristili VNLab. Računarska laboratorija u kojoj se izvode vežbe ima 20 računara sa Windows XP operativnim sistemom. Jedan računar sa bim projektorom je namenjen za rad predavača. Svi računari su povezani na LAN Fakulteta. VNLab server i VNLab portal se fizički nalaze u odvojenoj prostoriji. Laboratorijske vežbe nalaze se na VNLab portalu u tekstualnom obliku (pdf format), i mogu se preuzet ili preko brausera ili preko servera datoteka.

Studenti nisu imali nikakvo prethodno znanje o nastavnoj jedinici koju uče i od njih se očekivalo da o datoj temi uče kroz materijal koji je prezentovan uz vežbu. U okviru pratećeg tekstualnog materijala studenti dobijaju kratko objašnjenje o osnovim mrežnim konceptima koji se koriste u u lekciji, a nakon toga imaju detaljne (korak-po-korak) instrukcije kako da konfigurišu njima dodeljeni ENN ruter (VM AccessNode) za dati scenario. Četiri laboratorijske vežbe se koriste u ovoj evaluaciji. Prva vežba se odnosi na RIP (Routing Information Protocol), a ostale tri na OSPF (Open Shortest Path First) protokol. Tri OSPF vežbe pokrivaju različita scenarija sa ruterima organizovanim u jednu područje (*backbone*), ili više područja (*stub* i *totally stub* tipovi područja). Za svaki scenario, nakon što su studenti završili vežbu prema uputstvima, predavač je proveravao konfiguraciju i funkcionisanje rutera. Ako je ovaj deo vežbe uspešno završen studenti su mogli da nastave sa radom.

Drugi deo vežbe namenjen je za proveru znanja u kojem su studenti morali da odgovore na dva pitanja koja se odnose na rutersku konfiguraciju za isti protokol koji su učili u toku vežbe, ali u drugom mrežnom scenariju. Prilikom rešavanja problema studenti su mogli da se oslone na prethodno stečeno znanje, kao i materijal koji su koristili u završenoj laboratorijskoj vežbi. Studenti su mogli da koriste ENN ruter za praktičan rad i proveru svojih ideja.

Tabela 6.1. – Opis VNLab scenarija

Scenario	Opis scenarija
Lab 1 RIP	Vežba se sastoji od logovanja na ENN ruter uz pomoć VMRC klijenta, adresiranja IP interfejsa, provere tabele rutiranja, provere podrazumevane rute, startovanja zebra i ripd softvera za rutiranje, pristupa softveru za rutiranje upotrebom telnet softvera, konfiguracije RIP protokola u skladu sa datim scenarijom, proverom ruterske konfiguracije, provere tabele rutiranja sa dinamičkim rutama, proverom komande ping i traceroute, simulacijom prekida linka obaranjem interfejsa, proverom tabele rutiranja i komandi ping i traceroute ponovo.
Lab 2 OSPF (no area)	Vežba se sastoji od sličnih koraka kao u vežbi Lab 1 RIP sa razlikom u pokretanju

	različitog softvera (Zebra, ospfd) i konfiguracijom OSPF protokola u skladu sa scenarijom koji ima jednu (backbone) područje.
Lab 3 OSPF (stub areas)	Vežba se sastoji od sličnih koraka kao u vežbi Lab 2 OSPF sa razlikom u scenariju koji ima dva nova područja (backbone, area 1, area 2). Na kraju vežbe nema simulacije prekida linka. Područja 1 i 2 su tipa stub.
Lab 3 OSPF (totally stub areas)	Vežba se sastoji od sličnih koraka kao u vežbi Lab 2 OSPF sa razlikom u scenariju koji ima dve nove područja (backbone, area 1, area 2). Na kraju vežbe nema simulacije prekida linka. Područja 1 i 2 su tipa totally stub.

U toku provere studentskih odgovora koji su rađeni posle svake završene vežbe (na kraju svakog časa), tri ili četiri ključne aktivnosti su proveravane za svaki scenario. Za RIP scenario ključne aktivnosti su provera ispravnosti komandi za RIP protokol, pravilno adresiranje mreža u pogledu ispravnog formata adrese ali i po tačnosti navedenih mreža koje su asocirane sa RIP procesom rutiranja. Ključne aktivnosti za OSPF vežbe su slične, ali sa dodatkom nekih specifičnosti za taj protokol. To su ispravno navođenje i konfigurisanje područja, pasivnih interfejsa i konfiguracija cena putanje. U tabeli 6.1. prikazana su VNLab scenarija sa opisom važnih koraka (ključnih aktivnosti) koje su proveravane. Ti važni koraci zvaće se atributi u procesu analize podataka. Vrednost ovih atributa je 1 ili 2. Vrednost 2 predstavlja tačno urađene korake (TRUE) a vrednost 1 predstavlja netačno urađene korake (FALSE).

Poslednji atribut u tabeli je završna ocena za svakog studenta. Ocena se kreće u rasponu od kategorije 1 do kategorije 6 i predstavlja znanje studenata.

Table 6.2. VNLab scenarija i opis važnih koraka description

Atribut	Naziv scenarija	Opis koraka	Koeficijent korelacije sa atributom a19 (konačnom ocenom)
a1	RIP	Ispravna sintaksa komandi RIP protokola	0.116071
a2	RIP	Ispravno adresiranje mreže	0.546896
a3	RIP	Ispravna asocijacija sa RIP procesom rutiranja	0.389249
a4	OSPF (bez područja)	Ispravna sintaksa komandi OSPF protokola	0.355808
a5	OSPF (bez područja)	Ispravna asocijacija sa OSPF procesom rutiranja	0.460681
a6	OSPF (bez područja)	Ispravna konfiguracija područja	0.059082
a7	OSPF (bez područja)	Ispravna konfiguracija cene putanje	0.279762
a8	OSPF (bez područja)	Ispravna konfiguracija pasivnog interfejsa	0.123025
a9	OSPF (stub područja) 1	Ispravna asocijacija sa OSPF procesom rutiranja	0.438995
a10	OSPF (stub područja) 1	Ispravna konfiguracija područja	0.236403

a11	OSPF (stub područja) 1	Ispravna konfiguracija cene putanje	0.274685
a12	OSPF (stub područja) 1	Ispravna konfiguracija pasivnog interfejsa	0.565297
a13	OSPF (stub područja) 2	Ispravna konfiguracija područja	0.13858
a14	OSPF (stub područja) 2	Ispravna asocijacija sa OSPF procesom rutiranja	0.291918
a15	OSPF (stub područja) 2	Ispravna konfiguracija područja	0.574351
a16	OSPF (stub područja) 2	Ispravna konfiguracija cene putanje	-0.133631
a17	OSPF (stub područja) 2	Ispravna konfiguracija pasivnog interfejsa	0.324677
a18	OSPF (stub područja) 2	Ispravna konfiguracija područja	0.309739
a19	OSPF (stub područja) 2	Finalna ocena	1.0

Metod zasnovan na teoriji grubih skupova odabran je za analizu podataka [Br08, ČS02] jer se na taj način može generisati skup „if-then“ pravila iz tabelarno organizovanih podataka. „Značajna prednost metoda koji generišu stabla odluke ili skupove if-then pravila je u tome što su takvi modeli veoma pregledni i laki za tumačenje, a njihovi rezultati lako objašnjivi“ [KØ99].

### 6.2.1. Relacija nerazberivosti i teorija grubih skupova

Teorija grubih skupova je razvijena od strane Zdzislava Pavlaka (Zdzisław Pawlak) ranih 1980-ih godina [PGS95, PS07]. Ovaj pristup se pokazao kao veoma efikasan za analizu podataka iz različitih domena.

Relacija nerazberivosti je matematička osnova za teoriju grubih skupova. „Filozofija grubih skupova je zasnovana na pretpostavci da je za svaki objekat univerzuma moguće asociirati neke informacije (podatke, znanje)...“ [Paw97]. Tako, objekti koji su karakterisani istom informacijom su nerazberivi u pogledu dostupnih informacija o njima.

Formalno, kao u [GBS98], neka je univerzum  $U$  konačni skup objekata,  $Q = \{q_1, q_2, \dots, q_m\}$  je konačan skup atributa,  $V_q$  je domen atributa  $q$  i  $V = \bigcup_{q \in Q} V_q$ . Informacioni sistem je četvorka  $S = \langle U, Q, V, f \rangle$  gde je  $f = U \times Q \rightarrow V$  funkcija takva da zadovoljava  $f(x, q) \in V_q$  za  $q \in Q$  i  $x \in U$ , i zove se informaciona funkcija. Svaki objekat univerzuma je opisan pomoću vektora:  $Des_q(x) = [f(x, q_1), f(x, q_2), \dots, f(x, q_m)]$ , za  $x \in U$ . Za svaki neprazan podskup atributa  $P$  asociirana je relacija nerazberivosti relacija na  $U$ , označena sa  $I_P$ :

$$I_P = \{(x, y) \in U \times U : f(x, q) = f(y, q), \forall q \in P\} \quad (1)$$

Ova relacija je relacija ekvivalencije (refleksivna, simetrična i tranzitivna). Familija svih klasa ekvivalencije od  $I_P$  je označena sa  $U|I_P$ , a klasa koja sadrži element  $x$  sa  $I_P(x)$ .

Jednostavan primer informacionog sistema u tabelarnoj formi prikazan je u tabeli 6.3. Ovaj primer je zasnovan na sličnom primeru datom u [Øhr99]. U tabeli 6.3, postoji univerzum od 6 objekata  $U = \{x_1, \dots, x_6\}$ . Svaki objekat je prikazan sa četiri atributa: RIP protocol command syntax (RIPCS), Network address check (NAC), Network association (NA) i Final grade (FG). Objekat  $x_1$  je opisan na sledeći način: RIPCS=excellent, NAC=correct, NA=poor, FG=low; objekat  $x_2$  je opisan na sličan način: RIPCS=acceptable, NAC=wrong, NA=poor, FG=high itd. Objekti koji imaju isti opis su nerazberivi u odnosu na vrednosti njihovih atributa. Relacija nerazberivosti uzrokuje podelu univerzuma u blokove nerazberivih objekata nazvanih elementarnim skupovima.

Tabela 6.3. Jednostavan primer informacionog sistema

Objekat	RIP protocol command syntax (RIPCS)	Network address check (NAC)	Network announcement (NA)	Final grade (FG)
$x_1$	excellent	correct	poor	Low
$x_2$	acceptable	wrong	poor	High
$x_3$	acceptable	wrong	poor	Low
$x_4$	tolerable	wrong	poor	High
$x_5$	acceptable	correct	good	High
$x_6$	excellent	wrong	good	Low

Ako je  $P = \{RIPCS, NAC, NA\}$  tada, prema (1), imamo:

$$I_P = \{\{x_1, x_1\}, \{x_2, x_2\}, \{x_2, x_3\}, \{x_3, x_2\}, \{x_3, x_3\}, \{x_4, x_4\}, \{x_5, x_5\}, \{x_6, x_6\}\},$$

$$U|I_P = \{\{x_1\}, \{x_2, x_3\}, \{x_4\}, \{x_5\}, \{x_6\}\}.$$

Dalje, neka je  $X$  neprazan skup konačnog univerzuma  $U$  i  $\emptyset \neq P \subseteq Q$ . Skup  $X$  se može aproksimirati uz pomoć P-donje (2) i P-gornje (3) aproksimacije  $X$ :

$$\underline{P}(X) = \{x \in U : I_P(x) \subseteq X\} \quad (2)$$

$$\overline{P}(X) = \bigcup_{x \in X} I_P(x) \quad (3)$$

P-granični region za  $X$  je označen sa  $Bn(X)$ :

$$Bn(X) = \overline{P}(X) - \underline{P}(X) \quad (4)$$

Atributi iz skupa  $P$  se nazivaju uslovni atributi, a atribut FG je atribut odluke. Informacioni sistem sa definisanim skupom uslovnih atributa i atributom odluke zove se sistem odluke.

Ako se razmotri slučaj kada skup  $X$  sadrži samo one elemente kod kojih atribut odluke FG ima vrednost *low*:  $X = \{x_1, x_3, x_6\}$ , prema tabeli 6.3 može se izvršiti aproksimacija skupa  $X$  upotrebom samo informacija sadržanih u  $P$  (skup uslovnih atributa) upotrebom P-donje (2) i P-gornje (3) aproksimacije  $X$ :  $\underline{P}(X) = \{x_1, x_6\}$  i  $\overline{P}(X) = \{x_1, x_2, x_3, x_6\}$ .

P-granični region (4) za  $X$  je:  $Bn(X) = \{x_2, x_3\}$ .

Elementi P-donje aproksimacije skupa  $X$  sigurno pripadaju skupu  $X$ , ali se može zaključiti da P-granični skup čini "neizvestan region" – ništa se sa sigurnošću ne može reći o elementima koji pripadaju skupu  $X$ . Može se primetiti da objekti  $x_2$  i  $x_3$  (P-granični region za  $X$ ) imaju iste vrednosti uslovnih atributa, ali različite vrednosti atributa odluke.

### 6.2.2. Redukcija podataka i sinteza if-then pravila

Veoma bitan korak u procesu analize podataka je izdvajanje samo onih uslovnih atributa koji mogu da očuvaju relaciju nerazberivosti i aproksimaciju skupa. Otklonjeni uslovni atributi su suvišni zato što njihovo izostajanje ne može pogoršati klasifikaciju.

Neka je  $\emptyset \neq P \subseteq Q$  i  $a \in P$ . Atribut  $a$  je suvišni u  $P$  ako je  $I_P = I_{P-\{a\}}$ . Ako je  $R = \{\text{RIPCS}, \text{NAC}\}$ ,  $S = \{\text{RIPCS}, \text{NA}\}$ , i  $T = \{\text{NAC}, \text{NA}\}$ , (videti tabelu 6.3) tada je  $I_R = I_P$  i  $I_S = I_P$  a  $I_T \neq I_P$ . To znači da su  $R$  i  $S$  redukti od  $P$ , dok  $T$  to nije. Atribut RIPCS je neophodan atribut, ali atributi NAC i NA se međusobno mogu promeniti.

Obično postoji nekoliko redukt skupova uslovnih atributa. Pronalaženje redukta sa minimalnom kardinalnošću atributa među svim reduktima jeste NP-težak problem (eng. *NP-hard*). Proračun svih redukata je veoma kompleksan, ali u većem broju situacija nije potrebno pronaći sve redukte, nego samo neke od njih.

Izraz  $a = v$ , gde  $a$  predstavlja atribut, a  $v$  je vrednost atributa koja se naziva deskriptor. Ako se istraže pravila u obliku: IF  $\alpha$  THEN  $\beta$ , tada  $\alpha$  predstavlja konjukciju (AND logički operator) deskriptora koji uključuju samo attribute nekog redukta (pravila), a  $\beta$  (sledbenik) predstavlja deskriptor  $d = v$ , gde je  $d$  atribut odluke.

Metod generisanja skupa pravila je rađen prema [KØ99, PGS95, Paw97, GBS98, Øhr99], i sastoji se od sledećih koraka:

1. diskretizacija (kvantizacija) realnih vrednosti atributa (postoji veliki broj različitih tehnika),
2. izračunavanje redukta (više različitih tehnika) i
3. sinteza pravila koja se dobija preklapanjem redukta preko tabele odluke i očitavanjem vrednosti.

Ako koristimo redukt  $R=\{RIPCS, NAC\}$ , prema tabeli 6.3. Imamo pravila:

1. IF RIPCS=excellent AND NAC=correct then FG=low      pravilo podržava  $x_1$
2. IF RIPCS=acceptable AND NAC=wrong then FG=high      pravilo podržava  $x_2$
3. IF RIPCS=acceptable AND NAC=wrong then FG=low      pravilo podržava  $x_3$
4. IF RIPCS=tolerable AND NAC=wrong then FG=high      pravilo podržava  $x_4$
5. IF RIPCS=acceptable AND NAC=correct then FG=high      pravilo podržava  $x_5$
6. IF RIPCS=excellent AND NAC=wrong then FG=low      pravilo podržava  $x_6$

Moguća je dalja redukcija skupa pravila. Pravila 2 i 3 se mogu spojiti tako da formiraju jedno pravilo tzv. nekonzistentno (nedeterminističko) pravilo:

IF RIPCS=acceptable AND wrong then FG=high OR FG=low.

Pravila podržana objektima koji pripadaju donjoj aproksimaciji se nazivaju univokalna ili konzistentna pravila, a pravila koja su podržana objektima koji pripadaju graničnom regionu su nekonzistentna.

### 6.2.3. Eksperiment

Eksperiment je sproveden na realnim podacima koje sadrže 18 izmerenih vrednosti (uslovnih atributa) i jedan atribut odluke (Final grade ili konačna ocena), kao što je prikazano u tabeli 6.2. Vrednosti uslovnih atributa su binarne, dok je vrednost atributa odluke u rasponu od 1 do 6. To znači da diskretizacija nije potrebna. Podaci sadrže 55 objekata (vrsta) koji predstavljaju 55 studenata i 19 kolona, koje predstavljaju atribute (18 uslovnih i 1 atribut odluke).

Eksperiment je urađen u softveru Rosetta - A Rough Set Toolkit for Analysis of Data. Rosetta je razvijena kao deo zajedničkog istraživanja institucija Knowledge Systems Group, na odseku za Računarske i informacione nauke (eng. Department of Computer and Information Science) na NTNU, Norveška, i grupi Logic Group, sa Instituta za matematiku univerziteta u Varšavi, Poljska [Øhr99].

Koeficijent koleracije  $r$  između atributa  $a_{19}$  (Final grade) i bilo kog drugog atributa se može proračunati kao:

$$r = \frac{\sum_{i=1}^K (x_i - m_1)(y_i - m_2)}{\sqrt{\sum_{i=1}^K (x_i - m_1)^2 \sum_{i=1}^K (y_i - m_2)^2}}, \text{ gde je } K = 55.$$

Veličina  $m_1$  je srednja vrednost za atribut odluke, a  $m_2$  je srednja vrednost izabranog uslovnog atributa (Tabela 6.2). Koeficijent korelacije ima vrednosti u intervalu  $[-1, 1]$ . Kada je  $r = -1$  tada postoji jaka negativna korelacija između atributa. Kada je  $r = 1$  tada postoji jaka pozitivna korelacija, a kada je  $r = 0$  tada korelacija ne postoji. Kada postoji jaka korelacija to ukazuje na

linearnu zavisnost. Kao što je prikazano u tabeli 6.2., koeficijenti korelacije nisu blizu vrednosti 1, tako da ne postoji jaka linearna korelacija između atributa *a19* i ostalih atributa.

Eksperiment je sproveden na sledeći način:

Prvo su redukti izračunati upotrebom Exhaustive algoritma [Øhr99]. Taj algoritam je implementiran u programskom paketu Rosetta i omogućuje izračunavanje višestrukih skupova redukata.

Drugo, if-then pravila su sintetizovana upotrebom softvera Rosetta na osnovu prethodno izračunatih skupova redukata.

#### 6.2.4. Analiza podataka

Za proračun redukt skupova korišćen je Exhaustive algoritam koji je implementiran u programu Rosetta. Tako su dobijena dva redukt skupa. Svaki od njih ima 10 atributa:

Redukt skup 1: {a1, a3, a4, a5, a6, a7, a9, a10, a15, a18} i

Redukt skup 2: {a1, a3, a4, a5, a6, a7, a10, a14, a15, a18}.

Rosetta je generisala 70 pravila upotrebom oba redukt skupa. Postoje 4 nekonzistentna pravila, tj. pravila sa OR operatorom u THEN delu. Svako pravilo karakteriše parametar LHS koji ukazuje na broj objekata koji podržavaju IF deo pravila. Pravila sa većom LHS vrednošću su „jača“. Jaka pravila su prikazana u tabeli 6.4.

Pravila 1, 2 i 5 su generisana upotrebom redukt skupa 2, a pravila 3, 4 i 6 su generisana upotrebom redukt skupa 1. Očigledno je da nema nekonzistentnih pravila u tabeli 6.4.

Tabela 6.4.

If – Then pravila sintetizovana softverom Rosetta

	<b>If</b>	<b>Then</b>	<b>LHS podrška</b>
1.	a1(2) AND a3(2) AND a4(2) AND a5(1) AND a6(2) AND a7(2) AND a10(2) AND a14(1) AND a15(2) AND a18(2)	a19(5)	7
2.	a1(2) AND a3(2) AND a4(2) AND a5(2) AND a6(2) AND a7(2) AND a10(2) AND a14(2) AND a15(2) AND a18(2)	a19(6)	7
3.	a1(2) AND a3(2) AND a4(2) AND a5(1) AND a6(2) AND a7(2) AND a9(1) AND a10(2) AND a15(2) AND a18(2)	a19(5)	7
4.	a1(2) AND a3(2) AND a4(2) AND a5(2) AND a6(2) AND a7(2) AND a9(2) AND a10(2) AND a15(2) AND a18(2)	a19(6)	7
5.	a1(2) AND a3(2) AND a4(2) AND a5(1) AND a6(2) AND a7(2) AND a10(2) AND a14(2) AND a15(2) AND a18(2)	a19(6)	4
6.	a1(2) AND a3(2) AND a4(2) AND a5(1) AND a6(2) AND a7(2) AND a9(2) AND a10(2) AND a15(2) AND a18(2)	a19(6)	4

Pravilo 1 iz tabele 6.4. se može interpretirati na sledeći način:

Ako je a1 tačno i a3 tačno i a4 tačno i a5 tačno i a6 tačno i a7 tačno i a10 tačno i a14 je netačno i a15 tačno i a18 tačno, tada je finalna ocena u kategoriji 5. Analizom pravila može se pretpostaviti da korišćeni atributi u pravilima ukazuju na zadatke, tako da prethodno pravilo možemo da predstavimo na sledeći način: Ako je student tačno uradio deo zadatka a1 i tačno uradio deo zadatka a3 itd., ... tada je studentova konačna ocena u kategoriji 5.

Daljom analizom ovih pravila dolazi se do jasnih indikatora koji su glavni problemi za studente koje oni imaju u rešavanju zadataka i koji delovi zadatka utiču na dobijanje nižih ocena (npr. ocene kategorije 4 i niže). Dolazi se do zaključka da studenti imaju najviše problema sa tačnom asocijacijom odgovarajućih mreža sa RIP i OSPF procesom rutiranja, ali takođe i u konfiguraciji područja u scenariju za OSPF protokol sa višestrukim područjima (Tabela 6.2.).

Analizom skupa redukta postaje očigledno da postoji samo jedna razlika: redukt skup 1 sadrži atribut a9, a ne sadrži a14, dok redukt skup 2 sadrži atribut a14, ali ne sadrži atribut a9. To znači da se atributi a9 i a14 mogu međusobno zameniti.

Takođe, može se zaključiti da atributi: a2, a8, a11, a12, a13, a16, a17 nisu potrebni za određivanje konačne ocene, tako da se aktivnosti koje odgovaraju tim atributima mogu isključiti iz ocenjivanja u daljem radu ili se može smanjiti broj bodova koje one donose pri ocenjivanju.

Sa stanovišta predavača rezultati se mogu komentarisati na sledeći način: analizirane vežbe su dovoljno dobre da objasne studentima kako da koriste odgovarajuće mrežne adrese, pasivne interfejsa, cene putanja i izvrše konfiguraciju područja i podešavanje njihovih tipova kada je potrebno konfigurisati rutere. Neki od ovih koncepata su razumljivi iz prve vežbe (npr. pasivni interfejsi), a neki su razumljivi iz druge po redu vežbe (npr. cena putanje).

Takođe je očigledno da se „if-then pravila“ iz tabele 6.4. mogu koristiti da se shvati koje su vrednosti uslovnih atributa potrebne u slučajevima kada je konačna ocena visoka (kategorija 5 ili kategorija 6). Analizom se dolazi do zaključka da je za postizanje najviše ocene potrebno uraditi većinu koraka koji su predviđeni u zadatku, kao i koji su to koraci.

Iz tabele 6.4. može se zaključiti da je za najvišu ocenu potrebno da se urade sledeći zadaci (atributi): a1, a3, a4, a6, a7, a10, a15 i a18. Kada je konačna ocena iz kategorije 5 (Pravilo 1 i Pravilo 3), vrednosti atributa a5 su FALSE u oba pravila. Može se zaključiti da je tačnost zadatka ili atributa a5 presudna za studenta u pokušaju da postigne najveću ocenu, tj. ocenu iz kategorije 6.

Analizom svih 70 pravila koje je za ove podatke generisao softver Rosetta mogu se izvesti sledeći zaključci: Prvo, analiza pravila daje preciznu indikaciju u čemu je najveća razlika u znanju između studenata koji su postigli najviše ocene i onih koji to nisu. To nam omogućava da usavršimo i poboljšamo vežbe sa kojima studenti imaju najviše problema i pokušamo da ih drugačije i bolje koncipiramo i osmislimo u cilju podizanja efikasnosti virtuelne laboratorije. Dodatno, ovaj metod nam daje mogućnost da izolujemo one delove vežbi koji su lako razumljivi svim studentima i da ih isključimo iz ocenjivanja ili samo da smanjimo njihov uticaj na konačnu ocenu tako što će se broj poena koje student dobija ako uspešno reši taj deo zadatka smanjiti. To pomaže u stvaranju mnogo realističnijih modela za vrednovanje studentskog znanja.



Na posletku, ovaj metod daje nam mogućnost da shvatimo koje su uobičajene greške koje studenti prave u slučajevima kada dobijaju niže ocene (kategorije 4 ili 3) i da saznamo koji deo vežbi treba da se unapredi da bi se unapredio nivo znanja većine studenata.

Analizirani metod je još efikasniji u slučajevima kada se obrađuje veća količina informacija, npr. veći broj studenata čije se znanje proverava ili veći broj generacija studenata za vreme nekoliko kurseva.

### 6.3. Vrednovanje modela upotrebom kontrolnih testova

Drugi pristup koji je korišćen da se utvrdi nivo znanja studenata koji je postignut u toku upotrebe ove laboratorije su kontrolni testovi u toku semestra. U prvom delu ove sekcije prezentovan je test koji je izvršen u toku letnjeg semestra 2010. god. Test, nazvan Kontrolni test br. 1., je najavljen unapred, a održan je posle šest završenih laboratorijskih vežbi.

#### 6.3.1. Kontrolni test br. 1

Test se sastojao iz tri zadatka. Prvi zadatak bio je zasnovan na konfiguraciji rutera u okviru OSPFv2 mrežnog scenarija (postupak konfiguracije u ovom scenariju sličan je kao i laboratorijska vežba opisana u prethodnom delu teksta - Poglavlje 5.7.). Drugi zadatak se odnosi na konfiguraciju BGP-4 protokola (sličan vežbi opisanoj u poglavlju 5.8). Treći zadatak bio je fokusiran na IPv6 adresiranje i statiko rutiranje (vežba opisana u Poglavlju 5.5.). Detaljan opis zadataka i postupaka koje su potrebne za njihovo izvršenje dati su u tabeli 6.5. Test je izvršen u pismenoj formi. Prikaz mrežnih scenarija koji su korišćeni u zadacima dat je na slikama 6.1., 6.2 i 6.3. Studentima je pored zadataka dat i spisak komandi koje su potrebne za njihovo rešavanje. Studenti su morali da zajedno sa komandama koriste i odgovarajuće parametre koje su sami određivali u skladu sa traženom konfiguracijom i sa ciljem da ruter učine funkcionalnim delom mreže.

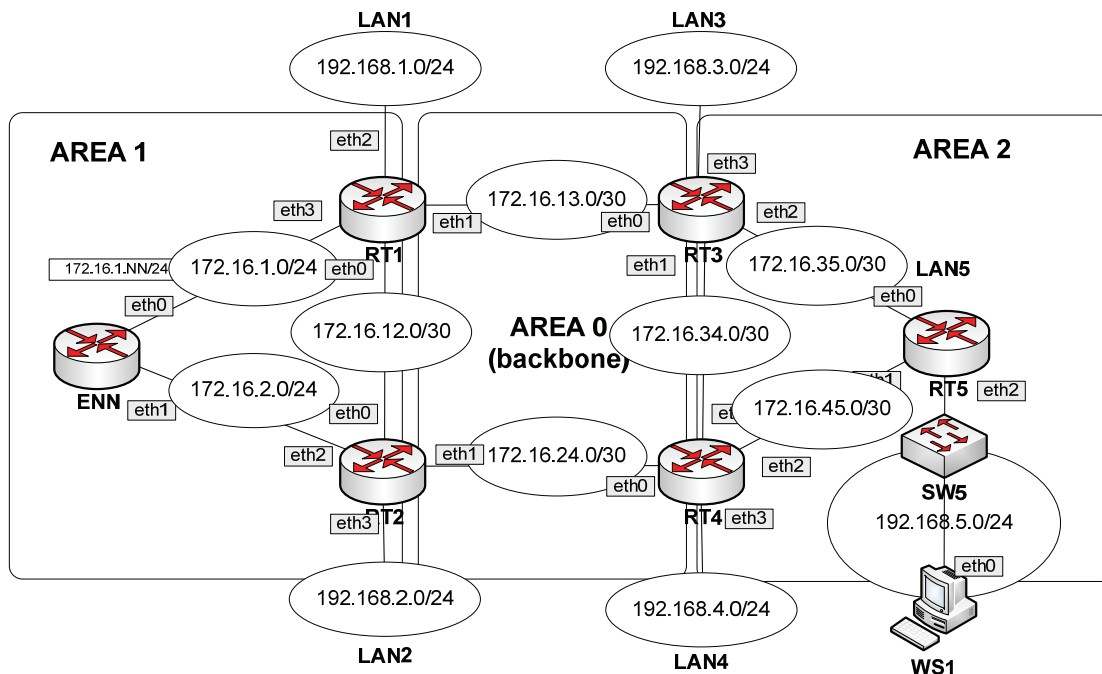
Tabela 6.5. – Detaljan opis zadataka u kontrolnom testu br. 1.

Opis zadatka	Required steps in the exercises
Zadatak 1 - OSPFv2	1.1 IPv4 adresiranje mrežnih interfejsa 1.2. Konfiguracija pasivnih interfejsa 1.3. Asocijacija mreža sa OSPF ruting procesom 1.4. Konfiguracija područja 1.5 Konfiguracija cene putanje
Zadatak 2 – BGP-4	2.1. IPv4 adresiranje mrežnih interfejsa 2.1. Konfiguracija BGP ruting procesa 2.3. Specifikacija lokalnim mreža za AS 2.4. Dodavanje IP adrese susednog rutera
Zadatak 3 - IPv6 adresiranje i statičko rutiranje	3.1. Dodavanje Ipv6 adresa 3.2. Dodavanje Ipv6 ruta 3.3. Dodavanje gateway-a za rute

Forma testa je prikazana u nastavku:

#### Zadatak 1. – Kontrolni test br. 1

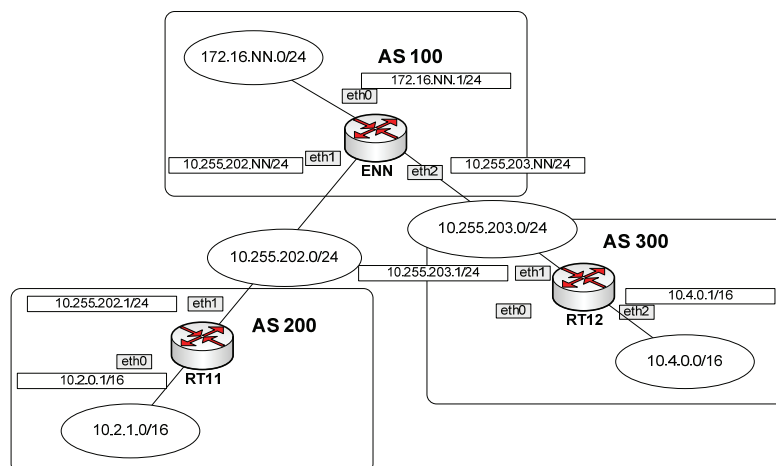
Napisati konfiguraciju rutera **RT2** prikazanog na slici (Slika 6.1). Potrebno je adresirati interfejsе, pokrenuti GNU Zebra softver i OSPF protokol, odrediti pasivni interfejs, oglasiti sve mreže koje su direktno povezane sa ruterom koji konfiguriramo, podesiti tip područja na **stub no-summary** i na kraju, definisati cenu putanje (eng. *path cost*) za sve interfejsе za koje je to potrebno.



Slika 6.1. Mrežni scenario za OSPF protokol

## Zadatak 2. – Kontrolni test br. 1

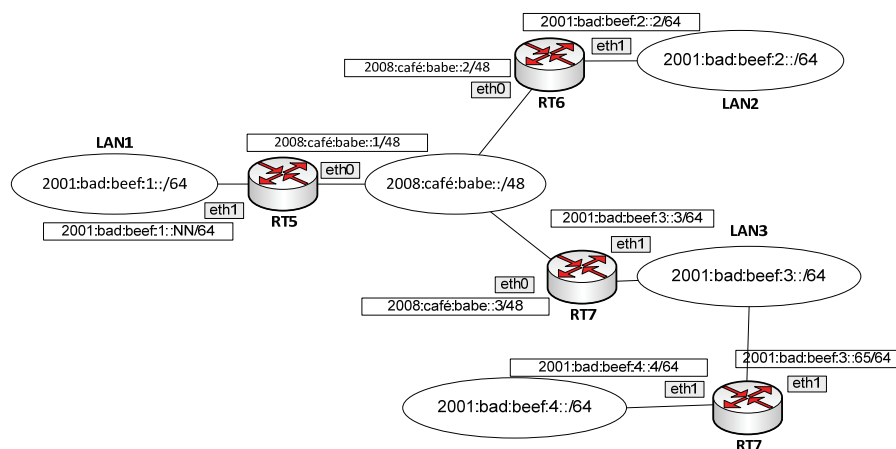
Napisati konfiguraciju rutera **RT12** prikazanog na slici (Slika 6.2.). Potrebno je adresirati interfejsе, pokrenuti GNU Zebra softver i BGP protokol, odrediti AS, oglasiti sve mreže koje su direktno povezane sa ruterom koji konfiguriramo i na kraju, definisati neighbor router sa odgovarajućim AS.



Slika 6.2. Mrežni scenario za BGP-4 protokol

### Zadatak 3. - - Kontrolni test br. 1

Postaviti rutu na ruteru RT7 za mrežu 2001:bad:beef:2::/64 koja je prikazana na slici 6.3.



Slika 6.3. Mrežni scenario za IPv6 mreže

Efikasnost laboratorije je analizirana sa dve grupe studenata. Jedna grupa je brojala 34, a druga 14 studenata. Obe grupe se sastoje od studenta osnovnih studija sa odseka za Informatiku i računarstvo, Tehničkog Fakulteta “Mihajlo Pupin” - Zrenjanin. Veću grupu čine studenti smera Inženjer Informatike, a druga grupa se sastoji od studenata smera Poslovna informatika. Ukupno je u testu učestvovalo 48 studenata. Maksimalan broj poena koji je mogao da se osvoji 40. Prezentacija postignutih rezultata (procenat osvojenih bodova) data je u tabeli 6.6.

Tabela 6.6. - Postignuti rezultati kontrolnog testa br. 1

Grupa	Broj studenata	Prosek osvojenih poena	Procenat osvojenih poena
Grupa 1	34	31.88	80%
Grupa 2	14	28.64	72%
<b>Ukupno</b>	<b>48</b>	<b>30.93</b>	<b>77%</b>

Postignuti uspeh studenata, prikazan je u tabeli 6.7., a njihovo grupisanje dato je u skladu sa rangiranjem u radovima [GCJ11a, GCJ11b, CG09]. U tim radovima je prezentovan uspeh studenata u radu sa virtuelnom mrežnom laboratorijom Kivans, a njihovi rezultati testa i ocene koje su studenti dobili rangirani su u poređenju sa standardima za rangiranje u Španiji (Spanish Academic Grading System) i SAD (US GPA). Da bi se dobio što univerzalniji prikaz postignutih znanja studenata, izvršena je sinteza dva standarda za rangiranje [GCJ11a]. Tako su studenti podeljeni u četiri grupe. Prva grupa studenata su oni studenti koji su osvojili 85% do 100% bodova, druga grupa 65% do 85% bodova, treća grupa 50% do 65% bodova i četvrta grupa ispod 50% bodova.

Tabela 6.7. - Uspeh studenata na kontrolnom testu br. 2

Uspeh na testu / grupa	Grupa 1	Grupa 2	Ukupno	Procenat osvojenih bodova
85%-100%	12	2	16	29%
65%-85%	17	9	26	54%
50%-65%	4	2	6	13%
Ispod 50%	1	1	2	4%
<b>Ukupno studenata</b>	<b>34</b>	<b>14</b>	<b>48</b>	

Analizom rezultata kontrolnog testiranja, VNLab se pokazao kao efikasno okruženje za nastavu, jer je prosečan broj postignutih poena 30.93 u poređenju sa maksimalnih 40. Procenat ukupno osvojenih poena obe grupe studenata je 77% (Tabela 6.6). Procenat studenata koji su postigli prelaznu ocenu iznosi 96% (Tabela 6.7).

Dalja detaljna analiza pokazala je da su najlošije urađeni zadaci bili zadaci u vezi sa BGP-4 protokolom, gde je samo 35% studenata u potpunosti izvršilo zadatak. Slična situacija je sa IPv6 statičkim rutiranjem gde je uspešnost 51%. Najveći problem u rešavanju zadataka predstavljala je definicija mrežnog prolaza (eng. *gateway*). Ovakve i slične analize pomažu da se otkriju delovi laboratorijskih vežbi koje nisu u potpunosti razumljive studentima i rezultati ovih analiza takođe mogu da se koriste u poboljšanju vežbi.

### **6.3.2. Kontrolni test br. 2**

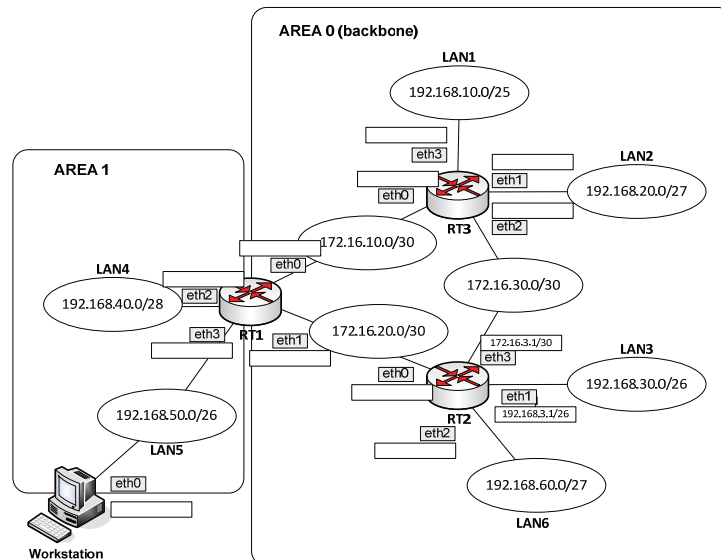
Sledeći rezultati su postignuti u kontrolnom testu koji je izvršen u toku letnjeg semestra 2011. god. Test, nazvan kontrolni test br. 2., najavljen je 2 dana ranije. Za razliku od prethodnog testa, postavke zadatka nisu slične kao u laboratorijskim vežbama. Pre testiranja odrađeno je 7 laboratorijskih vežbi.

U testiranju su učestvovali studenti osnovnih studija odseka za Informatiku i računarstvo, Tehnolog Fakulteta "Mihajlo Pupin" - Zrenjanin sa III godine i smerova: Inženjer Informatike – Grupa 2 (30 studenata), Poslovna informatika – Grupa 3 (22 studenta) i Profesori informatike – Grupa 4 (16 studenata). Pored njih u testiranju su učestvovali studenti II godine studija obrazovnog profila Informacione tehnologije (najnoviji akreditovani smer) – Grupa 1 (58 studenta). Ukupno je testirano 126 studenata. Bez obzira na pripadnost različitim obrazovnim profilima, svi studenti su pohađali predmet Računarske mreže istog sadržaja.

Opis datih zadataka dati su u nastavku kao i odgovarajuća scenarija (Slika 6.4. i 6.5.).

#### **Zadatak 1. – Kontrolni test br. 2**

- a) Napisati konfiguraciju rutera RT3 na slici 1. potrebnu za konfigurisanje OSPF protokola. Cena putanje na interfejsu eth0 je 15, a na interfejsu eth1 je 20. Area 1 je stub no-summary, a Area 0 je backbone area.
- b) Adresirati sve potrebne interfejse (dodeliti potrebne IPv4 adrese prema konfiguraciji na slici).
- c) Na ruteru RT1 na slici 1. postaviti fajervol pravilo koje zabranjuje pristup svim računarima sa spoljne mreže na server 192.168.5.2, a dozvoljava pristup na njegov veb i mail server računarima sa mreže 192.168.2.0/27.



Slika 6.4. Mrežni scenario za OSPF mreže i fajervol scenario u kontrolnom testu br. 2

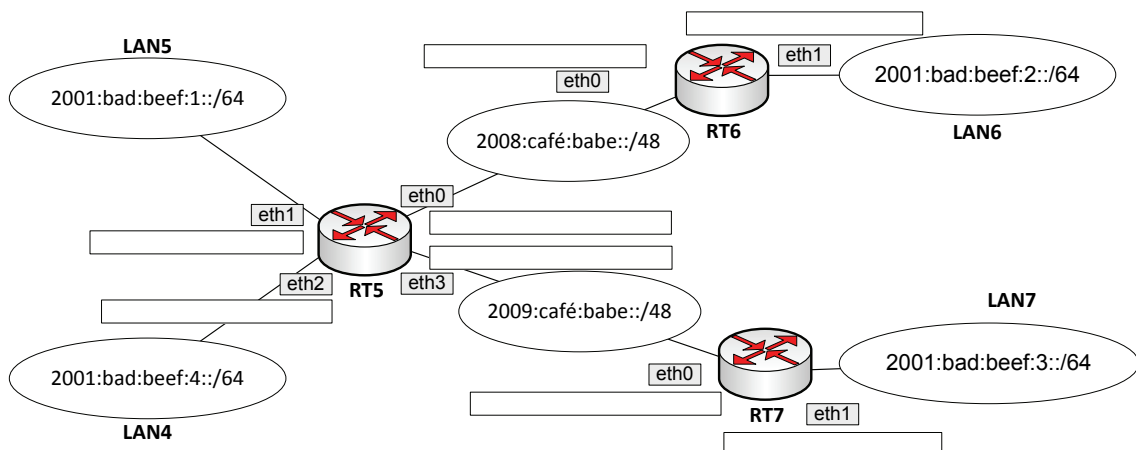
### Zadatak 2. – Kontrolni test br. 2

- Adresirati sve potrebne interfejsse (dodeliti potrebne IPv6 adrese prema konfiguraciji na slici).
- Uneti sve potrebne rute za IPv6 mreže na ruteru RT5 na slici 2 (Slika 6.5.).
- Zbog konfiguracije NDP (Neighbor discovery protocol) asociirati IPv6 adresu interfejsa eth0 rutera RT7 na slici 2. (Slika 6.5.) sa njegovom MAC adresom AA:01:C2:D3:74:05. Objasniti šta je MAC adresa.
- Navesti koje su od navedenih IPv6 adresa neispravne i objasniti zašto

```

2001:bad:beef:beefe::1
2001::bad:beef: F10::1
1:bad:beef:bef::72A
1234:1234:AAAA:BBBB:CCCC:2234:2356:23AD:3455
2001:0db8:85a3:0000:0000:8a2e:0370:7334

```



Slika 6.5. Mrežni scenario za IPv6 mreže u kontrolnom testu br. 2

U ovom slučaju rezultati su bili lošiji. Prosečan broj postignutih poena je 16.08 u poređenju sa maksimalnih 30. Procenat ukupno osvojenih poena svih studenata je 54%. Broj ostvarenih poena po grupama dat je u tabeli (Tabela 6.8).

Tabela 6.8. – Postignuti rezultati kontrolnog testa br. 2

<b>Grupa</b>	<b>Broj studenata</b>	<b>Prosek osvojenih Poena</b>	<b>Procenat osvojenih poena</b>
Grupa 1	58	14.64	49%
Grupa 2	30	22.76	76%
Grupa 3	22	12.50	42%
Grupa 4	16	13.47	45%
<b>Ukupno</b>	<b>126</b>	<b>30.93</b>	<b>54%</b>

Postignuti uspeh studenata, prikazan je u tabeli 6.9., a njihovo grupisanje dato je u skladu sa rangiranjem u radovima [GC]11a], kao i u prethodnom primeru na kontrolnom testu br. 1. I ovi rezultati pokazuju da je postignut značajan uspeh, jer procenat studenata koji su postigli pozitivnu ocenu (više od 50% osvojenih poena) iznosi 56%.

Tabela 6.9. – Uspeh studenata na kontrolnom testu br. 2

<b>Uspeh na testu / grupa</b>	<b>Grupa 1</b>	<b>Grupa 2</b>	<b>Grupa 3</b>	<b>Grupa 4</b>	<b>Ukupno</b>	<b>Procenat osvojenih bodova</b>
85%-100%	16	14	0	3	33	26%
65%-85%	8	9	3	2	22	17%
50%-65%	4	1	8	3	16	13%
Ispod 50%	30	6	11	8	55	44%
<b>Ukupno studenata</b>	<b>58</b>	<b>30</b>	<b>22</b>	<b>16</b>	<b>126</b>	

I pored znatno manje osvojenih bodova (gledano u procentima) od prethodnih testiranih grupa, rezultati ovog testa takođe pokazuju efikasnost VNLab-a. Treba se uzeti u obzir da je test najavljen samo 2 dana pre njegovog održavanja. Studenti nisu bili upoznati sa oblastima koje će biti na testu, tako da su za njegovo rešavanje koristili isključivo znanje koje su sticali u toku laboratorijskih vežbi. Pored toga, test je činilo težim i to što zadaci nisu bili slični sa zadacima iz laboratorijskih vežbi, kao u prethodnom testu. To potvrđuje da VNLab pruža mogućnost studentima da usvoje praktična i teorijska znanja koje mogu koristiti u različitim situacijama.

Ako bi se vršila analiza delova zadataka koji su najlošije odrađeni pokazalo bi se da su to zadaci 1b i 2b, što opet daje jasne pokazatelje za dalje usavršavanje laboratorije i laboratorijskih vežbi, kao i za davanje većeg akcenta na problematične teme u nastavku semestra i buduće generacije.

#### **6.4. Vrednovanje laboratorije na osnovu formativne procene**

Jedan on načina da se unapredi nastavni proces u VNLab okruženju je i upotreba čestih testova znanja studenata. U tom procesu može se izvršiti testiranje po modelu na formativne procene. Na početku, potrebno je objasniti način sprovođenja formativne procene (eng. *formative assasment*), kao i njenu ulogu.

Formativna procena je pristup u proceni znanja učenika, koji se razlikuje od sumativne procene. Sumativna procena (eng. *summative assasment*) se obično vrši periodično, na kraju određenog ciklusa, npr. na kraju semestra. Procena se vrši sa ciljem da bi se utvrdilo šta studenti znaju, a šta

ne. Sumativne procene se obično koriste za formiranje ocene, ali se u tu grupu ne svrstavaju samo završni ispiti.

I pored toga što informacije koje se dobiju sumativnom procenom mogu biti od izuzetno velike važnosti, one se mogu koristiti za vrednovanje samo određenih aspekata procesa učenja. Bez obzira na učestalost obavljanja ovih procena, tj. da li se procene vrše na nekoliko nedelja, meseci ili jednom godišnje, sumativne procene mogu da pomognu pri evaluaciji efikasnosti programa, poboljšanju nastave, usavršavanju programa ili poziciji studenata u odnosu na specifični program. Nasuprot tome, ova vrsta procene nalazi se na kraju procesa učenja te je stoga neprikladna da omogući informacije potrebne za poboljšanje nastave i slične intervencije u toku procesa učenja. To je funkcija formativnih procena.

Formativna procena je deo nastavnog procesa. Kada se inkorporira u nastavnu praksu, omogućava informacije potrebne za poboljšanje i podešavanje učenja i podučavanja u toku njihovog izvršavanja. Na taj način, formativna procena daje informacije i nastavniku i studentima o nivou razumevanja koji imaju studenti, tj. o nivou usvojenog znanja. Tako, korekcija nastavnog procesa i prezentovanje gradiva mogu da se izmene u cilju otklanjanja uočenih nedostataka [GE07].

Povratna informacija, koju formativna procena nudi kao rezultat, predstavlja osnovnu interaktivnu komponentu koja pruža podršku učenju [Cla95, Per98, Sad89, Sad98]. Procena se može smatrati formativnom samo u slučaju kada je njen rezultat akcija koju sprovodi predavač ili učenik u cilju unapređenja učenja [BW98]. Preciznije, informacija koju daje formativna procena koristi se od strane nastavnika ili studenata za modifikaciju rada u cilju postizanja veće efikasnosti [Bla95]. Formativna procena se definiše kao proces analize, vrednovanja i prosuđivanja rada studenata ili njihovih postupaka u toku rada da bi se poboljšala studentska kompetencija [Gip94]. Na ovu vrste procene može se gledati kao na pojavu koja se događa u okviru interakcije nastavnik-student, a koja predstavlja presek nastave i učenja [Gip94].

Formativna procena je vršena u toku čitavog perioda upotrebe VNLab okruženja u periodu od tri godine. U tabeli 6.10. prikazan je broj studenata koji su učestvovali u formativnoj proceni i u kojoj školskoj godini. Broj studenata se razlikovao od vežbe do vežbe, u zavisnosti od njihove prisutnosti na času. U nekim slučajevima, nisu postojale još formirane vežbe, kao što je to slučaj sa vežbama za fajervol sistem (FW) i sistem domenskih imena (DNS) u školskoj godini 2007/2008. U nekim slučajevima, nije sprovedeno testiranje studenata, jer u toku časa nije ostalo dovoljno vremena za neometan rad studenata, pa je po proceni predavača izostavljeno testiranje da bi se izbeglo dobijanje neadekvatnih rezultata.

Tabela 6.10. – Broj studenata koji su učestvovali u formativnoj proceni

	<b>IPv4-1</b>	<b>IPv4-2</b>	<b>IPv6</b>	<b>RIP</b>	<b>OSPF-1</b>	<b>OSPF-2</b>	<b>BGP-4</b>	<b>FW</b>	<b>DNS</b>
2007/2008	46	47	49	45	43	44	34	N/A	N/A
2008/2009	52	52	59	48	56	68	58	25	51
2009/2010	-	58	64	54	51	27	46	72	-

Formativna procena u VNLab okruženju imala je veliki značaj. Razlozi za to su sledeći:

- veliki broj koncepata koji se nauče u vežbama mogu se kasnije koristiti u vežbama koje slede,

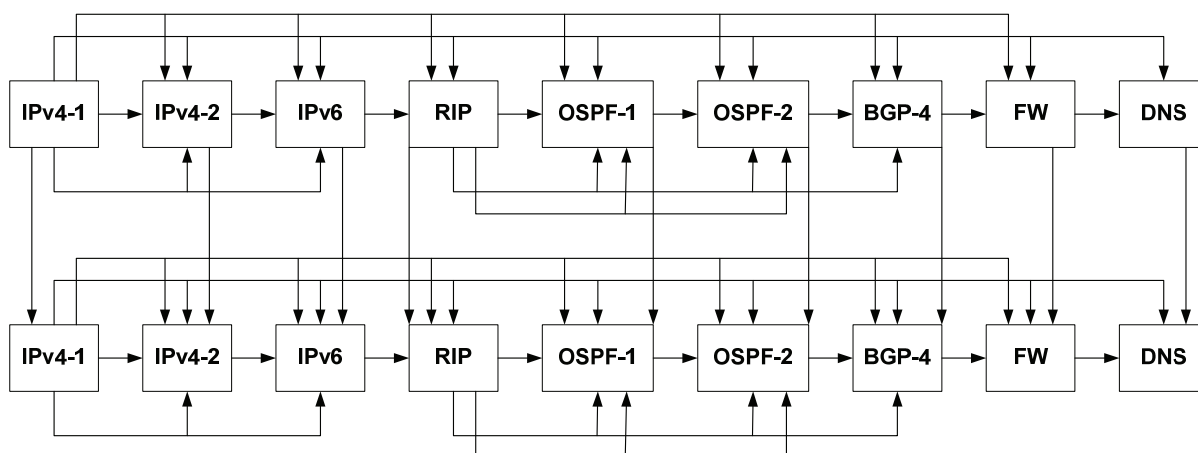
- okruženje u kojem studenti rade, takođe je nešto sa čime se studenti susreću u narednim vežbama.

To znači da se formativna procena može koristiti za uočavanje problema koje studenti imaju u shvatanju pojedinih delova vežbe, pojedinih koncepata i da se ti delovi mogu korigovati da bi studenti u narednim vežbama imali što je moguće manje problema. Rezultati formativne procene u jednoj vežbi, koristili su se da se identifikuju delovi na koje predavač treba u većoj meri da obrati pažnju prilikom njihovog objašnjavanja. Uočeni nedostaci korigovali su se tako da budu razumljiviji za studente sledeće generacije, a delovi koji su bili izostavljeni ili nedovoljno detaljni dodati su i prošireni po potrebi.

U praksi to je značilo sledeće. Prva vežba, vežba koja objašnjava osnovne koncepte IPv4 adresiranja i statičkog rutiranja bila je ujedno prva vežba koja koristi VNLab okruženje. Delovi vežbe koji su nedovoljno razumljivi, a koji se odnose na to okruženje, odražavaju se nepovoljno na rad studenata. Formativnom procenom identifikovani su ti delovi, a studentima je u daljem radu povećan stepen razumevanja i olakšan rad.

Dalje, u prvoj vežbi studenti se prvi put u praksi sreću sa važnim konceptima kao što su IP adresa, njeno dodeljivanje interfejsu, unos statičkih ruta za odgovarajuće mreže, podešavanje mrežne maske i gejtveja. Formativnom procenom, utvrđeno je koji od navedenih koncepata jesu, a koji nisu razumljivi i to je pomoglo da se dalji tok nastave i vežbe prilagode studentima.

Svi ovi koncepti imaju značajan uticaj i na dalji rad studenata. Ako studenti ne nauče i ne shvate dobro format IP adrese mreže, oni će imati problema i u narednim vežbama (IPv4-2, IPv6, RIP, OSPF-1, OSPF-2, BGP-4 i FW). Slično je i sa IP adresama interfejsa i mrežnom maskom, koje se ponovo koriste u svim pomenutim vežbama sa dodatkom DNS vežbe. Ostali koncepti takođe imaju uticaj na više od jedne vežbe. Npr. osnovni koncepti statičkog rutiranja, kao što su odabir za koje mreže je potrebno uneti rute i koji se gejtvej koristi potrebni su za vežbe IPv4-2 i IPv6. Podešavanje pasivnog interfejsa u vežbi RIP, se kasnije koristi u vežbama OSPF-1 i OSPF-2. Slično, asocijacija mreža sa odgovarajućim protokolom dinamičkog rutiranja se koristi u vežbama RIP, OSPF-1, OSPF-2 i BGP-4. Uticaj pojedinih koncepata na naredne vežbe, prikazan je na slici 6.6. Uvidom u ovu zavisnost moguće je u potpunosti shvatiti značaj upotrebe formativne procene u jednom okruženju kao što je VNLab.

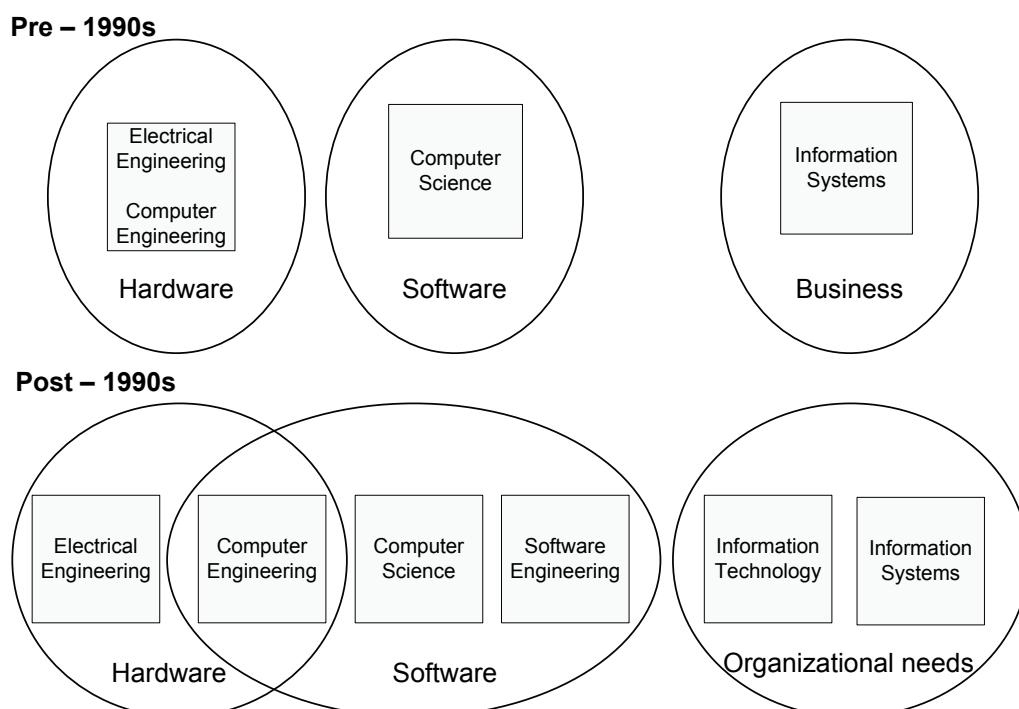


Slika 6.6. Prikaz uticaja koncepata iz jedne vežbe na uticaj koncepata u drugoj vežbi



## 6.5. Procena šire upotrebljivosti VNLab laboratorije

Upotrebljivost laboratorije je procenjena u skladu sa predloženim nastavnim planovima od strane svetske organizacije ACM [ACM11]. Na slici 6.7. prikazana je klasifikacija nastavnih programa iz oblasti računarstva, kao i podoblasti na koje su ti profili usmereni [Com05]. Prikaz je urađen za period do 1990. god. i posle. Procena upotrebljivosti je izvršena u skladu sa svakim od programa posebno. To je učinjeno da bi se stekao uvid u primenljivost ovakvog modela laboratorije u različitim predmetima i njihovim sadržajima na globalnom nivou, a koji nisu u direktnoj vezi sa određenom ustanovom.



Slika 6.7. Klasifikacija nastavnih programa u oblasti Računarskih nauka

### 6.5.1. Profil Informacione tehnologije

U okviru profila informacione tehnologije (eng. *Information technology*) preporučenog od organizacije ACM (Association for Computing Machinery) i IEEE (Institute of Electrical and Electronics Engineers), laboratorija VNLab je primenljiva u predmetu Networking (NET) – koji pokriva 22 obavezna časa [Lun08]. Taj predmet je podeljen na sledeće oblasti:

- NET. Foundations of Networking
- NET. Routing and Switching
- NET. Physical Layer
- NET. Security
- NET. Network Management
- NET. Application Areas

Zbog toga što sve IT aplikacije uključuju umrežavanje računara i računarske opreme, uloga IT profesionalca je da selektuje, dizajnira, upotrebi, integriše i administrira mrežnu i komunikacionu infrastrukturu u organizaciji. Njegove oblasti interesovanja uključuju prenos

podataka, *inter/intranetworking* i sigurnost sistema. Takođe, potrebna znanja uključuju primenu sistema za umrežavanje na multimedijalne sisteme, skladištenje podataka i World Wide Web.

U okviru modula/nastavne jedinice *NET.Foundations of Networking*, VNLab se može koristiti za postizanje jednog od navedenih ciljeva učenja, a to je:

– konfiguracija klijentskog i serverskog operativnog sistema i povezivanje klijentske mašine na server preko LAN mreže.

Za modul *NET. Routing and Switching* istog kursa, VNLab se može koristiti za sledeće osnovne ciljeve učenja:

- opis potrebnog hardvera, svičeva, rutera i komponenata (algoritma rutiranja i protokola) koji se koriste za uspostavljanje komunikacije između više mreža

Kao i sledeće napredne ciljeve učenja:

- konstrukcija više mreža i njihovo povezivanje,
- analiza i objašnjenje rutiranja algoritama i protokola, tabela rutiranja i konfiguracija rutera za određenu operaciju,
- poređenje rutiranja protokola za interno (eng. *interior routing protocols*) i eksterno rutiranje (eng. *exterior routing protocols*),

U okviru modula *NET.Security*, VNLab pokriva cilj koji obuhvata rad sa fajervol sistemima, i napredni cilj koji obuhvata instalaciju i konfiguraciju takvih sistema. *NET.Network Management* modul podržan je u VNLab okruženju za napredne ciljeve kao što su:

- dizajn i implementacija LAN mreža,
- dizajn i implementacija WAN mreža,
- konfiguracija mrežnih čvorova (računari, ruteri itd.),
- rešavanje problema u mrežama.

### **6.5.2. Profil Računarstvo**

U okviru preporučenog programa za profil Računarstvo (*Computer Science*) koji preporučuju zajednički ACM i IEEE, predviđen je kurs pod nazivom *Net Centric Computing* [Com08]. Najnovija dostignuća u oblasti računarskih mreža i telekomunikacija, posebno onih sistema koji su zasnovani na TCP/IP protokolu, uvećala su važnost tehnologija umrežavanja za računarstvo. Kurs pod nazivom *Net Centric Computing* obuhvata široko područje koje uključuje: koncepte i protokole računarskih mreža, multimedijalne sisteme, veb standarde i tehnologije, sigurnost mreža, bežične i mobilne mreže i distribuirane sisteme.

Odgovarajući nivo znanja u ovoj oblasti uključuje podjednako i teoriju i praksu. Iskustva u učenju koja uključuju eksperimentisanje i analizu preporučuju se jer pospešuju razumevanje konceptata i njihovih primena u realnim uslovima. Laboratorijski eksperimenti uključuju sakupljanje podataka i sintezu, empirijsko modelovanje, analizu protokola, monitoring mrežnih

paketa, konstrukciju softvera i dr. Svi ovi važni koncepti mogu se razumeti najbolje kroz laboratorijske eksperimente.

Kurs *NC.Net Centric Computing* (18 obaveznih časova) sastoji se od sledećih modula:

- NC/Introduction [obavezni]
- NC/NetworkCommunication [obavezni]
- NC/NetworkSecurity [obavezni]
- NC/WebOrganization [izborni]
- NC/NetworkedApplications [izborni]
- NC/NetworkManagement [izborni]
- NC/Compression [izborni]
- NC/MultimediaTechnologies [izborni]
- NC/MobileComputing [izborni]

Pojedinačno gledano, modul *NC/Introduction* ima za cilj učenja demonstraciju osnovnih mrežnih aplikacija kao što su: e-mail, telnet, FTP, veb brauseri itd. To je deo koji je podržan u VNLab okruženju.

Modul *NC/NetworkCommunication* pokriva teme kao što sedmoslojni ISO referentni model i njegovo preslikavanje na TCP/IP model, zatim povezivanje mreža i rutiranje sa algoritmima rutiranja. Ove teme se mogu podržati upotrebom VNLab-a. Isti modul ima sledeće napredne ciljeve učenja koji mogu da se rade u VNLab-u:

- opis prva četiri sloja ISO modela,
- opis usmeravanja (rutiranja) paketa na Internetu,
- instalacija jednostavnih mreža sa dva klijenta i jednim serverom,

Redom, *NC/NetworkSecurity* modul obuhvata teme kao što su fajervol sistemi i NAT (Network Address Translation), *NC/NetworkManagement* modul obuhvata domenska imena i DNS. Dodatno, ovaj modul preporučuje kao napredne ciljeva učenja implementaciju fajervol sistema. Sve navedeno može se ostvariti upotrebom VNLab okruženja.

### **6.5.3. Profil Računarsko inženjerstvo**

Za obrazovni profil Računarskog inženjerstva (eng. *Computer engineering*), koji takođe preporučuju ACM i IEEE, predviđen je kurs *Computer Networks* (CE-NWK) sa sledećim modulima [Com04]:

- CE-NWK0 History and overview [obavezni]
- CE-NWK1 Communications network architecture [obavezni]
- CE-NWK2 Communications network protocols [obavezni]
- CE-NWK3 Local and wide area networks [obavezni]
- CE-NWK4 Client-server computing [obavezni]
- CE-NWK5 Data security and integrity [obavezni]
- CE-NWK8 Wireless and mobile computing [obavezni]
- CE-NWK6 Performance evaluation [izborni]

- CE-NWK7 Data communications [izborni]
- CE-NWK9 Network management [izborni]
- CE-NWK10 Compression and decompression [izborni]

U odnosu na ovaj kurs, VNLab pokriva teme i ciljeve koji su opisani u nastavku za svaki odgovarajući modul. U okviru modula *CE-NWK0 History and overview* teme podržane od strane VNLab okruženja su:

- opis funkcija mrežnih uređaja kao što su ripiteri, mostovi, ruteri i mrežni prolazi,
- opis popularnih protokola.

Modul *CE-NWK1 Communications network architecture* ima teme kao što su:

- uređaji za povezivanje (repetitori, mostovi, svičevi, ruteri, mrežni prolazi),
- konekcioni i bezkonekcioni servisi.

Modul *CE-NWK2 Communications network protocols*, kao što mu i ime govori, obuhvata temu TCP/IP protokola, a cilj učenja preporučuje poređenje OSI modela sa TCP/IP modelom. U modulu *CE-NWK5 Data security and integrity* izučavaju se fajervol sistemi.

Na kraju, modul *CE-NWK9 Network management*, obuhvata temu domenskih imena i *name servisa*, a napredni ciljevi učenja uključuju i implementaciju fajervol sistema.

#### **6.5.4. Profil Informacioni sistemi**

Četvrti obrazovni profil – Informacioni sistemi (eng. *Information system*), preporučen je od strane AIS (Asociation of Information Systems) i ACM. U svojoj strukturi ima kurs IS 2010.4 IT Infrastructure [IS10]. Ovaj kurs pruža uvod u IT infrastrukturu za studente koji su upisali smer Informacioni sistemi. Pokriva teme vezane za arhitekturu računara, računarskih sistema i komunikacionih mreža sa fokusom na servise i infrastrukturna rešenja. U toku ovog kursa studenti dobijaju znanja i iskustva potrebna za efikasnu komunikaciju sa profesionalcima koji su specijalizovani za hardver i sistemski softver kao i za razumevanje ograničenja i mogućnosti IT infrastrukture. Takođe, ovaj kurs priprema studente za interakciju sa spoljnim snabdevačima IT komponenti i rešenja. Kurs je fokusiran na Internet bazirana rešenja, sigurnost računara i računarskih mreža i poslovne aplikacije.

Upotrebljivost VNLab u nastavi u ovom kursu je vidljiva kroz sledeće ciljeve učenja:

- razumevanje razlika i sličnosti između osnovnih elemenata IT infrastrukture kao što su klijenti, server, mrežni uređaji, žični i bežični linkovi, sistemski softver i specijalizovani sigurnosni uređaji,
- razumevanje kroz praktične primere kako se protokoli koriste da omoguće komunikaciju između računarskih sistema koji su međusobno povezani,
- konfiguracija IT infrastrukturnog rešenja za male organizacije, uključujući i mreže zasnovane na standardnoj tehnologiji i komponentama, serverima, sigurnosnim sistemom i različitim komunikacionim klijentima,

- primena osnovnih koncepata na IP mreže da bi se rešili jednostavni problemi kod dizajniranja mreža, uključujući i podelu na podmreže,

Teme koje su obuhvaćene ovim kursem, a koje su u vezi sa umrežavanjem su:

- osnovne mrežne komponente,
- TCP/IP model,
- mrežni sloj: IP, IP adresiranje i rutiranje,
- transportni sloj,
- konfiguracija mrežnih uređaja,
- sigurnost i fajervol sistemi.

Preporuka je da se, kada god je to moguće, koristi praktičan rad i praktične vežbe da bi se naučili kompleksni koncepti računarskih mreža i da je učenje ovih tema suviše apstraktno bez praktičnog rada.

#### **6.5.5. Zaključak**

Nakon analize preporučenih nastavnih programa za obrazovne profile iz oblasti Računarstva, može se zaključiti da je model virtuelne mrežne laboratorije VNLab primenljiv u delu navedenih kurseva sva četiri profila. Primena VNLab laboratorije u većem obimu u okviru ovih kurseva takođe je moguća, te se to nameće kao jedan od mogućih ciljeva u razvoju ove laboratorije u budućnosti. Tehnologija na kojoj je laboratorija zasnovana, kao i model virtuelne mrežne laboratorije, omogućuju takav razvoj.

## 7. Zaključna razmatranja

### 7.1. Rezultati istraživanja

Nakon istraživanja koje je opisano u disertaciji, rezultati se mogu i ukratko sumirati. Osnovni rezultat koji je nastao tokom istraživanja jeste model virtuelne mrežne laboratorije, prilagođen za pristup na daljinu i posebno prikazan u poglavlju 5. U tom prikazu navedene su sve bitne komponente modela, softverske i hardverske, kao i struktura laboratorijskih vežbi i mrežna scenarija koja obezbeđuju osnovnu funkcionalnost laboratorije. Laboratorija je upotrebljena u nastavi sa ciljem realizacije vežbi iz predmeta Računarske mreže, kao i za potrebe testiranja njene funkcionalnosti. Primena u nastavi je realizovana u nekoliko etapa.

Prve dve godine, u inicijalnoj fazi razvoja modela, laboratorija je upotrebljena u nastavi u okviru tri laboratorijske vežbe, a kasnije u okviru sedam laboratorijskih vežbi u okruženju sa lokalnim pristupom (školske godine 2005/2006 i 2006/2007). Iste školske godine 2006/2007, prvobitno osmišljeni deo vežbi (ukupno 3) je prebačen u novo okruženje, tj. okruženje sa mogućnošću pristupa na daljinu. U naredne četiri školske godine, tj. 2007/2008, 2008/2009, 2009/2010 i 2010/2011 godine, laboratorija je korišćena u nastavi u okviru 9 laboratorijskih vežbi. Vežbe iz predmeta Računarske mreže su realizovane tokom letnjeg semestra svake od navedenih godina. Upotrebljivost laboratorije u nastavi analizirana je na osnovu metoda i alata predstavljenim u poglavlju 2, kao i na osnovu iskustvenog obrasca koji je formiran tokom same upotrebe laboratorije.

Uporedo sa razvojem modela laboratorije i njenom upotrebom u nastavi, razvijena je metodologija za vrednovanje efikasnosti laboratorije, njene upotrebljivosti u nastavi, kao i za analizu i vrednovanje laboratorijskih vežbi i uspeha studenata. Akcenat je stavljen na vrednovanje efikasnosti laboratorijskih vežbi, uočavanje njihovih nedostataka i njihovo unapređivanje. Metod koji je odabran za ovakav pristup, a koji predstavlja novinu u toj oblasti i svojevrsan naučni doprinos je metod zasnovan na teoriji grubih skupova. Upotreba tog metoda, njegov detaljan opis, kao i rezultati analize dati su u poglavlju 6.1. Metod je upotrebljen na podacima koji su dobijeni na osnovu testiranja 55 studenata nakon tri laboratorijske vežbe. Analizom je dobijeno 70 pravila koja su generisana upotrebom programa Rossetta i izvedeni su sledeći zaključci:

- analiza pravila je dala preciznu indikaciju u vezi sa najvećom razlikom u znanju između studenata koji su postigli najviše ocene i ostalih, što je doprinelo poboljšanju vežbi. Ovo je realizovano kroz koncentrisanje na one delove vežbi sa kojima studenti imaju najviše problema, što je dovelo do poboljšanja vežbi i podizanja efikasnosti virtuelne laboratorije,
- ovaj metod je omogućio da se izoluju oni delovi vežbi koji su lako razumljivi svim studentima, kako bi bili isključeni iz ocenjivanja ili kako bi se umanjio njihov uticaj na konačnu ocenu. Prema toj analizi, broj poena koje student dobija ako uspešno reši taj deo zadatka je smanjen,
- omogućeno je uočavanje uobičajenih grešaka koje studenti prave u slučajevima kada dobijaju niže ocene, kao i utvrđivanje dela vežbi koji treba unaprediti kako bi se povećao opšti nivo znanja studenata.

Upotreba kontrolnih testova je takođe poslužila u cilju vrednovanja laboratorije. Kontrolni testovi su održani dva puta, u toku školske godine 2009/2010 i 2010/2011. Prvi test je najavljen u dužem vremenskom periodu pre njegovog održavanja i njime je obuhvaćeno 48 studenata sa dva smjera: inženjeri poslovne informatike i profesori informatike III godina. Drugi test je bio najavljen neposredno pred održavanje (dva dana ranije) i sproveden je nad 126 studenata sa II i III godine obrazovnih profila Informacione tehnologije (II godina) i smerova: inženjer informatike, profesor informatike i inženjer poslovne informatike (III godina). Posmatrani indikator ovih kontrolnih testova je procenat ostvarenih poena u odnosu na mogući broj. U prvom slučaju taj procenat iznosi 77%, drugom slučaju 54%. Drugi posmatrani indikator bio je broj studenata koji su dobili pozitivnu ocenu. U prvom slučaju je to 96% studenata, a u drugom 56%. U oba slučaja ostvaren je zadovoljavajuće visok rezultat što ukazuje na zaključak:

- VNLab predstavlja efikasno okruženje za rad i upotrebu u nastavi Računarskih mreža i za učenje mrežnih koncepata.

Vrednovanje laboratorije, kao i laboratorijskih vežbi realizovano je na osnovu formativne procene u periodu od tri školske godine (2007/2008, 2008/2009 i 2009/2010). Procena je obuhvatala rešavanje zadataka na kraju svakog časa, tj. svake završene laboratorijske vežbe. Struktura vežbi i zadataka prikazana je u dodacima (Dodatak A, B, C i D). Studentima su bili zadati problemi slični onima koje su imali u toku vežbe. Odgovori koje su studenti davali u pisanoj formi poslužili su da se kontinuirano prati stepen usvojenih znanja, zatim da se identifikuju elementi koji su najteži za razumevanje, kao i da se utvrde eventualne greške i nepravilnosti u materijalu za laboratorijske vežbe. Broj studenata koji su obuhvaćeni ovim testiranjem je 189.

Pored pronalazanja grešaka u materijalu i mrežnim scenarijima, analiza podataka je ukazala i načine na osnovu kojih se greške mogu korigovati. Korekcije su se zatim odrazile na prilagođavanje vežbi potrebama studenata u toku nastave, kao i na eliminisanje nedostataka i eventualnu promenu koncepta pojedinih vežbi za naredne školske godine tj. za buduću upotrebu.

Konačno, metod komparacije je poslužio za analizu upotrebljivosti VNLab okruženja u nastavi na računarskim obrazovnim profilima. Poređenje je ostvareno u odnosu na preporučene profile iz oblasti računarstva od strane organizacija ACM (Association for Computing Machinery), IEEE (Institute of Electrical and Electronics Engineers) i AIS (Association for Information Systems).

Komparacija je pokazala da je VNLab upotrebljiv u predmetima koji su vezani za temu računarskih mreža u okviru sledećih obrazovnih profila:

- Informacione tehnologije (*Networking – NET* kurs)
- Računarsko inženjerstvo (*Computer Networks – CE-NWK* kurs)
- Informacioni sistemi (*IS 2010.4 IT Infrastructure* kurs)
- Računarstvo (*Net Centric Computing – NC* kurs)

Na osnovu prikazane analize, VNLab se može karakterisati kao sredina dovoljno stabilna i pouzdana za korišćenje u nastavi. Ona je takođe prilagođena upotrebi u nastavi na odgovarajući način, a njeni manji nedostaci i specifičnosti korišćenja, koji se ne mogu ispraviti ili promeniti, ne utiču negativno na rad laboratorije.

## 7.2. Dalji razvoj

Nakon prikaza rezultata evidentno je da se mogu odrediti i budući pravci istraživanja i to više njih. VNLab se može proširiti sa još većim brojem laboratorijskih vežbi i novim konceptima kao što su VOIP, MPLS, i napredni aspekti dinamičkog rutiranja. One mogu biti iz istog predmeta kao postojeće vežbe, tj. prilagođene predmetu Računarske mreže. Ovakav pristup bi neznatno doprineo unapređenju okruženja.

Pravac razvoja kojim bi se omogućilo prilagođavanje VNLab-a drugim predmetima ima daleko veliki značaj. VNLab bi se mogao koristiti u nastavi sledećih predmeta:

- Operativni sistemi i njihova administracija,
- Zaštita podataka i računarskih mreža,
- Napredne računarske komunikacije,
- Internet programiranje,
- Programiranje serverskog softvera i aplikacija,
- Distribuirani sistemi i druge predmete.

Sledeći pravac razvoja odnosi se na prelazak na novu platformu. VNLab je realizovan na relativno starom softveru i njegova modernizacija može povećati efikasnost okruženja, što ne mora da bude i jedini cilj prelaska na novu platformu. Primena novog softvera bi mogla da omogući i nove oblike njegove upotrebe, i to one koji do sada nisu bili mogući. Nova softverska platforma se može uvesti na svim nivoima modela. Prvi od njih se odnosi na prelazak na novi operativni sistem servera na kome se nalazi laboratorije (VNLab server) i novi softver za virtuelizaciju. Drugi vid upotrebe novog softvera je prelazak na novije verzije i različite tipove operativnih sistema koji se koriste za virtuelne mašine. Na kraju, tu bi se mogla uvrstiti i upotreba novijih verzija postojećeg softvera ili upotreba drugih verzija softvera koja se koriste za razne namene (veb, mail, multiprotokol ruting).

Treća, ali nikako manje važna orijentacija razvoja ovog okruženja se odnosi na mehanizam koji omogućava samostalan daljinski pristup VNLab-u, kao i samostalan rad studenata i pristup raspoloživim resursima ovakvog okruženja. Dosadašnja laboratorija nema ovu mogućnost, budući da je prisustvo predavača obavezno zbog koordinacije rada. Pružanjem takvih usluga ovo okruženje bi se moglo koristiti i za rad studenata koji mogu pristupati laboratoriji od kuće i u vannastavnim terminima. Za to je potrebno razviti sistem za određivanje kojim računarima će pristupati određeni studenti, kao i za sprečavanje višestruke konekcije na istim virtuelnim mašinama.

Na kraju, jedno od mogućih pravaca razvoja je i integracija sistema za podršku formativnoj proceni u VNLab okruženje. Takav sistem bi trebao da omogući testiranje znanja studenata na kraju svake vežbe elektronskim putem i automatizaciju pregleda studentskih odgovora.

## 7.3. Zaključak

Na osnovu svega iznetog može se zaključiti da je potvrđena glavna hipoteza istraživanja koja glasi:



*„Moguće je kreirati efikasnu virtuelnu mrežnu laboratoriju baziranu na tehnologiji za virtuelizaciju, kao okruženje za emulaciju računarskih mreža sa mogućnošću lokalnog i daljinskog pristupa, koja je pogodna za upotrebu u nastavi na visokoškolskim ustanovama.“*

Uporedo sa njom potvrđene su i pothipoteze:

- *moguće je kreirati virtuelnu mrežnu laboratoriju baziranu na tehnologiji za virtuelizaciju*

Ova pothipoteza potvrđena je prikazom i razvojem modela virtuelne mrežne laboratorije sa lokalnim pristupom i njegovom realizacijom, što je opisano u poglavlju 4., kao i prikazom, modeliranjem i realizacijom modela virtuelne mrežne laboratorije sa pristupom na daljinu – VNLab, što je dato u poglavlju 5.

- *kreirana mrežna laboratorija predstavlja emulaciju realnih računarskih mreža*

Pothipoteza je potvrđena prikazom i analizom scenarija (poglavlje 5.4. do poglavlja 5.9. i Dodaci A, B, C i D) koji ukazuju da laboratorija predstavlja okruženje koje odgovara realnim sistemima, i da situacije i scenarija koja su data studentima za učenje predstavljaju okruženje koje ih može pripremiti za rad u realnim sistemima.

- *nastava iz pojedinih predmeta na visokoškolskim ustanovama se može organizovati u okruženju virtuelne mrežne laboratorije*

Pothipoteza je potvrđena na osnovu poređenja sa preporučenim programima obrazovnih profila računarske struke koja su prikazana u Poglavlju 6.4.

- *virtuelna mrežna laboratorija se može realizovati i kao deo sistema za učenje na daljinu*

Ova hipoteza je delimično potvrđena ovim istraživanjem jer je omogućen pristup na daljinu laboratoriji, ali ona u toku istraživanja nije integrisana u neki postojeći sistem za učenje na daljinu.

Glavni naučni doprinos ovog istraživanja je modeliranje i realizacija virtuelne mrežne laboratorije koja omogućuje pristup na daljinu i koja je bazirana na tehnologiji za virtuelizaciju. Realizovana laboratorija razvijena je po ugledu na model razvijen u toku ovog istraživanja, a koji se razlikuje od svih ostalih modela.

Drugi naučni doprinos predstavljaju mrežni scenariji koji su prilagođeni specifičnom modelu laboratorije, koji su verifikovani u nastavi, i koji su se pokazali dovoljno efikasnim za upotrebu u učenju mrežnih koncepata.

Treći doprinos predstavlja jedinstveni metod, baziran na teoriji grubih skupova, koja se koristi za vrednovanje usvojenih znanja studenata u toku korišćenja virtuelne mrežne laboratorije, kao i za analizu laboratorijskih vežbi i laboratorijskih materijala.

## LITERATURA

- [AA06] Adams K, Agesen O., *A comparison of software and hardware techniques for x86 virtualization*, Proceedings 12th International Conference on Architectural support for programming languages and operating system, pp 2–13, San Jose, California, USA, 2006.
- [Abo03] Abolela, E., *Network Simulation Experiments Manual*, 3ed., Elsevier Science Ltd , USA, 2003.
- [ACM07] Paul B. Albee, Laura A. Campbell, Martin A. Murray, Christopher M. Tongen, Jennifer L. Wolfe, *A student-managed networking laboratory*, Proceedings of the 8th ACM SIGITE conference on Information technology education (SIGITE '07), ACM, New York, NY, USA, 2007.
- [Acm11] ACM (Asociation for Computer Machinery) official web site – curricula recommendations, <http://www.acm.org/education/curricula-recommendations>, pristupljeno 2011.
- [AJD09] Benjamin R. Anderson, Amy K. Joines, Thomas E. Daniels, *Xen worlds: leveraging virtualization in distance education*, Proceedings of Annual Joint Conference Integrating Technology into Computer Science Education (ITiCSE), pp 293-297, Paris, France, July 2009.
- [Al-04] Salah M.S. Al-Buraiky, *Building a Virtual IPv6 Lab Using User-Mode Linux*, ;login:, The USENIX Magazine, Vol. 29, No. 1, February, 2004.
- [Ale09] Paulo Dias de Alecrim, *Livro Simulação Computacional para Redes de Computadores (Portuguese)*, Ciência Moderna, 2009.
- [Anu07] Anuzelli G, *Dynagen: The Network Configuration Generator for Dynamips*, <http://dynagen.org>, 2007.
- [Arn05] Richard Arndt, *VNUML – Virtual Network User Mode Linux*, <http://www.uni-koblenz.de/~steigner/seminar-routingsim/arndt.pdf>, 28. January 2005.
- [BBC04] Andy Bavier, Mic Bowman, Brent Chun, David Culler, Scott Karlin, Steve Muir, Larry Peterson, Timothy Roscoe, Tammo Spalink, Mike Wawrzoniak, *Operating System Support for Planetary-Scale Services*, Proceedings of the First Symposium on Network Systems Design and Implementation (NSDI'04), pp 253-266, USENIX Association Berkeley, USA, March, 2004.
- [BBS06] Bullers, Jr., W. I. , Burd S., Seazzu, A. F., *Virtual machines - an idea whose time has returned: application to network, security, and database courses*, Proceedings of the 37th SIGCSE technical symposium on Computer science education (SIGSCE'06), pp 102-106, March 03-05 Houston, Texas, USA, 2006.
- [BDF03] Barham, P., B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, A. Warfield., *Xen and the Art of Virtualization*, Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP 03), pp 164–177, ACM, New York, USA, October, 2003.
- [Beg06] Begnumm KM., *Managing large networks of virtual machines*, Proceedings of 20th large installation system administration conference (LISA'06), USENIX, El Cerrito, pp 205–214, USA, 2006.

- [Bir11] BIRD (Web site), <http://bird.network.cz/>, 2011.
- [Bla95] Black P., *Can teachers use assessment to improve learning?*, British Journal of Curriculum & Assessment, Vol.5, Issue 2, pp 7–11, 1995.
- [Bon01] James Boney, *Cisco IOS in a Nutshell*, O'Reilly Media, December 2001.
- [Bor07] Charles Border, *The development and deployment of a multi-user, remote access virtualization system for networking, security, and system administration classes*, Proceedings of the 38th SIGCSE technical symposium on Computer science education (SIGCSE '07), pp 576-580, ACM New York, NY, USA, 2007.
- [Bos11] Boson Holdings (Web site), <http://www.boson.com/>, pristupljeno 2011.
- [Brt08] V. Brtka: "Automatska sinteza baze pravila u inferentnim sistemima", doktorska disertacija, TFMPZR, 2008.
- [BW98] Black P., Wiliam, D., *Assessment and classroom learning*, Assessment in Education: Principles, Policy & Practice, Vol. 5, No. 1, pp 7–74, 1998.
- [CBE10] Tim Cerling, Jeff Buller, Chuck Enstall, Richard Ruiz, *Mastering Microsoft Virtualization*, Wiley Publishing, Inc., Indianapolis, Indiana, USA, 2010.
- [CFM08] R. Coltun, D. Ferguson, J. Moy, RFC 5340, OSPF for IPv6, July 2008.
- [CG09] F. Candelas Herías, P. Gil Vázquez, *Practical Experiments with KivaNS: A virtual Laboratory for Simulating IP Routing in Computer Networks Subjects*, Proceedings of V International Conference on Multimedia and ICT in Education (m-ICTE 2009), Vol.3 (virtual participants), pp. 1414-1418. Lisboa, 2009.
- [Cis06] Cisco IOS Configuration Fundamentals Command Reference Release 12.2, Cisco Systems, Inc., [http://www.cisco.com/en/US/docs/ios/12\\_2/configfun/command/reference /frf\\_bk.pdf](http://www.cisco.com/en/US/docs/ios/12_2/configfun/command/reference/frf_bk.pdf), 2006.
- [Cla95] Clarke D., *Constructive assessment: Mathematics and the student*, AAMT Proceedings, Adelaide, 1995.
- [CM01] B. Carpenter, K. Moore, Connection of IPv6 Domains via IPv4 Clouds, Internet Enginnering Task Force, February 2001.
- [CN01] Peter M. Chen, Brian D. Noble, *When virtual is better than real*, Proceedings of the 2001 Workshop on Hot Topics in Operating Systems (HotOS), pp 133–138, May, 2001.
- [Com00] Comer Douglas E., *Internetworking with TCP/IP*, Vol. 1: Principles, Protocols, and Architecture, 4<sup>th</sup> Edition, Prentice Hall Inc, New Jersey, USA, 2000.
- [Com04] Computer Engineering 2004, Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering - A Report in the Computing Curricula Series, The Joint Task Force on Computing Curricula, IEEE Computer Society and Association for Computing Machinery, [http://www.acm.org/education/education/curric\\_vols/CE-Final-Report.pdf](http://www.acm.org/education/education/curric_vols/CE-Final-Report.pdf), December 2004 .

- [Com05] Computing Curricula 2005 - The Overview Report, The Joint Task Force for Computing Curricula 2005, A cooperative project of The Association for Computing Machinery (ACM), The Association for Information Systems (AIS), The Computer Society (IEEE-CS), [http://www.acm.org/education/education/curric\\_vols/CC2005-March06Final.pdf](http://www.acm.org/education/education/curric_vols/CC2005-March06Final.pdf), September 2005.
- [Com08] Computer Science Curriculum 2008: An Interim Revision of CS 2001 Report, Interim Review Task Force, <http://www.acm.org//education/curricula/ComputerScience2008.pdf>, December, 2008.
- [Cos02] Bryan Costales, *Sendmail, 3rd Edition*, O'Reilly Media, December 2002.
- [Cre81] R. J. Creasy, *The origin of the VM/370 time-sharing system*, IBM Journal of Research & Development, Vol. 25, No. 5, pp. 483-90, September 1981.
- [ČS02] M. ČupiĆ, M. Suknović: "Teorija odlučivanja: Formalni pristup", Beograd, FON, 2002.
- [Dav05] Davoli R., *VDE: Virtual Distributed Ethernet*, Proceedings of First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMMunities (TRIDENTCOM'05), pp 213-220, February 23-25, Trento, Italy, 2005.
- [DH98] S. Deering, R. Hinden, RFC 2460, *Internet Protocol - Version 6 (IPv6) Specification*, December, 1998.
- [Dik00] J. Dike, *A user-mode port of the Linux kernel*, Proceedings of the 4th Annual Linux Showcase & Conference ALS'00, Vol. 4., Atlanta, Georgia, USA, October, 2000.
- [Dik01] J. Dike, *User-mode Linux*, Proceedings of the 5th Annual Linux Showcase and Conference ALS'01, Vol. 5, Oakland, USA, Novembar, 2001.
- [Dik02] J. Dike, *Making Linux Safe for Virtual Machines*, Proceedings of the 2002 Ottawa Linux Symposium (OLS2002), pp 107-116, Ottawa, Canada, July, 2002.
- [DBB09] Dobrilovic, D., Brtko, V., Berkovic, I. and Odadzic, B., *Evaluation of the virtual network laboratory exercises using a method based on the rough set theory*, Computer Applications in Engineering Education, published online June 24th, 2009.
- [DSD09] Dobrilovic Dalibor, Stojanov Zeljko, Duka Milan, Odadzic Borislav, *Application for network node configuration based on NNDL (Network Node Description Language)*, Proceeding of 7th international symposium on intelligent systems and informatics (SISY 2009), pp 335-339, Subotica, Serbia, Septembar 25-26. 2009.
- [D006a] Dobrilovic Dalibor, Odadzic Borislav, *Virtualization Technology as a Tool for Teaching Computer Networks*, Proceedings of world academy of science, engineering and technology, Vol. 13, pp 126-130, 2006.
- [D006b] Dobrilović Dalibor, Odadžić Borislav, *Simulacija računarskih mreža na open-source platformi*, Zbornik radova 11<sup>th</sup> INFOTECH 2006, 12-16. Jun, Srbija, Vrnjačka Banja, 2006.

- [D006c] Dobrilovic D., Odadzic B., *Simulacija OSPF protokola u virtuelnoj mrežnoj laboratoriji*, 5<sup>th</sup> Infotech-Jahorina 2006, Vol. 5, Ref. E-II-1, pp 334-338, mart 22-24, Bosnia & Herzegovina, Jahorina, 2006.
- [DS06] Dobrilovic Dalibor, Stojanov Zeljko, *Using virtualization software in operating systems course*, Proceedings of the 4th IEEE International Conference on Information Technology: Research and Education (ITRE 2006), pp 222-226, Tel Aviv, Israel, 16-19 October 2006.
- [DS008] Dobrilovic Dalibor, Stojanov Zeljko, Odadzic Borislav, *Implementing BGP-4 Protocol Scenario in VNLab Open Networking Environment*, Proceedings of the IEEE 6th International Symposium on Intelligent Systems and Informatics (SISY2008), pp 1-5, Subotica, Serbia, September 26-27, 2008.
- [DS011] Dobrilovic D., Stojanov Z., Odadzic B., *Design and Implementation of Online Virtual Network Laboratory*, Advancement in Online Education: Exploring the Best Practices, Vol. 1, Chapter 10, Publication date: 2011 4th quarter, 2011.
- [DW08] Wenliang Du, Ronghua Wang, *SEED: A Suite of Instructional Laboratories for Computer Security Education*, Journal on Educational Resources in Computing (JERIC), Vol. 8, Issue 1, ACM New York, NY, USA, March 2008.
- [Ein06] Einar project (Web site), URL:<http://www.isk.kth.se/proj/einar/>, retrieved 2006.
- [Eis96] Technology Briefing Report, System Modelling, RENAISSANCE Esprit Project 22010, [http://www.comp.lancs.ac.uk/projects/renaissance/RenaissanceWeb/project/Acrobat/System\\_Modelling.pdf](http://www.comp.lancs.ac.uk/projects/renaissance/RenaissanceWeb/project/Acrobat/System_Modelling.pdf), School of Computing and Communications, Lancaster University, September 1996.
- [FD09] Jerry FitzGerald, Alan Dennis, *Fundamentals of Business Data Communications*, 10th Edition, Wiley, April 2009.
- [FGM04] David Fernández, Fermín Galán, Tomás de Miguel, *Study and Emulation of IPv6 Internet Exchange (IX) based Addressing Models*, IEEE Communications Magazine, Vol. 42, No. 1, pp 105-112, January, 2004.
- [FMJ02] Lluís Fàbrega, Jordi Massaguer, Teodor Jové, David Mérida, *A virtual network laboratory for learning IP networking*, Proceedings of the 7th annual conference on Innovation and technology in computer science education (ITiCSE'02), pp 161-164, ACM, New York, USA, 2002.
- [FN10] Li Feng, Yu Nana, *Design and implementation of TCP/IP protocol learning tool*, Proceedings of the Entertainment for education, and 5th international conference on E-learning and games, Springer-Verlag Berlin, Heidelberg, Germany, 2010.
- [FPG05] David Fernández, F. Javier Ruiz Piñar, Fermín Galán, Vicente Burillo, Tomás de Miguel, *Uso de técnicas de virtualización para mejorar la docencia en laboratorios de redes de comunicaciones*, V Jornadas de Ingeniería Telemática (JITEL 2005), Vigo (Spain), September 2005.

- [GBS98] Greco S, Benedetto M, Slowinski R., *New Dewelopments in the Rough Set Approach to Multi – Attribute Decision Analysis*, Bulletin of International Rough Set Society, Vol. 2 (2/3), pp 57–87, 1998.
- [GCJ11a] P. Gil, F. Candelas, C. A. Jara, *Constructive Learning for Network Courses based on Compact Simulations and SCORM*, Proceedings of IEEE 2nd Engineering Education Conference (EDUCON 2011 - IEEE Education Society), pp 105-110, Amman, Jordan, 4-6 April, 2011.
- [GCJ11b] P. Gil, F. Candelas, C.A. Jara, *Computer Networks E-learning Based on Interactive Simulations and SCORM*, International Journal of Online Engineering (iJOE), Vol. 7, No. 2, pp 15-23, 2011.
- [GE07] Garrison, C., Ehringhaus, M., *Formative and summative assessments in the classroom*, NMSA Assessment Resources, <http://www.nmsa.org/Publications/WebExclusive/Assessment/tabid/1120/Default.aspx>, 2007.
- [GF04] Fermín Galán, David Fernández, *VNUML: Una herramienta de virtualización de redes basada en Software Libre* (in Spanish), Open Software World Conference, Málaga (Spain), February 2004.
- [GF07] Galán F., Fernández D., *Distributed Virtualization Scenarios Using VNUML*, Proceedings of the First System and Virtualization Management Workshop (SVM'07), Tolouse, France, October 2007.
- [GFF09] Galán, F., Fernández, D., Fuertes, W., Gómez, M., López de Vergara J. E., *Scenario-based virtual network infrastructure management in research and educational testbeds with VNUML Application cases and current challenges*, Annals of Telecommunication, Vol. 64, No. 5-6, pp 305-323, 2009.
- [GFR04] Fermín Galán, David Fernández, Javier Rúiz, Omar Walid, Tomás de Miguel, *A Virtualization Tool in Computer Network Laboratories*, Proceedings of 5th International Conference on Information Technology Based Higher Education and Training (I THET'04), Istanbul, Turkey, May, 2004.
- [Gip94] Gipps, C, *Beyond testing: Towards a theory of educational assessment*, London, The Falmer Press, 1994.
- [GLA08a] Alessio Gaspar, Sarah Langevin, William D. Armitage, Matt Rideout, *March of the (virtual) machines: past, present, and future milestones in the adoption of virtualization in computing education*, Journal of Computing Sciences in Colleges, Vol. 23, Issue 5, pp 123-132, May, USA, 2008.
- [GLA08b] Gaspar, A., Langevin, S., Armitage, W., & Rideout, M., *Enabling new pedagogies in operating systems and networking courses with state of the art open source kernel and virtualization technologies*, Journal of Computing Sciences in Colleges, Vol. 23, Issue 5, pp 189-198, 2008.
- [Gol73] R. Goldberg, *Architectural Principles for Virtual Computer Systems*, PhD thesis, Harvard University, February 1973.

- [Gol74] Robert P. Goldberg, *Survey of Virtual Machine Research*, IEEE Computer, pp 34–45, June, 1974.
- [Gor06] Gordon A. D., *V for Virtual*, Electronic Notes in Theoretical Computer Science, pp 177–181, 2006.
- [GT07] John Gerdes, Scott Tilley, *A conceptual overview of the virtual networking laboratory*, SIGITE '07 Proceedings of the 8th ACM SIGITE conference on Information technology education, pp 75-81, October 18–20, Destin, Florida, USA, 2007.
- [GVF08] Galán F, Lopez de Vergara J E, Fernández D, Muñoz R, *A Model-driven Configuration Management Methodology for Testbed Infrastructures*, IEEE/IFIP Network Operations and Management Symposium (NOMS 08), pp 747-750, Salvador da Bahia, Brazil, April 7-11, 2008.
- [HBS02] H. J. Hoxer, K. Buchacker, V. Sieh, *Implementing a User-Mode Linux with Minimal Changes from Original Kernel*, Proceedings of the 2002 International Linux System Technology Conference, pp 72– 82, September, 2002.
- [Hen98] C. Hendrick C., RFC 1058, *Routing Information Protocol*, The Internet Society, June 1988.
- [HHK03] Mark Handley, Orion Hodson, Eddie Kohler, *XORP: An Open Platform for Network Research*, ACM SIGCOMM Computer Communication Review, Vol. 33, Issue 1, pp 53-57, Berkeley, California, USA, January, 2003.
- [HN10] Kenneth Hess, Amy Newman, *Practical virtualization solutions : virtualization from the trenches*, Pearson Education Inc., Prentice Hall, USA, 2010.
- [HV04] Jeroen van der Ham, Gert Jan Verhoog, Virtual environments for networking experiments “Analytical Network Project” Masters System and Network Administration, University of Amsterdam, Netherlands, <http://staff.science.uva.nl/~vdham/research/publications/anp-jvdh-gjv.pdf>, July 1, 2004.
- [IPP97] Iglesias O, Paniagua C, Pessacq, *Evaluation of University Educational Software*, Computer Applications in Engineering Education, Vol. 5, Issue 3, pp 181–188, John Wiley & Sons Inc., 1997.
- [IS10] Curriculum Guidelines for Undergraduate Degree Programs in Information Systems, Association for Computing Machinery (ACM) and Association for Information Systems (AIS), <http://www.acm.org/education/curricula/IS%202010%20ACM%20final.pdf>, 2010.
- [Jov04] Nenad Jovanović, *Simulator računarske mreže zasnovan na web-u*, YUInfo, Kopaonik 2004.
- [JX03] Jiang, X., D. Xu., *vBET: a VM-Based Emulation Testbed*, Proceedings of the ACM SIGCOMM 2003 Workshops, pp 95–104, Karlsruhe, Germany, August, 2003.
- [KB03] B. Kneale, L. Box, *A virtual learning environment for real-world networking*, Informing Science, pp 71, Pori, Finland, June 2003.
- [KD00] Olaf Kirch, Terry Dawson, *Linux Network Administrator's Guide, 2nd Edition*, O'Reilly Media, June 2000.

- [KDC03] King, S.T., G.W. Dunlap, P.M. Chen, *Operating System Support for Virtual Machines*, Proceedings of the 2003 USENIX Annual Technical Conference, pp 71–84, San Antonio, Texas, June 2003.
- [KHB04] B. Kneale, A. Y. De Horta, I. Box, *Velnet: virtual environment for learning networking*, Proceedings of the 6th conference on Australian computing education ACE2004, Vol. 30, pp 161 – 168, Dunedin, New Zealand, 2004.
- [Kom00] Komar Brian, *Sams Tech Yuorself TCP/IP Network Administration in 21 Days*, Sams Publishing, 1999.
- [Kov10] Ivana Kostić Kovačević, MODELIRANJE SISTEMA, Singidunum revija, Vol.6 / No.2, pp 143-153, 2010.
- [KØ99] Komorowski J, Øhrn A., *Modelling prognostic power of cardiac tests using rough sets*, Artificial Intelligence in Medicine, Vol. 15, No. 2, pp 167–91, 1999.
- [Kra04] Arjen C. Krap, *Setting up a Virtual Network Laboratory with User-Mode Linux*, <https://alumni.os3.nl/~arjen/snb/asp/asp-report.pdf>, Universiteit van Amsterdam, 27th February, 2004.
- [KSR05] Kumar Krishna, Weiqing Sun, Pratik Rana, Tianning Li and R. Sekar, *V-NetLab: A Cost-Effective Platform to Support Course Projects in Computer Security*, Proceedings of 9th Annual Colloquium for Information Systems Security Education (CISSE 05), Atlanta, USA, Jun, 2005.
- [Kuz05] Wojciech Kuczborski, *A computer network laboratory based on the concept of virtual machines*, World Transactions on Engineering and Technology Education 2005, UICEE, Vol.4, No.1, Perth, Australia, 2005.
- [KVV09] Kappel J., Velte T. J., Velte A. T., *Microsoft® Virtualization with Hyper-V*, The McGraw-Hill Companies, 2009.
- [LA07] Cricket Liu, Paul Albitz, *DNS and BIND, 5th Edition*, O'Reilly Media, May 2006.
- [LC07] Robert Larson, Janique Carbone, *Microsoft Virtual Server 2005 R2 Resource Kit Chapter 7: Best Practices for Configuration and Performance*, Microsoft Press, Redmond, USA, August, 2007.
- [LJD08] Jaime Lloret, Jose M. Jimenez, Juan R. Diaz, Gines Lloret, *A Remote Network Laboratory to Improve University Classes*, 5th WSEAS / IASME International Conference on engineering education EE'08, pp 299-304, Heraklion, Greece, July 22-24, 2008
- [LL02] Ben Laurie, Peter Laurie, *Apache: The Definitive Guide, 3rd Edition*, O'Reilly Media, December 2002.
- [LMZ01] Liu S., Marti W., Zhao W., *Virtual Networking Lab (VNL): its concepts and implementation*, Proceedings of the 2001 American Society for Engineering Education Annual Conference and Exposition, Albuquerque, USA, 2001.



- [LRS03] Lorincz, K., Redwine, K., Sheh, A., *Stacking Virtual Machines – VMware and VirtualPC*, [www.eecs.harvard.edu/~konrad/References/notes/CS253-VMware-Report.pdf](http://www.eecs.harvard.edu/~konrad/References/notes/CS253-VMware-Report.pdf), March 18, 2003.
- [LS06] Lawson E, Stackpole W., *Does a Virtual Networking Laboratory Result in Similar Student Achievement and Satisfaction?*, Proceedings of SIGITE'06, pp 105-111, Minneapolis, Minnesota, USA, 2006.
- [LS07] Jean-Vincent Loddo, Luca Saiu, *Status report: marionnet or "how to implement a virtual network laboratory in six months and be happy"*, ML'07 Proceedings of the 2007 workshop on Workshop on ML, Freiburg, Germany, October 5, 2007.
- [LS08] Jean-Vincent Loddo, Luca Saiu, *Marionnet: a virtual network laboratory and simulation tool*, SimulationWorks, March 4, Marseille, France, 2008.
- [Lun08] Barry M. Lunt et al., Information Technology 2008, Curriculum Guidelines for Undergraduate Degree Programs in Information Technology Association for Computing Machinery (ACM) and IEEE Computer Society, <http://www.acm.org//education/curricula/IT2008%20Curriculum.pdf>, November 2008.
- [Lur05a] Kuthonuzo Luruo, *Virtual Networking with User-Mode Linux*, Linux for you pro, No. 27, March 2005.
- [Lur05b] Kuthonuzo Luruo, *Getting to Virtual Networking Nirvana*, Linux for you pro, No. 27, May 2005.
- [Mal98] G. Malkin, RFC 2453, *RIP Version 2*, November 1998
- [McE02] McEwan, W., *Virtual Machine Technologies and Their Application In The Delivery Of ICT*, Proceedings 15th Annual, Conference of the NACCQ, Hamilton, New Zealand, pp 55-62, 2002.
- [MM97] G. Malkin, R. Minnear, *RFC 2080, RIPng for IPv6*, The Internet Society, January 1997.
- [MMN09] Muthucumar Maheswaran, Alexis Malozemoff, Daniel Ng, Sheng Liao, Song Gu, Balasubramaniam Maniymaran, Julie Raymond McGill, Reehan Shaikh, Yuanyuan Gao, *GINI: a user-level toolkit for creating micro internets for teaching & learning computer networking*, SIGCSE '09 Proceedings of the 40th ACM technical symposium on Computer science education, March 3–7, 2009, Chattanooga, Tennessee, USA, 2009.
- [Moy98] J. Moy, RFC 2328, *OSPF Version 2*, The Internet Society, April 1998.
- [MPF06] S. Muir, L. Peterson, M. Fiuczynski, J. Cappos, J. Hartman, *Privileged operations in the PlanetLab virtualised environment*, Operating Systems Review, Vol. 40, No. 1, pp 75-88, January 2006.
- [MW05] Muller A., Wilson S., *Virtualization with VMware ESX Server*, Syngress Publishing, 2005.
- [NCE07] Nickerson J., Corter J., Esche S., Chassapis C., *A model for evaluating the effectiveness of remote engineering laboratories and simulations in education*, Computers & Education, Vol. 49, pp 708–725, Elsevier Science Ltd, 2007.

- [Net06] Netkit project (Web site), <http://www.netkit.org>, retrieved 2006.
- [Net11a] Netkit project (Web site) – section success stories [http://wiki.netkit.org/index.php/Success\\_stories](http://wiki.netkit.org/index.php/Success_stories), retrieved 2011.
- [Net11b] Netkit project (Web site) – section official labs, [http://wiki.netkit.org/index.php/Labs\\_Official](http://wiki.netkit.org/index.php/Labs_Official), retrieved 2011.
- [Nes11] NetSim (Web site), <http://tetcos.com/software.html>, 2011
- [NO00] Jason Nieh, Ozgur Can Leonard, *Examining Vmware*, Dr. Dobb's Journal, August 2000.
- [NO06] Jason Nieh, *Experiences Teaching Operating Systems Using Virtual Platforms and Linux*, ACM SIGOPS Operating Systems Review, pp 100-104, Vol. 40, Issue 2, April 2006.
- [Ns11] Ns-2 home page <http://www.isi.edu/nsnam/ns/edu/index.html>, pristupljeno 2011.
- [OD06] B. Odadzic, D. Dobrilovic, *Using Network Simulation Software in Teaching Routing Concepts*, Proceedings of 29th International Convention MIPRO 2006, Vol. II, CTI – Telecommunications & Information, pp 56-60, May 22-26, Croatia, Opatija, 2006.
- [Øhr99] Øhrn A., *Discernibility and Rough Sets in Medicine: Tools and Applications*, PhD thesis, Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway, 1999.
- [Opn11] OPNET technologies Inc. (Web site), <http://www.opnet.com/>, pristupljeno 2011.
- [Ora11] Oracle Corporation, *Oracle VM VirtualBox - User Manual*, <http://download.virtualbox.org/virtualbox/UserManual.pdf>, Version 4.0.0, 2004-2010, <http://www.virtualbox.org>, pristupljeno 2011.
- [Ora90] D. Oran, RFC 1142, *OSI IS-IS Intra-domain Routing Protocol*, February 1990.
- [PAC02] Larry Peterson, Tom Anderson, David Culler, Timothy Roscoe, *A Blueprint for Introducing Disruptive Technology into the Internet*, Proceedings of the First ACM Workshop on Hot Topics in Networking (HotNets), Princeton, USA, October 2002.
- [Paw97] Pawlak Z, *Rough set approach to knowledge-based decision support*, European Journal of Operational Research, Vol. 99, pp 48–57, 1997.
- [PBF06] Larry Peterson, Andy Bavier, Marc Fiuczynski, Steve Muir, *Experiences Building PlanetLab*, Proceedings of the Seventh Symposium on Operating System Design and Implementation (OSDI'06), pp 351–366, November 2006.
- [PD07] Larry L. Peterson, Bruce S. Davie, *Computer networks: a systems approach*, Morgan Kaufmann, 2007.
- [PDJ07] Valerie J. H. Powell, Christopher T. Davis, Randall S. Johnson, Peter Y. Wu, John C. Turchek, Ian W. Parker, *VLabNet: the integrated design of hands-on learning in information security and networking*, Proceedings of the 4th annual conference on Information security curriculum development (InfoSecCD'07), ACM New York, NY, USA, 2007.

- [Per98] Perrenoud, P., *From formative evaluation to a controlled regulation of learning processes, Towards a wider conceptual field*, Assessment in Education, Vol. 5, Issue 1, pp 85–102, 1998.
- [PGS95] Pawlak Z, Grzymala-Busse J, Slowinski R, Ziarko W, *Rough sets*, Association for Computing Machinery, Communications of the ACM, Vol. 38, No. 11, ABI/INFORM Global, pp 89, November, 1995.
- [Pla11] Planet Lab (Web site), <http://www.planet-lab.org/>, pristupljeno 2011.
- [PMS10] S. Papanastasiou, J. Mittag, E. G. Ström, H. Hartenstein, *Bridging the Gap Between Physical Layer Emulation and Network Simulation*, Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), pp 1-6, April 18-21, Sidney, Australia, 2010.
- [PP11] Raymond Panko, Julia Panko, *Business Data Networks and Telecommunications*, 8/E, Prentice Hall, 2011.
- [PR08] Pizzonia M, Rimondini M, *Netkit: easy emulation of complex networks on inexpensive hardware*, Proceedings of the 4th International Conference on Testbeds and research infrastructures for the development of networks & communities (TridentCom2008), Innsbruck, Austria, March 2008
- [PRB01] Piccoli G., Rami A., Blake I., *Web-based virtual learning environments: A research framework and a preliminary assessment of effectiveness in basic IT skills training*, Management Information Systems Quarterly, Vol. 25, No. 4, pp 401-426, 2001.
- [Pro11] Proftpd (Web site), <http://www.proftpd.org/>, pristupljeno 2011.
- [PS07] Pawlak Z, Skowron A., *Rudiments of rough sets*, An International Journal of Information Sciences, Vol. 177, pp 3–27, 2007.
- [PTR02] J. Potemans, J. Theunis, B. Rodiers, B. Van den Broeck, P. Leys, E. Van Lil, A. Van de Capelle, *Simulation of a Campus Backbone Network, a case-study*, Proceedings of OPNETWORK 2002, Washington D.C., USA, 2002.
- [PTT01] J. Potemans, J Theunis, M. Teughels, E. Van Lil and A. Van de Capelle, *Student Network Design Projects Using OPNET*, Proceedings of OPNETWORK 2001, Washington D.C., USA, 2001.
- [PTW07] Valerie J. H. Powell, John C. Turchek, Peter Y. Wu, Lawrence C. Franzi, Randall S. Johnson, Ian W. Parker, Christopher T. Davis, *VLabNet: A Virtual Laboratory Environment for Teaching Networking and Data Communications*, Proceedings of ISECON 2007, Vol. 24, pp 1-9, Pittsburgh, USA, 2007.
- [Pur04] Gregor N. Purdy, *Linux iptables Pocket Reference, Firewalls, NAT & Accounting*, O'Reilly Media, August 2004.
- [Qua11] Quagga Routing Suite (Web site), <http://www.quagga.net/>, pristupljeno 2011.
- [Ram11] Avinash Ramanath, *A Study of the interaction of BGP/OSPF in Zebra/ZebOS/Quagga*, Quagga official site, [http://www.quagga.net/docs/BGP\\_OSPF\\_Interaction\\_Report.pdf](http://www.quagga.net/docs/BGP_OSPF_Interaction_Report.pdf), pristupljeno 2011.

- [RFC971] *Internet Protocol Darpa Internet Program Protocol Specification*, September 1981.
- [Rim07] Rimondini M., *Emulation of Computer Networks with Netkit*, Technical Report RT-DIA-113-2007, Dipartimento di Informatica e Automazione, Universit`a di Roma Tre: Rome, Italy, 2007.
- [RLH06] Y. Rekhter, T. Li., S. Hares, RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*, The Internet Society, January 2006.
- [RR06] E. Rosen, Y. Rekhter, RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*, February 2006
- [RR09] Ruest Danielle, Ruest Nelson, *Virtualization: A Beginner's Guide*, Mc Graw Hill, USA, 2009.
- [Sad89] Sadler, D. R., *Formative assessment and the design of instructional systems*, Instructional Science, Vol. 18, Issue 2, pp 119–144, 1989.
- [Sad98] Sadler, R., *Formative assessment: Revisiting the territory*, Assessment in Education, Vol. 5, Issue 1, pp 77–84, 1998.
- [Sar06] Nurul Sarkar, *Tools for teaching computer networking and hardware concepts*, Information Science Publishing, London, UK, 2006.
- [SC05] Susanta Nanda, Tzi-cker Chiueh, *A Survey on Virtualization Technologies*, Research Proficiency report, <http://www.ecsl.cs.sunysb.edu/tr/TR179.pdf>, Februar 2005.
- [SC08] Stewart Brent D., Gough Clare, *CCNP BSCI – Official Exam Certification Guide*, Cisco Systems, Indianapolis, USA, 2008.
- [SD10] Stojanov Z., Dobrilovic D., *An approach to integration of maintenance services in educational Web portal*, 8th IEEE International Symposium on Intelligent Systems and Informatics SISY2010, pp 443 - 448, Septembar, Subotica, Serbia, 2010.
- [Sed01] Jeff Sedayao, *Cisco IOS Access Lists*, O'Reilly Media, June 2001.
- [Sha08] Shah A., *Kernel-based virtualization with KVM - deep virtue*, Linux Magazine, Issue 86, January, 2008.
- [SHA09] Kyle E. Stewart, Jeffrey W. Humphries, Todd R. Andel, *Developing a virtualization platform for courses in networking, systems administration and cyber security education*, Proceedings of the 2009 Spring Simulation Multiconference (SpringSim '09), Society for Computer Simulation International San Diego, CA, USA, 2009.
- [SKK08] Weiqing Sun, Varun Katta, Kumar Krishna, R. Sekar, *V-NetLab: An Approach for Realizing Logically Isolated Networks for Security Experiments*, Proceedings of (CSET'08), San Jose, USA, July 28 – August 1, 2008.
- [Slo02] Sloan J., *A Remotely Accessible Networking Laboratory*, The Journal of Computing Science inColleges, Vol. 18, No. 2, pp 215-222, December 2002.
- [SMR05] Steffen A., Marchionni E., Rayo P., *Advanced Network Simulation under User-Mode Linux*,

19. DFN-Arbeitstagung über Kommunikationsnetze, pp 321-333, Düsseldorf, Germany, 2005.
- [SPF07] Stephen Soltesz, Herbert Pötzl, Marc Fiuczynski, Andy Bavier, Larry Peterson, *Container-based Operating System Virtualization: A Scalable, High-performance Alternative to Hypervisors*, Proceedings of EuroSys '07, March 2007.
- [Sta04] William Stallings, *Data and Computer Communications*, 7/E, Prentice Hall, 2004
- [Sta05] William Stallings, *Business Data Communications*, 5/E, Prentice Hall, 2005.
- [Sta11] *Upotreba informaciono-komunikacionih tehnologija u republici Srbiji*, Republički zavod za statistiku Srbije, 2010.
- [Sto03] M. Stockman, *Creating Remotely Accessible "Virtual Networks" on a Single PC to Teach Computer Networking and Operating Systems*, Proceedings of the 4th conference on Information technology curriculum, pp 67 – 71, Lafayette, Indiana, USA, 2003.
- [SWF05] Ellen Siever, Aaron Weber, Stephen Figgins, Robert Love, Arnold Robbins, *Linux in a Nutshell, 5th Edition*, O'Reilly Media, July 2005.
- [SWL01] Sugerman, J., G. Venkitachalam, B.-H. Lim., *Virtualizing I/O Devices on VMware Workstation's Hosted Virtual Machine Monitor*, Proceedings of the 2001 USENIX Annual Technical Conference, 1–14, Boston, Massachusetts, USA, June, 2001.
- [Šeš71] Šešić, B. Opšta metodologija. Naučna knjiga: Beograd, 1971.
- [TBL02] J. Theunis, B. Van den Broeck, P. Leys, J. Potemans<sup>1</sup>, E. Van Lil, A. Van de Capelle, *OPNET in Advanced Networking Education*, Proceedings OPNETWORK 2002, Washington D.C., USA, August 26-30, 2002.
- [TGE08] Triantafillou E, Georgiadou E, Economides A, *The design and evaluation of a computerized adaptive test on mobile devices*, Computers & Education, Vol. 50, pp 1319–1330, 2008.
- [TIY10] Yuichiro Tateiwa, Tomohiro Iwasaki, Takami Yasuda, Naohisa Takahashi, *Evaluation of network construction exercise system LiNeS on the basis of heterogeneous and distributed virtual machine network composition function*, International Journal of Knowledge and Web Intelligence archive, Vol. 1, Issue 3/4, Inderscience Publishers, Geneva, Switzerland, July 2010.
- [TLP03] J. Theunis, P. Leys, J. Potemans<sup>1</sup>, B. Van den Broeck, E. Van Lil, A. Van de Capelle, *Advanced Networking Training for Master Students Through OPNET Projects*, Proceedings of OPNETWORK 2003, 2003.
- [Tul03] Mitch Tulloch, *IIS 6 Administration*, McGraw Hill, April, 2003.
- [Tul10] Tulloch M., *Understanding Microsoft Virtualization Solutions*, Microsoft Press, Redmond, USA, 2010.
- [Vin11] VINT Virtual InterNetwork Testbed (Web sites), <http://www.isi.edu/nsnam/vint/>, pristupljeno 2011.
- [Vmw11] VMware Inc. (Web site), <http://www.vmware.com/>, pristupljeno 2011.

- [Wai09] Wainer Gabriel A., *Discrete-event Modeling and Simulation - A Practitioner's Approach*, Taylor & Francis Group, Boca Raton, USA, 2009.
- [Wal02] Waldspurger, C.A., *Memory Resource Management in VMware ESX Server*, Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI'02), pp 181-194, Boston, MA, USA, December, 2002.
- [WG07] Williams D. E., Garcia J., *Virtualization with Xen*, Syngress Publishing, Elsevier, 2007.
- [WGG10] Klaus Wehrle, Mesut Günes, James Gross (Editors), *Modeling and Tools for Network Simulation*, Springer, 2010.
- [WH07] Wolf C, Halter E. M., *Virtualization - From the Desktop to the Enterprise*, Springer-Verlag, New York, USA, 2005.
- [WHY10] Xinli Wang, Guy C. Hembroff, Rick Yedica, *Using VMware VCenter lab manager in undergraduate education for system administration and network security*, Proceeding of ACM conference on Information technology education (SIGITE '10), pp 43-51, October 7-9, Midland, Michigan, USA, 2010.
- [WJ09] A. Wang and W. Jiang, *Research of Teaching on Network Course Based on NS-3*, First International Workshop on Education Technology and Computer Science (ETCS'09), Vol. 2, pp 629-632, 2009.
- [WLW09] E. Weingartner, H. vom Lehn, and K. Wehrle, *A performance comparison of recent network simulators*, Proceedings of the IEEE International Conference on Communications 2009 (ICC2009), Dresden, Germany, 2009.
- [WN09a] Muhammad Wannous, Hiroshi Nakano, *NVLab, a Networking Virtual Web-Based Laboratory that Implements Virtualization and Virtual Network Computing Technologies*, IEEE Transactions on Learning Technologies archive, Vol. 3, Issue 2, IEEE Computer Society Press Los Alamitos, CA, USA, April 2010.
- [WN09b] Muhammad Wannous, Hiroshi Nakano, *Supporting the Delivery of Learning-Contents with Laboratory Activities in Sakai Work-in-Progress report*, 2010 IEEE Global Engineering Education Conference (EDUCON), April 14-16, Madrid, Spain, 2010.
- [WNN11] Muhammad Wannous, Hiroshi Nakano, Takayuki Nagai, *To Know What They Are Doing in a Web-Based Laboratory, A System For Saving Information on The Learners' Activities Into Sakai's Logs*, Work-in-progress report, 2011 IEEE Global Engineering Education Conference (EDUCON), April 4 - 6, 2011, Amman, Jordan
- [Wor11] Internet World Stats, Miniwatts Marketing Group., <http://www.internetworldstats.com/stats.htm>, 2011.
- [WWG07] Ken Wong, Tilman Wolf, Sergey Gorinsky, Jonathan Turner, *Teaching experiences with a virtual network laboratory*, SIGCSE '07 Proceedings of the 38th SIGCSE technical symposium on Computer science education, ACM, New York, NY, USA, 2007.
- [Xor11] XORP (Web site), <http://www.xorp.org/>, 2011.

[Yan07] Li Yang, *Teaching system and network administration using virtual PC*, Journal of Computing Sciences in Colleges archive, Vol. 23, Issue 2, Consortium for Computing Sciences in Colleges, USA, December 2007.

[YH04] Sung Yoo, Scott Hovis, *Remote access internetworking laboratory*, Proceedings of the 35th SIGCSE technical symposium on Computer science education, pp 311-314, Norfolk, Virginia, USA, 2004.

[ZCC00] Elizabeth D. Zwicky, Simon Cooper, D. Brent Chapman, *Building Internet Firewalls, 2nd Edition*, O'Reilly Media, June 2000.

[Zeb11] GNU Zebra (Web site), <http://www.zebra.org/>, 2011.

## **RADOVI OBJAVLJENI U TOKU IZRADE DISERTACIJE**

[DS06] Dobrilović Dalibor, Stojanov Željko, Using virtualization software in operating systems course, Proceedings of the 4th IEEE International Conference on Information Technology: Research and Education - ITRE 2006, pp. 222-226, ISBN: 1-4244-0859-8, Tel Aviv, Israel, 16-19 October 2006, DOI: 10.1109/ITRE.2006.381569

[DO06a] Dobrilovic Dalibor, Odadzic Borislav, Virtualization Technology as a Tool for Teaching Computer Networks, PROCEEDINGS OF WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY, VOL 13, vol. 13, str. 126-130, 2006.

[DO06b] Dobrilović Dalibor, Odadžić Borislav, Simulacija računarskih mreža na open-source platformi, 11<sup>th</sup> INFOTECH 2006, 12-16. Jun, Srbija, Vrnjačka Banja, 2006.

[DO06c] Dobrilović D., Odadžić B., OSPF protokola u virtuelnoj mrežnoj laboratoriji, 5<sup>th</sup> Infoteh-Jahorina 2006, Vol. 5, Ref. E-II-1, pp 334-338, ISBN-99938-624-2-8, mart 22-24, Bosnia & Herzegovina, Jahorina 2006.

[DS008] Dobrilović Dalibor, Stojanov Željko, Odadžić Borislav, Implementing BGP-4 Protocol Scenario in VNLab Open Networking Environment, Proceedings of the IEEE 6th International Symposium on Intelligent Systems and Informatics, ISBN 978-1-4244-2407-8. Subotica, Serbia. September 26-27, 2008, DOI: 10.1109/SISY.2008.4664933

[DBB09] Dobrilovic, D., Brtko, V., Berkovic, I. and Odadzic, B. (2011), Evaluation of the virtual network laboratory exercises using a method based on the rough set theory. Computer Applications in Engineering Education, 19: n/a. doi: 10.1002/cae.20370

[DSD09] Dobrilovic Dalibor, Stojanov Zeljko Duka Milan Odadzic Borislav, Application for network node configuration based on NNDL (Network Node Description Language) (Proceedings Paper), 2009 7TH INTERNATIONAL SYMPOSIUM ON INTELLIGENT SYSTEMS AND INFORMATICS, 2009, vol. , pp 303-307

[DS011] Dobrilovic D., Stojanov Z., Odadzic B., Advancement in Online Education: Exploring the Best Practices. Volume 1, Chapter 10. Design and Implementation of Online Virtual Network Laboratory, ISBN: 978-1-61470-897-1, Pub. Date: 2011 4th quarter.

[OD06] B. Odadžić, D. Dobrilović, Using Network Simulation Software in Teaching Routing Concepts, Proceedings of 29th International Convention MIPRO 2006, Vol. II, CTI – Telecommunications & Information, pp 56-60, ISBN 953-233-019-4 (ed. Stjepan Golubić , Branko Mikac, Vlasta Hudek), May 22-26, Croatia, Opatija, 2006.



## Summary

In this Thesis the research is focused on modeling and realization of virtual network laboratory based on virtualization technology. This work is motivated with the growing development of ICT technologies, the rapid growth of Internet users and importance of ICT infrastructure and TCP/IP networks in this development. Together with the importance of computer network environments and related technologies, the importance of educated and well trained experts in this field is growing too. One of the main focus points in this research is the area of engineering education, especially in the field of computer networks and data communications. This problem is rather complex. The leading idea is to obtain and construct the efficient environment for training experts in this fields.

One approach in creation of such learning environments is the creation of real physical laboratories with real network equipment. Although this approach provides the environment with the highest level of reality and similarity with the real systems it has its own disadvantages. The first disadvantage is the high prices for network equipment needed for completion of such laboratory. The second is its low level of adaptability to the new technologies and costly upgrade. That's why the usage of alternative environments has its importance.

The alternative approach relies mainly on simulation software, special kind of software developed to simulate network environments. Despite the existence of highly real and developed software in this field, the level of its usability in the learning process is still questionable. That's why another approach, creation of virtual network environments based on virtualization technology, become quite popular recently.

Virtualization technology is not a new one. Its beginning dates back to 60s of the last century. In that time the need for virtualization lies in existence of huge but very expensive systems. The percentage of its utilization was very important factor. The virtualization technology allows emulation of multiple virtual machines that run on the single physical one. That was the way to increase the utilization of the expensive computers. Eventually, with the development of the computer systems, especially the personal computers, their cost decreased rapidly. This also decreased the need for virtualization technology. Today, the virtualization technology has its revival with new and alternative appliance. One of its appliances is usage of the technology in creation of virtual network laboratories.

Virtualization software emulates the complete hardware of the hosting physical machine. Being capable of emulation of host's network interface cards, the virtualization technology supports virtual connection between those interfaces and in that way it allows creation of virtual networks.

The appliance of this technology for creation of virtual network laboratories used in experiments, testing and education starts in past several years. The modeling and realization of virtual network laboratory that can be used in the courses of computer networks and other related courses at universities are the central point of this research. Together with the modelling of the environment and its effectiveness, the network educational scenarios are also modeled as well. At the end, the method used for evaluation of this environment is also results of this work.

This thesis is structured as follows. In the first chapter is introduction (Chapter 1.). In this chapter is explained the main motivation for this research and the reasons that influenced definition of the research goal. The structure of the Thesis is also presented here.

Second chapter is focused on the goals and methodology of this research. First, the problem and area of research is defined. Goal and the research tasks are defined next. The main research hypothesis is pointed out as well as the sub hypotheses of the work. The main hypothesis is: *It is possible to create efficient virtual network laboratory based on virtualization technology, like the environment for emulation of computer networks with local and remote access, suitable for usage in university courses.* The sub hypotheses are: *it is possible to create virtual network laboratory based on virtualization technology, created laboratory presents realistic emulation of real network systems, it is possible to use this laboratory within computer networks related course at universities and virtual network laboratory can be implemented as a part of distance learning system.* The expected research results, methods and techniques of research, number of student participated in the researches and instruments for research metrics are presented as well.

Third section describes the three important components of the virtual laboratories, as well as related work in this field. The first section in this chapter (Chapter 3.1) makes brief explanation of virtualization technology, its main characteristics, history, types and architectures and available software on the market. The second section (Chapter 3.2.) presents the existing solutions in the world for virtual network laboratories used for testing and education, their main characteristics and ways of usage. In the last section (Chapter 3.3.); software components needed for creation of virtual network laboratories are presented. The main software component in this environment is multiprotocol routing software, and it is presented in greater detail.

Fourth chapter describes first phase of the research where the basis of virtual network laboratory are made. Considering the experiences from the similar projects, and as well as from own Computer Network courses, the core network of the virtual laboratory is made. As a beginning point for this are used network scenarios for simulation software OPNET ITGuru Academic Edition. Created scenarios (Chapter 4.2.) are used in the course in the environment with the local access. The results of its usage are presented as well as the main disadvantages in its architecture. The usage of laboratory is described with performance tests of computer running virtual laboratory (Chapter 4.4.) and with the results of student questionnaires (Chapter 4.5.). Student answers showed that they accepted virtual network laboratory as efficient tool for leaning computer networks and it is shown that they prefer this laboratory slightly more than OPNET IT Guru Academic Edition software as a discrete event network simulation software.

The next chapter describes the methodology for creation and portability of existing local access environment to the remotely accessible environment. The architecture of the laboratory is presented, as well as the hardware and software components of the laboratory (Chapter 5.1). Other components of the system such as supporting portal (Chapter 5.2), design and administration issues (Chapter 5.3.) are presented in the following sections. The presented system VNLab is additionally described with characteristic network scenarios (Chapter 5.4.–5.9.) and leaning configurations for IPv6 static routing and addressing, OSPF and BGP-4 dynamic routing protocol configuration and with system security and firewall configuration.

The sixth chapter is subdivided in several sections and it is focused on evaluation of effectiveness of VNLab system. The introduction presents existing approaches in evaluation of virtual network environments and similar learning systems (Chapter 6.1). The first evaluation of VNLab is conducted using newly developed methodology based on rough set theory (Chapter 6.2.). This methodology is one of the scientific results of this work and allows lecturer to analyze student work and to find out the differences between various levels of student knowledge. Also, it allows lecturer to identify parts in exercise material which is not understandable by students. Next section (Chapter 6.3.) presents the results of student achievements during two control tests conducted after laboratory usage in Computer Network course. In following section (Chapter 6.4), the method of evaluation and improvement of laboratory scenarios is done using formative assessment. At the end of chapter (Chapter 6.5.) usability of VNLab environment in various Computer Science based curriculums is shown by comparing VNLab supported features with recommended curriculums of ACM (Association of Computer Machinery), IEEE (International organization of Electrical and Electronic Engineering) and AIS (Association of Information Systems) organizations for 4 areas of computer science: Computer Science, Computer Engineering, Information Technology and Information Systems.

In the last chapter (Chapter 7.) the resume of the research is performed. First, the summarization of period and number of students that took part in the exams are presented. Local access virtual network laboratory is created and used during the first, initial phase, of the research.

The research highlights are as follows. During the school year 2005/2006 three laboratory exercises are used and during school year 2006/2007 eight laboratory exercises are completed. In the finish of the second year, new environment with remote access is created – VNLab. Immediately, three first three exercises are adapted to the new environment. VNLab is used in 9 laboratory exercises during school years 2007/2008, 2008/2009, 2009/2010 and 2010/2011. The laboratory is used within Computer Networks Course.

Together with the implementation of the VNLab in learning process, the method for its evaluation based on rough set theory is developed. Method is applied to data acquired from the testing of 55 students. The 70 IF-THEN rules are generated using program Rosetta. These rules had precisely pointed out three important indicators. First, they showed difference in the knowledge of the students with the highest grades which helped in upgrade and improvement of the exercises. These rules helped to locate the easiest part of exercises which are understandable for all students and also to locate the most un-understandable parts of exercise allowing the lecturer to modify exercise material or to modify way of grading students.

The second techniques used in laboratory evaluation are two control tests. First test is performed during school season 2009/2010 and 48 students took part in it. The second test is performed in school year 2010/2011 with 126 participating students. The analyzed parameters are percentage of achieved points and number of students achieved positive result. In the first test percentage of achieved points is 77% and percentage of students that passed exam is 96%. In the second test results are 54% and 56% respectively. The differences between these two results are understandable because the first test is announced in advance and the second test is performed almost without announcement. Both results show that the laboratory is efficient environment that gives opportunity to the student to learn network concepts well.

Evaluation and improvement of laboratory exercises is performed using formative assessment. Assessment is made with classwork task performed at the end of each exercise. The structure of these tasks is given in Annexes together with laboratory material. Assessment is performed during three years (2007/2008, 2008/2009 and 2009/2010). The number of students participated in assessment is 189. The student answers were written on the paper and analyses of their answers helped in identification of the bad parts, errors and mistakes in the laboratory materials. This increased the effectiveness of the laboratory, because lecturer was able to improve the part of exercises that are not understandable by the students. It was very important because some concepts can be implemented in more than one exercise so they can be improved for the next exercise in the same school year, or for the next generation of students.

At the end, the usage of the laboratory is evaluated by comparison with the recommended curriculums for five educational profiles in the field of Computer Science. These curriculums are joined work of organizations ACM, IEEE and AIS for undergraduate studies. The following curriculums and subject are compared with the features supported with the VNLab: Information Technology (learning module: Networking – NET), Computer Science (learning module: Net Centric Computing – NC), Computer Engineering (module Computer Networks CE-NWK), Information Systems (learning module: IT Infrastructure). The analyzes showed that parts of the curriculums are cover with the VNLab and that VNLab is applicable in the laboratory exercises within these curriculums.

According to the presented results, the main hypothesis of the research is confirmed. Furthermore, the all other sub hypothesis of this research are confirmed as well, except the sub hipotesis that this virtual network laboratory model and its realization may be implemented in the distance learning environment. This last sub hipotesis is only partially confirmed by this research because the VNLab system is not integrated in a distance learning environment, but still allows remote access and usage of the laboratory. After the resume of the research results the negative features of the VNLab are outline as well as the further work.

## REČNIK MANJE POZNATIH REČI I IZRAZA

### **ABR (Area Border Router)**

Granični ruter područja. Predstavlja ruter kod OSPF protokola koji komunicira sa drugim područjem.

### **Area – područje**

U protokolu OSPF to je grupa rutera koji razmenjuju informacije o rutama.

### **ARP (Address Resolution Protocol)**

Protokol internet sloja koji se koristi za razrešavanje IP adresa i MAC adresa na mrežnom interfejsu.

### **AS (Autonomous System)**

Autonomni sistem. Grupa rutera i mreža koje podpadaju pod jednu administrativnu nadležnost koja međusobno saraduje na propagiranju informacija o dostupnosti i rutama u mreži pomoću unutrašnjeg protokola mrežnog prolaza koji sami odaberu. Da bi dva autonomna sistema mogla da komuniciraju, jedan ruter iz svakog sistema šalje ruteru iz drugog sistema informacije.

### **Atribut odluke (eng. *decision attribute*)**

Uobičajeno jedinstveni atribut sa binarnim vrednostima. Ovo je atribut po čijim se vrednostima objekti univerzuma svrstavaju u klase.

### **Backbone ruter**

OSPF ruter koji poseduje interfejs ka backbone području (okosnici).

### **BGP (Border Gateway Protocol)**

Protokol graničnog mrežnog prolaza. Glavni spoljašnji protokol mrežnih prolaza koji se koristi na Internetu. Pojavilo se više verzija ovog protokola, a trenutna verzija koja se koristi je BGP-4.

### **Default ruta**

Podrazumevana ruta (putanja). Stavka u tabeli rutiranja koja se koristi za sva odredišta koja nisu navedena u toj tabeli. Tabele rutiranja u većini slučajeva sadrže ovaj podatak.

### **Dinamičko rutiranje (eng. *Dynamic routing*)**

Mehanizam koji omogućava dinamičku konfiguraciju svih putanja za rutiranje preko mreže. Ovo se postiže primenom protokola za rutiranje koji objavljuju promene u mrežnoj topologiji.

### **Distance vector rutiranje**

Klasa protokola za rutiranje koji koriste distribuirani algoritam najkraćeg puta (SPF) u kojem svaki ruter šalje svojim susedima listu mreža koje može da dosegne i rastojanje do svake od njih.

### **Donja aproksimacija grubog skupa (eng. *lower approximation*)**

Sadrži elemente koji sigurno pripadaju skupu.

**EGP (Exterior Gateway Protocol)**

Spoljašnji protokol mrežnog prolaza. Naziv za svaki protokol koji ruter u jednom autonomnom sistemu koristi objavljivanje dostupnosti mreže drugom autonomnom sistemu. BGP-4 je spoljašnji protokol mrežnog prolaza koji se najviše koristi.

**Firewall**

Mrežna barijera (vatreni zid). Alat za zaštitu rutera, servera i unutrašnje mreže.

**Gateway – mrežni prolaz**

Uređaj koji povezuje dva heterogena sistema, ili više njih, i obavlja prenos podataka između njih.

**Granični region (eng. *boundary region*)**

Razlika između gornje aproksimacije skupa i donje aproksimacije skupa. U granični region spadaju elementi za koje se ne može sa sigurnošću tvrditi da pripadaju skupu.

**Grubi skup (eng. *rough set*)**

Skup čije granice nisu jasno definisane. Za razliku od fazi skupova gde svi elementi univerzuma pripadaju fazi skupu sa određenom merom, kod grubih skupova postoji verovatnoća pripadanja objekta skupu. Grubi skup je definisan pomoću gornje aproksimacije skupa, donje aproksimacije skupa i graničnog regiona skupa.

**Gornja aproksimacija grubog skupa (eng. *upper approximation*)**

Sadrži elemente donje aproksimacije ali i one elemente koji pripadaju skupu sa određenom verovatnoćom.

**Gostujući operativni sistem (eng. *Guest Operating System* ili *Guest OS*)**

Operativni sistem koji je pokrenut u okruženju virtuelne mašine, a koji se može postaviti i na stvarne računare.

**Host – računar**

Računarski sistem povezan na mrežu.

**If-Then pravila u smislu teorije grubih skupova**

Pravila u kojima je pretpostavka formirana konjukcijom deskriptora atributa iz redukt skupa, a zaključak je formiran pomoću deskriptora atributa odluke.

**IGP (Interior Gateway Protocol)**

Unutrašnji protokol mrežnog prolaza. Ime kojim se obuhvataju svi protokoli koji se koriste za propagiranje informacija o dostupnosti i rutama unutar autonomnog sistema. OSPF i RIP se najčešće koriste kao IGP protokoli.

**Informacioni sistem u smislu teorije grubih skupova (eng. *information system*)**

Definisan je pomoću uređene četvorke  $(U, Q, V, f)$ , gde je  $U$  univerzum,  $Q$  skup atributa,  $V$  skup vrednosti atributa,  $f$  funkcija koja povezuje attribute sa njihovim vrednostima.

**Internal router**

OSPF ruter čiji su svi mrežni interfejsi konektovani sa mrežama koje pripadaju istom području.

**Kondicioni atributi (eng. *condition attributes*)**

Atributi objekata univerzuma pomoću kojih su objekti opisani. Pomoću ovih atributa se definišu gornja aproksimacija, donja aproksimacija i granični region.

**Link state routing – usmeravanje na osnovu stanja veza**

Jedan od dva pristupa koje primenjuju protokoli usmeravanja, a u kojem ruteri difuzno upućuju poruke o stanju i koriste Dijkstra algoritam za izračunavanje najkraćih putanja.

**LSA (Link State Advertisement)**

Obaveštenje u vidu poruke koje se šalje mreži radi ažuriranja OSPF link state baze podataka. Ova obaveštenja obuhvataju svaki OSPF interfejs na ruteru kojim su povezane mreže na ruter i cenu (cost) za slanje u svaku od priključenih mreža.

**MAC adresa (Media Access Control)**

Fizička adresa mrežnog adaptera.

**Maska podmreže (eng. *subnet mask*)**

Bit maska koja se koristi da bi se u IP adresi označili bitovi koji odgovaraju podmreži. Maska ima 32 bita u verziji protokola IPv4. Jedinice su u delu adrese koji identifikuje mrežu, a nule u delu koji identifikuje računar (host).

**Metrika (eng. *metric*)**

Cena ili vrednost koja je dodeljena mrežnom linku. Koristi se za utvrđivanje putanje (rute) koja ima najnižu cenu do odredišta. Uobičajeni parametri za opis metrike su broj skokova (eng. *hop*), kašnjenje, propusni opseg, pouzdanost i komunikacioni troškovi.

**NAT (Network Address Translation)**

Adresna translacija. Tehnologija koja omogućava da računari sa privatnim adresama komuniciraju sa spoljnom mrežom.

**Nerazberivost, nerazlučivost (eng. *indiscernibility*)**

Nemogućnost razlikovanja objekata prema informacijama koje o njima znamo. Objekti su nerazberivi ako ih karakterišu (opisuju) iste ili slične osobine (atributi).

**Netconfig**

Program koji se koristi u UNIX okruženju radi ažuriranja kernela, kada se izvrši izmena u konfiguraciji hosta.

**Netstat**

Komanda koja omogućuje administratoru mreže da ispita upotrebu portova i konekcije na portove.

**Packet filter firewall**

Fukcija firewall sistema koja analizira saobraćaj na mreži na transportnom sloju OSI modela. Svaki dolazni i odlazni paket se upoređuje sa skupom pravila, na osnovu kojih će se paket primiti ili odbaciti.

**PING**

Program koji se u TCP/IP mrežama koristi za testiranje dostupnosti odredišta slanjem ICMP zahteva za ehom i čekanjem odgovora.

**POP (Post Office Protocol)**

Protokol koji se koristi za pristupanje i preuzimanje elektronske pošte iz poštanskog sandučeta.

**Port protokola**

Apstrakcija koju transporni protokoli koriste za razlikovanje različitih odredišta u datom računaru. Portovi se označavaju pozitivnim celim brojevima. Neki portovi su rezervisani za standardne servise.

**Redukt (eng. *reduct*)**

Podskup kondicionih atributa. Redukt skup sadrži one kondicione atribute čije vrednosti imaju uticaj na vrednost atributa odluke.

**Relacija nerazberivosti, relacija nerazlučivosti (eng. *indiscernibility relation*)**

Binarna relacija, najčešće relacija ekvivalencije (refleksivna, simetrična i tranzitivna). Često se koristi relacija sličnosti. U svakom slučaju, osobina refleksivnosti je zadržana, osobine simetričnosti i tranzitivnosti ne moraju važiti.

**RFC (Request for Comment)**

Serijski dokumenti koji sadrže preglede, merenja, ideje, tehnike i zapažanja, kao i standarde za TCP/IP skup protokola.

**RIP (Routing Information Protocol)**

Protokol za informacije o usmeravanju. Protokol koji se koristi za propagiranje informacija o putevima unutar autonomnog sistema. Postoji u tri verzije: RIPv1, RIPv2 i RIPng.

**Ruta – putanja (eng. *route*)**

Ruta je putanja kojom mrežni saobraćaj prolazi od izvora do odredišta i nazad. U TCP/IP mrežama svaki datagram se usmerava nezavisno. Putanje se mogu menjati dinamički.

**Ruter – usmerivač (eng. *router*)**

Uređaj koji povezuje dve ili više mreža i prenosi IP saobraćaj između njih. Ruter u tabeli usmeravanja traži odredišnu adresu datagrama da bi odredio gde treba usmeriti paket (sledeći skok).

**Sistem odluke (eng. *decision system*)**

Informacioni sistem u kome su definisani kondicioni atributi i atribut odluke.

**Shortest path routing – rutiranje najkraćom putanjom**

Rutiranje u kojem se datagrami usmeravaju najkraćom putanjom. Pronalazak najkraće putanje vrši protokol rutiranja.



**Statičko rutiranje (eng. *static routing*)**

Manuelna konfiguracija svih putanja za rutiranje paketa preko mreža. Primenjuje se u manjim mrežama.

**Tabela rutiranja (eng. *Routin table*)**

Baza podataka koju koristi ruter za utvrđivanje pravca kojim se šalju IP datagrami.

**TELNET**

Standardni servis udaljenog terminala u TCP/IP mrežama. Dozvoljava korisniku na jednom hostu da pristupi udaljenom sistemu na drugom hostu kao da su korisnik i njegova tastatura direktno povezani na udaljenom računaru.

**Teorija grubih skupova (eng. *rough set theory*)**

Teoriju grubih skupova je definisao poljski istraživač Z. Pavlak osamdesetih godina prošlog veka. Originalno, ova teorija koristi relaciju nerazberivosti koja je ekvivalencija.

**Traceroute**

Program koji pokazuje put do odredišta, tj. putanju ka odredišnoj mreži ili hostu. Program prikazuje svaki ruter na toj putanji.

**Tunelovanje (eng. *tunneling*)**

Tehnika kojom se enkapsulira paket u protokolu višeg nivoa i prenosi kroz transportni sistem. VPN koristi tunelovanje za prenos šifrovanih datagrama između dva hosta.

**Virtual Machine Monitor (VMM)**

Softver koji radi na sloju između hipervizora ili host operativnog sistema i jedne ili više virtuelnih mašina i koji omogućava apstrakciju virtuelne mašine gostujućem operativnom sistemu. Sa punom virtuelizacijom, VMM pruža apstrakciju virtuelne mašine identičnu kao i stvarna mašina (računar), tako da operativni sistemi (npr. Windows 2000, Windows Server 2003, Linux itd.) mogu da se pokrenu i rade kao da se nalaze na samoj stvarnoj mašini (računaru).

**Virtuelna mašina (eng. *Virtual Machine*)**

Reprezentacija stvarnog računara koja koristi softver koji omogućava operativno okruženje koje može da pokrene gostujući operativni sistem (Guest OS).

## IP adresiranje i statičko rutiranje – II deo

### Cilj vežbe

Cilj vežbe je upoznavanje studenata sa osnovnim konceptima rutiranja, statičkim rutiranjem i IP adresiranjem.

### Rutiranje (podsetnik)

Rutiranje ([routing](#)) predstavlja usmeravanje paketa u TCP/IP mrežama. To je proces u kojem se vrši odabir putanje u mreži preko koje će se slati paketi i vršiti saobraćaj. Rutiranje usmerava pakete od njihovog izvorišta do odredišta, tj. do hosta sa odgovarajućom logičkom adresom (IP adresom). Rutiranje vrše posebni uređaji nazvani [ruteri \(router\)](#). Ruteri mogu biti specijalizovani mrežni uređaji koje proizvode Cisco, Juniper, 3com ili računar sa više mrežnih interfejsa.

Ruter prosleđuje pakete na osnovu definisanih pravila za usmeravanje – [tabele rutiranja \(routing table\)](#), koja se još naziva i Routing Information Base (RIB). Tabela rutiranja sadrži podatke o putanjama do različitih mreža. Svaki ulaz u tabeli rutiranja ima nekoliko osnovnih parametara, a to su:

- [ciljna adresa](#) - adresa mreže ili adresa hosta,
- [mrežna maska](#) mreže ili hosta,
- [adresa sledećeg rutera](#) na putanji do cilja ([gateway](#)),
- izlazni mrežni interfejs i
- metric – cena (cost) rute

### Dinamičko i statičko rutiranje

Postoje dve vrste rutiranja – [statičko i dinamičko rutiranje](#). Protokoli kao što su [RIP](#) (sa verzijama RIPv1, RIPv2, RIPv3), [OSPF](#) (sa verzijama OSPFv2, OSPFv3) [BGP](#), IGRP, EIGRP su protokoli za dinamičko rutiranje i omogućuju dinamičku razmenu podataka potrebnih za rutiranje. Pored dinamičkog rutiranja postoji i statičko rutiranje. Kod statičkog rutiranja se vrši ručno definisanje ruta za svaku mrežu. U ovoj vežbi će se raditi samo statičko rutiranje.

### Proces rutiranja

Ruter prilikom usmeravanja paketa konsultuje tabelu rutiranja. U nastavku će biti opisana pojednostavljena verzija te tabele. Tabela rutiranja se sastoji od po jednog zapisa za svaku

mrežu koju ruter poznaje. Svaki taj zapis sadrži adresu odredišne mreže sa mrežnom maskom, sledeći čvor kome je potrebno proslediti paket (gateway) i izlazni interfejs. Primer je dat u Tabeli 1.

**Tabela 1. Primer pojednostavljene tabelle rutiranja**

Odredišna mreža	sledeći čvor (gateway)	izlazni interfejs
192.168.2.0/30	192.168.2.150	eth1
161.53.65.0/24		eth0
192.168.2.0/24		eth1
161.53.0.0/16	161.53.65.1	eth0

U ovoj tabeli prva kolona predstavlja adresu odredišne mreže. Adresa je data zajedno sa mrežnim prefiksom. Druga kolona predstavlja sledeći čvor kome je potrebno proslediti paket. Ako za neku mrežu ne postoji podatak u toj koloni tada je ta mreža direktno povezana sa ruterom. Treća kolona određuje izlazni interfejs kroz koji je potrebno poslati paket da bi on stigao do odredišta.

Pretraživanje tabelle se vrši po najdaljem preklapanju mrežne adrese. Takođe treba napomenuti da su ulazi u tabeli rutiranja poređani po opadajućem mrežnom prefiksu. Na vrhu se nalazi mreža koja u svojoj adresi ima mrežni prefiks od 30 bita, a na dnu se nalazi mreža koja ima samo 16 bita u mrežnom delu. Pretraživanje kreće od prvog reda i prestaje u trenutku kada se utvrdi poklapanje.

Rute prema mrežama kojima pripadaju interfejsi rutera postavljaju se automatski. Rute prema udaljenim mrežama (mrežama kojima se pristupa preko *gateway-a*) potrebno je uneti ručno pomoću komande `route` (Linux sistemi). U prethodnom primeru druga i treća ruta se postavljaju automatski, a prva i četvrta ruta se unose pomoću komandi:

```
route add -net 192.168.2.0 netmask 255.255.255.252 gw 192.168.2.150 dev eth1
route add -net 161.53.0.0 netmask 255.255.0.0 gw 161.53.65.1 dev eth0
```

Na Cisco IOS sistemima komanda za unos rute izgleda ovako:

```
ip route 192.168.2.0 255.255.255.252 192.168.2.150
```

gde je prvo navedena ciljna adresa mreže, zatim mrežna maska, a na kraju gateway.

Na Windows sistemima, komanda za unos rute izgleda ovako:

```
ROUTE ADD 192.168.2.0 MASK 255.255.255.252 192.168.2.150
```

## Default ruta

Zbog toga što se u malim tabelama rutiranja ne mogu nalaziti podaci o svim rutama, postoji poseban tip ruta nazvanih [default ruta](#) ili [podrazumevana ruta](#) (putanja). Default ruta služi da se na nju usmeri paket koji je usmeren na IP adresu, tj. IP mrežu koja se ne nalazi u tabeli

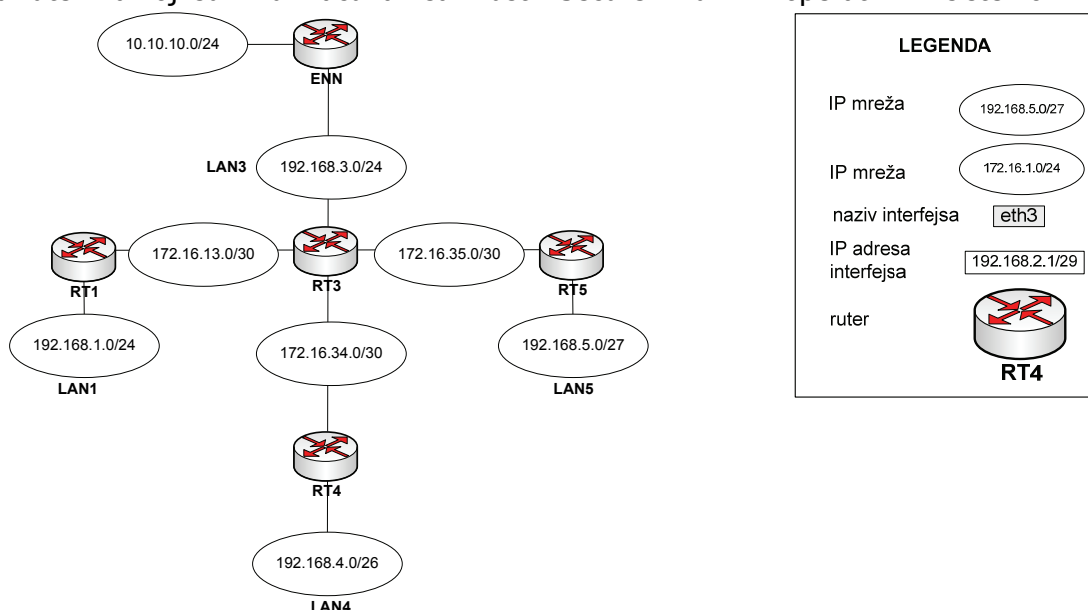
rutiranja. Default ruta se označava u nekim slučajevima IP adresom 0.0.0.0. Komanda za postavljanje default rute na Linux sistemima je

```
route add default gw 192.168.2.2 dev eth0
```

## Zadatak

Na slici 1. je prikazana topologija mreže sa 5 rutera i 7 mreža koje su povezane preko njih. U vežbi je potrebno izvršiti konfigurisanje rutera dodelom odgovarajućih IP adresa njegovim interfejsima i unosom potrebnih statičkih ruta.

Na ruterima koji su Linux računari sa Trustix Secure Linux 2.2 operativnim sistemom.



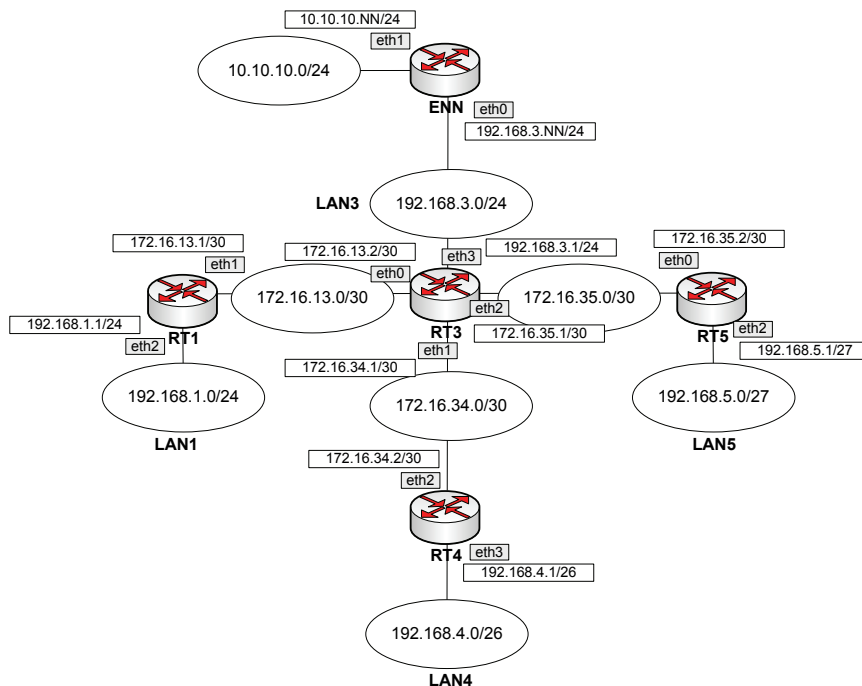
Slika 1.

Na sledećoj slici prikazana je detaljnija šema mreže, sa logičkim imenima interfejsa, sa jednim dodatnim ruterom ENN. U vežbi je potrebno izvršiti konfiguraciju rutera ENN.

Napomena: logičko ime ethernet interfejsa na Linux sistemima za Ethernet mrežni interfejs je eth0, eth1, eth2 itd.

IP adrese interfejsa rutera date su u tabeli

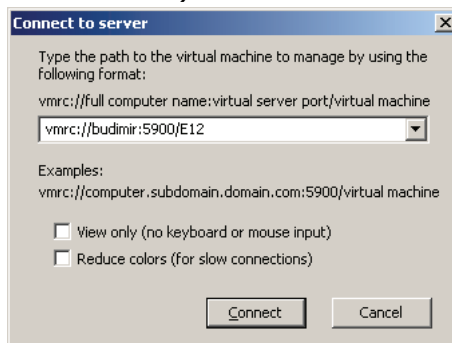
Čvor	IP adresa	Interfejs
RT1	172.16.13.1/30	eth1
RT1	192.168.1.1/24	eth2
RT3	172.16.13.2/30	eth0
RT3	172.16.34.1/30	eth1
RT3	172.16.35.1/30	eth2
RT3	192.168.3.1/24	eth3
RT4	172.16.34.2/30	eth2
RT4	192.168.4.1/26	eth3
RT5	172.16.35.2/30	eth0
RT5	192.168.5.1/27	eth2



Slika 2.

## Logovanje na ruter

Logovanje na ruter koji je potrebno konfigurisati se vrši programom vmrc.exe koji se nalazi na studentskom disku (Z disk - folder KRM).

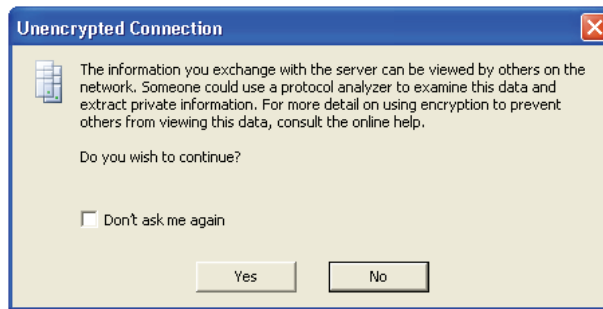


Kada se program startuje potrebno je upisati `vmrc://budimir:5900/ENN`

*NAPOMENA: Svako se loguje na drugi ruter, npr. E11, E12, E13 ... E20 itd.*

U sledećem koraku bi trebalo da se otvore dijalog prozori koji upozoravaju na sigurnost.

Na ovakve i slične dijalog prozore, ukoliko se otvore, odgovoriti sa OK. Može se čekirati box "Don't ask me again" ukoliko ne želite da se da se ti prozori ponovo otvore pri sledećem pokušaju pristupa.



Kada se otvori prozor za logovanje potrebno je uneti

```
username: itguru  
password: akademik
```



Kada se pojavi terminal rutera potrebno je prijaviti se na sistem. Prijava na sistem vrši se tako što se na mestu predviđenom za upis korisničkog imena (npr. **RT login:** ) upisuje korisničko ime **root** (administratorski nalog na Unix/Linux sistemima ) i pritiska taster **[Enter]**. Zatim se na mestu predviđenom za upis lozinke (**Password:** ) unosi lozinka **123456** i pritiska taster **[ Enter ]**. Unešena lozinka se ne vidi na ekranu radi sigurnosti.

## Konfigurisanje rutera

### Korak 1.

Pre početka rada potrebno je dodeliti odgovarajuće IP adrese mrežnim interfejsima. IP adrese na Linux sistemima dodeljuju se interfejsima komandom **netconfig**, npr.

```
root@RT1 ~# netconfig --ip=192.168.3.NN --netmask=255.255.255.0 -d eth0
```

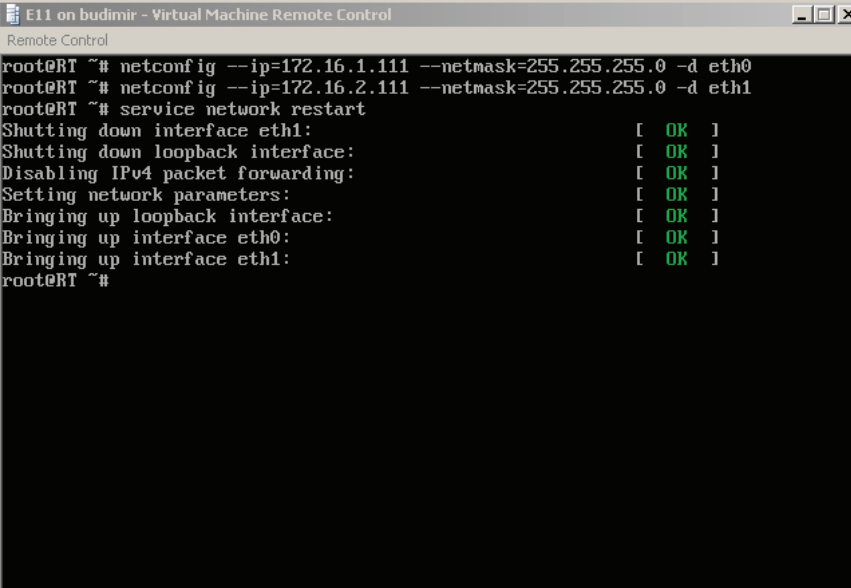
```
root@RT1 ~# netconfig --ip=10.10.10.NN --netmask=255.255.255.0 -d eth1
```

*NAPOMENA: na ruteru E11 IP adrese su 192.168.3.11 i 10.10.10.11, na ruteru E12 IP adrese su 192.168.3.11 i 10.10.10.12, tj. 192.168.3.NN i 10.10.10.NN za svaki ENN.*

Primena unetih parametara i dodela IP adresa interfejsima se inicira komandom **service network restart [Enter]**, koja prvo obori interfejse, pa ih kasnije podigne sa novim parametrima.

```
root@RT1 ~# service network restart  
Shutting down interface eth0: [ OK ]  
Shutting down interface eth1: [ OK ]  
Shutting down loopback interface: [ OK ]
```

```
Setting network parameters:      [ OK ]
Bringing up loopback interface:  [ OK ]
Bringing up interface eth0:      [ OK ]
Bringing up interface eth1:      [ OK ]
```



```
Remote Control
root@RT ~# netconfig --ip=172.16.1.111 --netmask=255.255.255.0 -d eth0
root@RT ~# netconfig --ip=172.16.2.111 --netmask=255.255.255.0 -d eth1
root@RT ~# service network restart
Shutting down interface eth1:      [ OK ]
Shutting down loopback interface:  [ OK ]
Disabling IPv4 packet forwarding:  [ OK ]
Setting network parameters:        [ OK ]
Bringing up loopback interface:    [ OK ]
Bringing up interface eth0:        [ OK ]
Bringing up interface eth1:        [ OK ]
root@RT ~#
```

Komandom `ifconfig eth0` i `ifconfig eth1` možemo proveriti da li su IP adrese ispravno dodeljene interfejsima. Komanda `ifconfig` zajedno sa nazivom interfejsa (npr. eth0) daje prikaz podataka o interfejsu kao što su IP adresa (inet addr), MAC adresa (Hwaddr), mrežna maska (Mask), broadcast adresa (Bcast) i statistiku ukupno prenešenih podataka.

```
root@RT1 ~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:04:75:F3:68:DE
          inet addr: 192.168.3.111 Bcast: 192.168.3.255  Mask:255.255.255.0
          inet6 addr: fe80::204:75ff:fef3:68de/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:152594130 errors:0 dropped:0 overruns:0 frame:0
          TX packets:153031871 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:449482470 (428.6 MiB)  TX bytes:392848524 (374.6 MiB)
          Interrupt:10 Base address:0xdc00
```

NAPOMENA: Komanda `ifconfig` navedena bez naziva posebnog interfejsa prikazuje sve interfejse na računaru

## Korak 2.

Sada se može proveriti tabela rutiranja. To se radi komandom `route -n`. Budući da nisu unešene statičke rute, tabela rutiranja na ruteru ENN ima samo dve rute. To su rute ka mrežama koje su direktno povezane na taj ruter, tj. ka mrežama kojima pripadaju njegova dva interfejsa i te rute se postavljaju automatski. Treća je loopback ruta (127.0.0.0).

```
root@RT1 ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
```

10.10.10.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
192.168.3.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0	lo

**NAPOMENA:** Kao što se vidi u prethodnom primeru, rute ka mrežama koje su direktno povezane na taj ruter, tj. rute ka mrežama kojima pripadaju njegovi interfejsi, automatski se postavljaju u tabelu rutiranja i njih nije potrebno posebno unositi. Rute se postavljaju samo prema mrežama koje nisu direktno povezane na taj isti ruter.

### Korak 3.

Sada će se proveriti da li postoji komunikacija sa hostovima van mreže 192.168.3.0/24. Npr. Sa hostovima u mreži 172.16.35.0/30. Provera se vrši komandom **ping** i navođenjem jedne od IP adresa (npr. 172.16.35.1) iz te mreže koja je dodeljena nekom hostu.

```
root@RT ~# ping 172.16.35.1
connect: destination host unreachable
```

Poruka koja se dobija ukazuje na to da ne postoji komunikacija, što je normalno jer ne postoji ruta ka toj mreži.

Ruta se dodaje sledećom komandom

```
root@RT ~# route add -net 172.16.35.0 netmask 255.255.255.252 gw 192.168.3.1
```

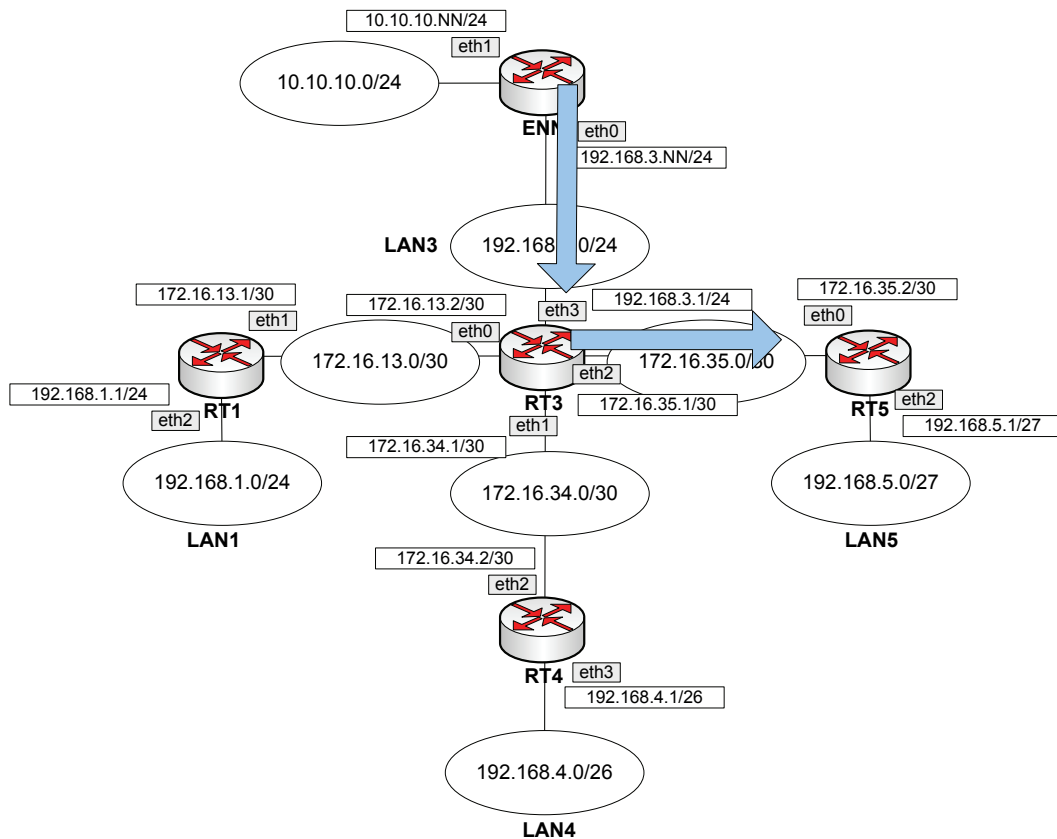
Mreža prema kojoj se postavlja ruta je 172.16.35.0. Mrežna maska se može izračunati iz mrežnog prefiksa /30, što znači da mrežna maska ima 30 jedinica u binarnom obliku 11111111.11111111.11111111.11111100. To iznosi 255.255. 255.252.

Gateway za mrežu 172.16.35.0 je adresa najbližeg interfejsa sledećeg rutera koji se nalazi na putanji do odredišne mreže (vidi sliku 3.), tj. adresa preko koje je potrebno poslati paket da bi stigao do odredišta. To je interfejs eth3 na ruteru RT3 sa IP adresom 192.168.3.1/24.

Sada je ponovo potrebno proveriti sadržaj tabele rutiranja komandom **route -n**.

```
root@RT1 ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
172.16.35.0 192.168.3.1 255.255.255.252 UG 0 0 0 eth0
10.10.10.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1
192.168.3.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```





Slika 3.

Ako je sve ispravno konfigurisano trebala bi da postoji komunikacija između ENN i 172.16.35.1. To se proverava komandom `ping 172.16.35.1` sa ENN. Prikaz se prekida tasterom `Ctrl+C`.

```
root@RT ~# ping 172.16.35.1
PING 172.16.35.1 (172.16.35.1) 56(84) bytes of data.
64 bytes from 172.16.35.1: icmp_seq=1 ttl=62 time=32.9 ms
64 bytes from 172.16.35.1: icmp_seq=2 ttl=62 time=20.0 ms
64 bytes from 172.16.35.1: icmp_seq=3 ttl=62 time=15.0 ms
64 bytes from 172.16.35.1: icmp_seq=4 ttl=62 time=9.97 ms
```

--- 172.16.35.1 ping statistics ---

5 packets transmitted, 5 received, 0% packet loss, time 4042ms  
rtt min/avg/max/mdev = 9.979/19.702/32.917/7.643 ms

#### Korak 4.

Kao i u prethodnoj vežbi moraju se postaviti rute prema svim mrežama koje nisu povezane direktno na ruter ENN, da bi taj ruter imao komunikaciju sa računarima unutar njih (to su sve mreže osim 10.10.10.0/24 i 192.168.3.0/24). Zato što to može da bude zamoran posao, upoznaćemo se sa konceptom default rute. Default ruta (podrazumevana ruta) služi da se na nju usmeri paket koji je usmeren na IP adresu, tj. IP mrežu koja se ne nalazi u tabeli rutiranja. Znači, svi paketi koji su usmereni na IP adrese koje se na nalaze u tabeli rutiranja šalju se preko ove rute.

Da bi smo proverili da li to i kako funkcioniše prvo ćemo **obrisati** prethodno unetu rutu

```
root@RT ~# route del -net 172.16.35.0 netmask 255.255.255.252
```

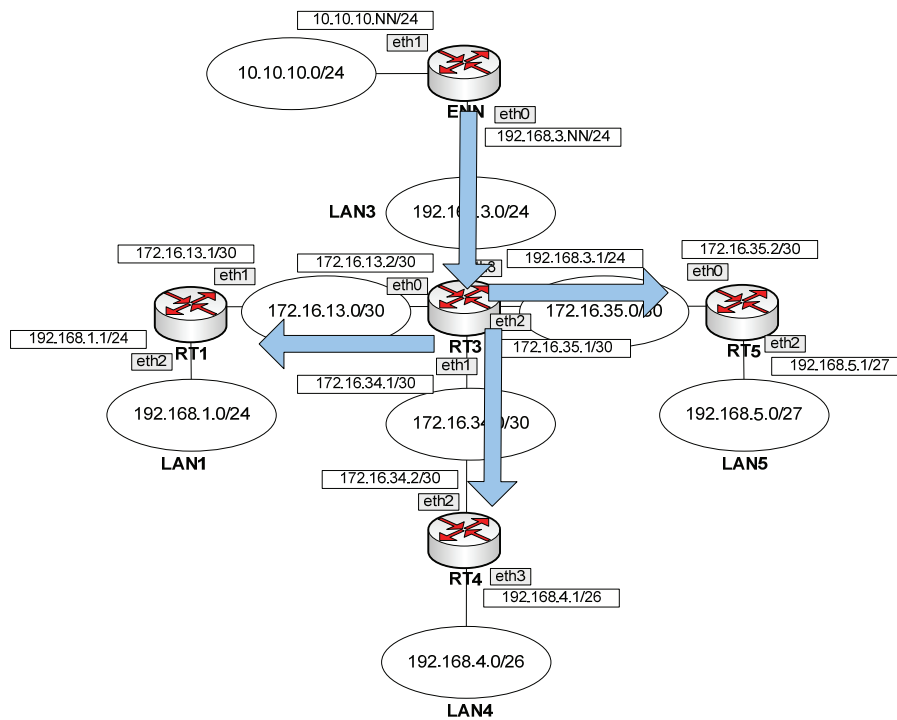
Sada tabela rutiranja treba da izgleda kao i na početku

```
root@RT1 ~# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
10.10.10.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1
192.168.3.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```

## Korak 5.

Unos default rute se vrši komandom

```
root@RT ~# route add default gw 192.168.3.1
```



Slika 4.

Prilikom unosa default rute za gateway se koristi prvi interfejs sledećeg rutera koji se nalazi na putu do spoljašnjih mreža. To je eth3 interfejs rutera RT3 sa IP adresom 192.168.3.1/24 (vidi sliku 4.).

Sada tabela rutiranja treba da izgleda kao i na početku

```
root@RT1 ~# route -n
Kernel IP routing table
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
10.10.10.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
192.168.3.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0	lo
0.0.0.0	192.168.3.1	0.0.0.0	UG	0	0	0	eth0

Kako radi mreža možemo proveriti na dva načina. Prvo, pomoću komande **ping** i IP adrese bilo kog udaljenog hosta npr. 192.168.4.1.

```
root@RT ~# ping 192.168.4.1
PING 192.168.4.1 (192.168.4.1) 56(84) bytes of data.
64 bytes from 192.168.4.1: icmp_seq=1 ttl=62 time=32.9 ms
64 bytes from 192.168.4.1: icmp_seq=2 ttl=62 time=20.0 ms
64 bytes from 192.168.4.1: icmp_seq=3 ttl=62 time=15.0 ms
64 bytes from 172.16.35.1: icmp_seq=4 ttl=62 time=9.97 ms

--- 172.16.35.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4042ms
rtt min/avg/max/mdev = 9.979/19.702/32.917/7.643 ms
```

Ili komandom **traceroute -n 192.168.4.1**

Ovom komandom se dobija celokupna putanja do odredišne IP adrese.

### Zadatak 1.

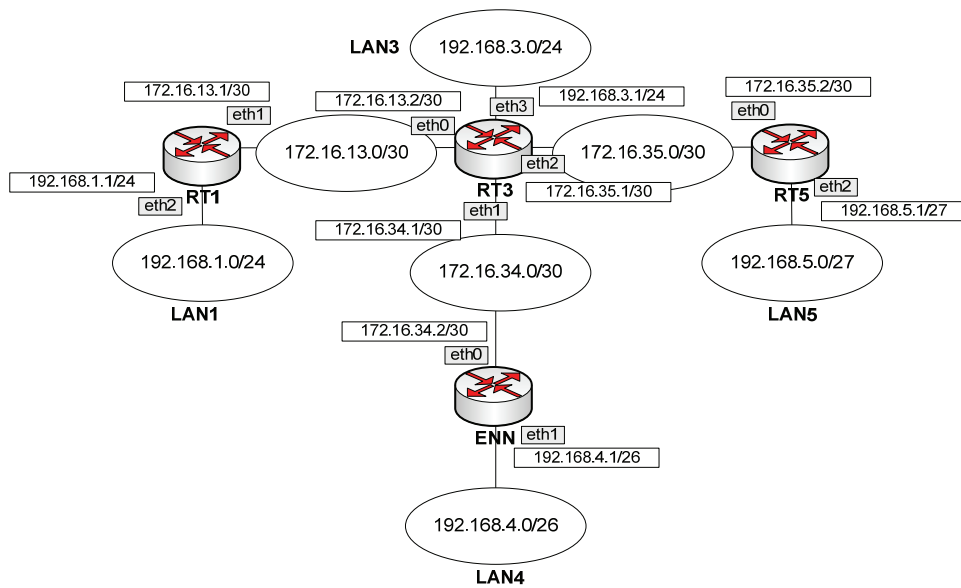
Napisati komande za postavljanje svih potrebnih ruta (bez default rute) na ruteru ENN. Za postavljanje rute prema nekoj mreži potrebna je **adresa te mreže, njena mrežna maska i gateway (adresa interfejsa rutera koji vodi to odredišne mreže)**.

Adresa mreže se može videti sa slike 5. Rute se postavljaju samo prema mrežama koje nisu direktno povezane sa ruterom ENN.

Mrežna maska se lako može izračunati pomoću mrežnog prefiksa (npr. mrežni prefiks /29 se može predstaviti kao IP adresa u binarnom obliku sa 29 binarnih jedinica 11111111.11111111.11111111.11111000).

Gateway je adresa najbližeg interfejsa sledećeg rutera koji se nalazi na putanji od rutera ENN do odredišne mreže, tj. adresa preko koje je potrebno poslati paket da bi stigao do odredišta.

Rute je potrebno predstaviti u pismenom obliku na papiru.



Slika 5.

## Zadatak 2.

Napisati komandu koja je potrebna za unos default rute na ruteru ENN (Slika 5.).

## Završna razmatranja

U vežbi je demonstrirano kako se vrši konfiguracija rutera dodelom IP adresa i postavljanjem statičkih ruta i default rute.

## IPv6 adresiranje i statičko rutiranje

### Cilj vežbe

Cilj vežbe je da upozna studente sa IPv6 adresiranjem i statičkim rutiranjem.

### IPv6

Zbog nedostatka adresnog prostora došlo je do razvoja IPv6 protokola. Osnovna razlika između IPv4 i IPv6 protokola je dužina mrežna adrese. **IPv6 adrese** se sastoje od **128 bita** (kao što je definisano RFC 4291 dokumentom), dok se **IPv4 adrese** sastoje od 32 bita. Adresni prostor IPv4 protokola ima otprilike 4 milijardi adresa, dok IPv6 ima prostora za  $3.4 \times 10^{38}$  jedinstvenih adresa.

IPv6 adrese se sastoje od dva logička dela: 64-bitnog mrežnog prefiksa, i 64-bitnog dela za adresiranje hosta. **IPv6 adresa** se predstavlja sa **osam grupa od po četiri heksadecimalne cifre razdvojenih dvotačkama (:)**.

Na primer,

```
2001:0db8:85a3:08d3:1319:8a2e:0370:7334
```

predstavlja ispravnu IPv6 adresu.

Adrese se mogu pisati i u pojednostavljenom obliku tako što se jedna ili **više grupa nula (0000)** mogu izostaviti i zameniti **dvostrukim dvotačkama (::)**, npr. adresa

```
2001:0db8:0000:0000:0000:0000:1428:57ab
```

se može predstaviti kao

```
2001:0db8::1428:57ab
```

Svaka spojena grupa nula (0000) može se smanjiti na dve dvotačke samo jedan put u adresi, tj. **nije dozvoljeno imati dva ili više puta po dve dvotačke**. Primer nedozvoljenog skraćivanja adrese

```
2001:0000:0000:FFD3:0000:0000:0000:57ab
```

```
2001::FFD3::57ab
```

**Vodeće nule se takođe mogu izostaviti, kao što je prikazano u primeru**

```
2001:0db8:0000:0000:0000:0000:1428:57ab
```

```
2001:0db8:0000:0000:0000::1428:57ab
```

```
2001:0db8:0:0:0:0:1428:57ab
```

2001:0db8:0:0::1428:57ab  
2001:0db8::1428:57ab  
2001:db8::1428:57ab

Sekvenca od 4 zadnja bajta u adresi se može predstaviti decimalnim ciframa koristeći tačke kao separator. Ovakva notacija se koristi zbog kompatibilnosti sa IPv4 adresama. Adresna šema je u tom slučaju x:x:x:x:x.d.d.d, gde su x heksadecimalne cifre, a d decimalne cifre, npr. ::ffff:12.34.56.78 je isto što i ::ffff:0c22:384e i 0:0:0:0:0:ffff:0c22:384e.

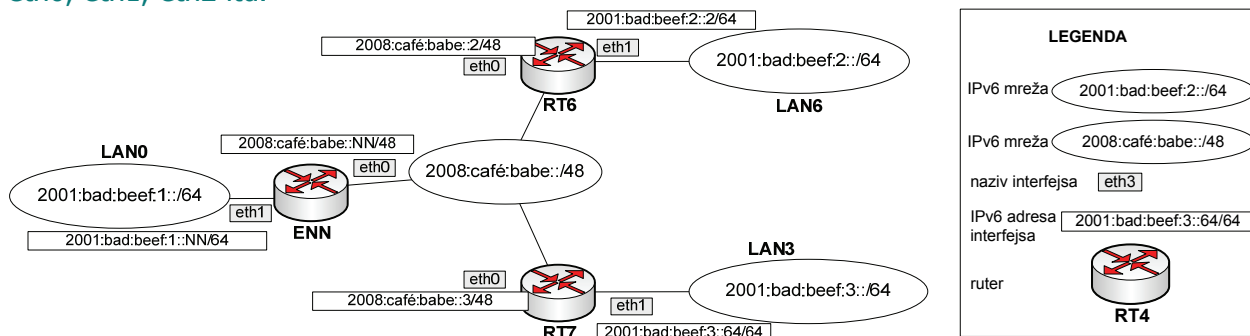
Dodatne informacije o IPv6 se mogu naći u RFC 4291 dokumentu - IP Version 6 Addressing Architecture.

## Zadatak

Na slici je prikazana topologija mreže. U vežbi je potrebno izvršiti konfigurisanje rutera dodelom odgovarajućih IPv6 adresa njegovim interfejsima i unosom potrebnih statičkih ruta na ruterima koji su Linux računari sa Trustix Secure Linux 2.2 operativnim sistemom.

U vežbi je potrebno izvršiti konfiguraciju rutera ENN na slici 1.

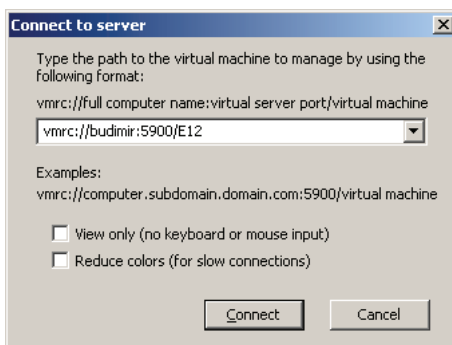
Napomena: logičko ime ethernet interfejsa na Linux sistemina za Ethernet mrežni interfejs je eth0, eth1, eth2 itd.



Slika 1.

## Logovanje na ruter

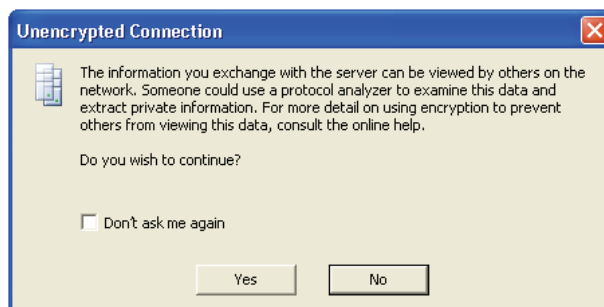
Logovanje na ruter koji je potrebno konfigurisati se vrši programom vmrc.exe koji se nalazi na studentskom disku (Z disk - folder KRM).



Kada se program startuje potrebno je upisati vmrc://budimir:5900/ENN

*NAPOMENA: Svako se loguje na drugi ruter, npr. E11, E12, E13 ... E20 itd.*

U sledećem koraku bi trebalo da se otvore dijalog prozori koji upozoravaju na sigurnost. Na ovakve i slične dijalog prozore, ukoliko se otvore, odgovoriti sa OK. Može se čekirati box "Don't ask me again" ukoliko ne želite da se ti prozori ponovo otvore pri sledećem pokušaju pristupa.



Kada se otvori prozor za logovanje potrebno je uneti:

```
username: itguru  
password: akademik
```



Kada se pojavi terminal rutera potrebno je prijaviti se na sistem. Prijava na sistem vrši se tako što se na mestu predviđenom za upis korisničkog imena (npr. **RT login:** ) upisuje korisničko ime **root** (administratorski nalog na Unix/Linux sistemima ) i pritiska taster **[Enter]**. Zatim se na mestu predviđenom za upis lozinke (**Password:** ) unosi lozinka **123456** i pritiska taster **[ Enter ]**. Unešena lozinka se ne vidi na ekranu radi sigurnosti.

## Konfigurisanje rutera

### Korak 1.

Pre početka rada potrebno je učitati modul za podršku IPv6 protokolu. Na ovim Trustix sistemima nalazi se kernel v2.4.31 koji podržava IPv6 protokol ali samo kao spoljni modul. To se radi komandom:

```
root@RT1 ~# modprobe ipv6
```

Zatim je potrebno dodeliti odgovarajuće IP adrese mrežnim interfejsima. IP adrese na Linux sistemima dodeljuju se interfejsima komandom `ip -6 addr add <ipv6address>/<prefixlength> dev <interface>`, npr

```
root@RT1 ~# ip -6 addr add 2008:cafe:babe::NN/48 dev eth0
root@RT1 ~# ip -6 addr add 2001:bad:beef:1::NN/64 dev eth1
```

*NAPOMENA: na ruteru E11 IP adresa je 2008:cafe:babe::11, na ruteru E12 IP adresa je 2008:cafe:babe::12 itd.*

Komandom `ifconfig eth0` i `ifconfig eth1` možemo proveriti da li su IP adrese ispravno dodeljene interfejsima. Komanda `ifconfig` zajedno sa nazivom interfejsa (npr. eth0) daje prikaz podataka o interfejsu kao što su IP adresa (inet addr), MAC adresa (Hwaddr), mrežna maska (Mask), broadcast adresa (Bcast) i statistiku ukupno prenešenih podataka. Pored IP adrese koju smo mi dodelili biće prikazana i IP adresa koja je **automatski dodeljena** interfejsu, a koja je izračunata na osnovu MAC adrese računara.

```
[root@zpupin root]# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:11:09:8D:28:3D
          inet addr:172.16.1.30  Bcast:172.16.1.255  Mask:255.255.255.0
          inet6 addr: 2008:cafe:babe::1/48  Scope:Global
          inet6 addr: fe80::211:9ff:fe8d:283d/64  Scope:Link

          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:52244677  errors:0  dropped:0  overruns:0  frame:0
          TX packets:41424761  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:704605347 (671.9 Mb)  TX bytes:89394363 (85.2 Mb)
          Interrupt:11 Base address:0x8c00

[root@zpupin root]# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:02:44:8F:6F:EC
          inet addr:172.16.2.30  Bcast:172.16.2.255  Mask:255.255.255.0
          inet6 addr: 2001:bad:beef:1::111/64  Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:37793481  errors:1  dropped:2  overruns:1  frame:0
          TX packets:55776157  errors:0  dropped:0  overruns:20  carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4231953583 (4035.9 Mb)  TX bytes:356804493 (340.2 Mb)
          Interrupt:10 Base address:0xb800
```

**NAPOMENA:** Komanda `ifconfig` navedena bez naziva posebnog interfejsa prikazuje sve interfejse na računaru

Prikaz podešenih IP adresa se može dobiti i komandom:

```
root@RT1 ~# ip -6 addr show
```

Prikaz parametara pojedinačnog interfejsa se dobija komandom:

```
root@RT1 ~# ip -6 addr show dev eth0
```



## Korak 2.

Sada se vrši provera tabele rutiranja. To se radi komandom `ip -6 route show`. Budući da nisu unešene statičke rute, tabela rutiranja na ruteru ENN ima samo rute koje se postavljaju automatski ka mrežama koje su direktno povezane na njih. To su rute ka mrežama koje su direktno povezane na taj ruter, tj. ka mrežama kojima pripadaju njegova dva interefejsa i te rute se daju automatski.

```
root@RT1 ~# ip -6 route show
2008:cafe:babe::/48 dev eth0 metric 256 mtu 15000 advmss 1440
2001:bad:beef:1::/64 dev eth1 metric 256 mtu 15000 advmss 1440
fe80::/64 dev eth0 metric 256 mtu 1500 advmss 1440
fe80::/64 dev eth1 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth0 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth1 metric 256 mtu 1500 advmss 1440
default dev eth0 proto kernel metric 256 mtu 1500 advmss 1440
default dev eth1 proto kernel metric 256 mtu 1500 advmss 1440
unreachable default dev lo proto none metric -1 error -101
```

**NAPOMENA:** Kao što se vidi u prethodnom primeru, rute ka mrežama koje su direktno povezane na taj ruter, tj. rute ka mrežama kojima pripadaju njegovi interfejsi se automatski postavljaju u tabelu rutiranja i njih nije potrebno posebno unositi. Rute se postavljaju samo prema mrežama koje nisu direktno povezane na taj isti ruter.

## Korak 3.

Zbog toga što ne postoji ruta ka mreži `2001:bad:beef:3::/64`, rutu je potrebno uneti. Rute se unose komandom `ip -6 route add <ipv6network>/<prefixlength> via <ipv6address>`, gde je `<ipv6address>` adresa gatewaya. U ovom slučaju je to `2008:cafe:babe::3` ili adresa `eth0` interefejsa na ruteru RT7 (Slika 1.).

```
root@RT1 ~# ip -6 route add 2001:bad:beef:3::/64 via 2008:cafe:babe::3
```

Nova tabela rutiranja sada treba da izgleda ovako:

```
root@RT1 ~# ip -6 route show
2008:cafe:babe::/48 dev eth0 metric 256 mtu 15000 advmss 1440
2001:bad:beef:1::/64 dev eth1 metric 256 mtu 15000 advmss 1440
2001:bad:beef:3::/64 via 2008:cafe:babe::3 dev eth0 metric 1024 mtu 15000 advmss
1440
fe80::/64 dev eth0 metric 256 mtu 1500 advmss 1440
fe80::/64 dev eth1 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth0 metric 256 mtu 1500 advmss 1440
ff00::/8 dev eth1 metric 256 mtu 1500 advmss 1440
default dev eth0 proto kernel metric 256 mtu 1500 advmss 1440
default dev eth1 proto kernel metric 256 mtu 1500 advmss 1440
unreachable default dev lo proto none metric -1 error -101
```

A komunikacija sada treba da postoji:

```
root@RT ~# ping6 2001:bad:beef:3::64
```

```
64 bytes from 2001:bad:beef:3::64: icmp_seq=1 ttl=62 time=32.9 ms
64 bytes from 2001:bad:beef:3::64: icmp_seq=2 ttl=62 time=20.0 ms
64 bytes from 2001:bad:beef:3::64: icmp_seq=3 ttl=62 time=15.0 ms
64 bytes from 2001:bad:beef:3::64: icmp_seq=4 ttl=62 time=9.97 ms
```

## Tasterom Ctr+C prekida se rad komande ping

### Pokrenuti komandu

```
root@RT ~# ping6 2001:bad:beef:3::64 -c 5 -s 20000
```

Da li postoji razlika u izvršavanju komande ping6 posle unetih parametara i kakva je razlika?

### Korak 4.

Proveriti putanje do zadatih mreža sa komandama traceroute6 i tracepath6. Na primer:

```
root@RT ~# tracepath6 2001:bad:beef:3::64
```

### Korak 5.

Prethodno uneta ruta se briše komandom:

```
ip -6 route del 2001:bad:beef:3::/64 via 2008:cafe:babe::3
```

Ruta se može postaviti i sledećom komandom:

```
root@RT ~# route -A inet6 add 2001:bad:beef:3::/64 gw 2008:cafe:babe::3
```

### Korak 6.

ND (Neighbor discovery protocol) je u IPv6 okruženju ekvivalent ARP protokola u IPv4 okruženju. On mapira IP adrese u MAC adrese. [Prikaz neighbour tabele se vrši komandom.](#)

```
root@RT ~# ip -6 neigh show
```

NDP sam popunjava sadržaj tabele. Sledeća komanda pomaže da se sadržaj unosi ručno. Komanda asocira IPv6 adresu hosta 2001:bad:beef:3::64 sa njegovom fizičkom MAC adresom

```
root@RT ~# ip -6 neigh add 2001:bad:beef:3::64 lladdr AA:01:C2:D3:74:05 dev eth0
```

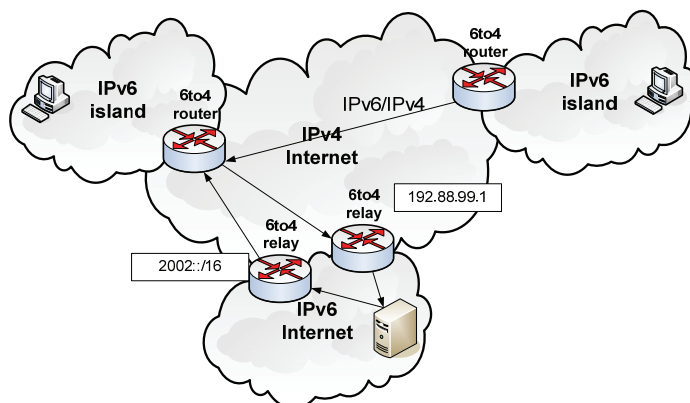
[Ubaciti umesto date MAC adrese MAC adresu računara na kome se nalazite.](#)

### Korak 7.

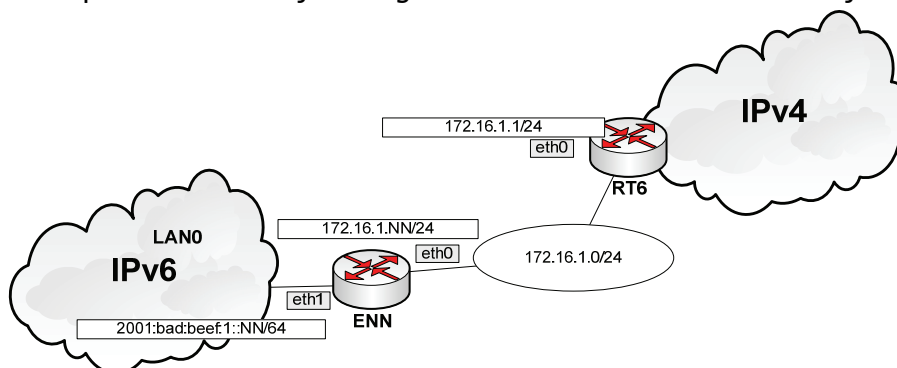
Tuneliranje IPv4/IPv6

Tuneliranje predstavlja proces u računarskim mrežama kada se mrežni protokol (protokol isporuke) koristi da bi enkapsulirao različite protokole koji prenose podatke. Tuneliranje u IPv4/IPv6 mrežama se može vršiti na više načina. Jedan od tih načina je 6to4.

6to4 (RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds) je Internet mehanizam tranzicije namenjen migraciji sa IPv4 na IPv6 mreže. Taj sistem omogućava IPv6 paketima da se prenesu preko IPv4 mreža (uglavnom se misli na IPv4 Internet). 6to4 se može koristiti za individualne hostove ili lokalne IPv6 mreže.



6to4 tuneliranje koristi jednostavan mehanizam za kreiranje automatskih tunela. Linux operativni sistem podržava kreiranje konfigurisanih tunela sa sit tunel-interfejsom.



Prvo se vrši provera postojećih tunela komandom `ip -6 tunnel show`.

```
root@RT ~# ip -6 tunnel show
sit0: ipv6/ip remote any local any ttl 64 nopmtudisc
```

Dodavanje i podešavanje tunela se sastoji iz nekoliko faza. Prvo se radi kreiranje interfejsa koji se označava sa sit1 i podešavanje IPv4 adrese komandom `ip tunnel add sit1 mode sit remote <router-ipv4> local <host-ipv4>`.

Gde je `<router-ipv4>` IPv4 adresa udaljenog rutera koji predstavlja krajnju tačku veze – tzv. tunnel endpoint, a `<host-ipv4>` lokalna IPv4 adresa rutera koja povezuje lokalni ruter sa udaljenim. Ako se upotrebi parametar any tada se koristi bilo koja lokalna adresa rutera (tj. sve adrese postojećih interfejsa na ruteru)

```
root@RT ~# ip tunnel add sit1 mode sit ttl 64 remote 172.16.1.1 local any
```

Link se podiže sa komandom:

```
root@RT ~# ip link set sit1 up
```

Sada se kreirani tunnel može videti komandom:

```
root@RT ~# ip -6 tunnel show
sit0: ipv6/ip remote any local any ttl 64 nopmtudisc
sit1: ipv6/ip remote 172.16.1.1 local any ttl 64
```

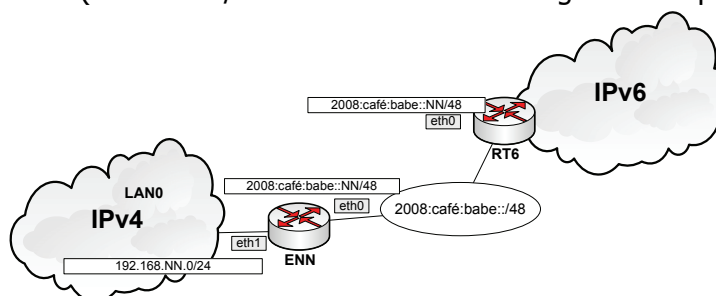
Podేశavanje default IPv6 rute preko kreiranog tunela odrađuje se komandom

```
root@RT ~# ip -6 route add ::/0 dev sit1
```

Na taj naćin se sav odlazni saobraćaj sa rutera koji konfiguriramo usmerava na tunnel i dalje na IPv4 mrežu koja se nalazi iza hosta 172.16.1.1

## Korak 8.

Slično, postoji mehanizam (IPv4-in-IPv6) za tuneliranje nekoliko razlićitih tipova paketa (npr. IPv4) preko IPv6 mreža (RFC 2473 / Generic Packet Tunneling in IPv6 Specification).



```
root@RT ~# ip -6 tunnel add ip6tnl1 mode ip4ip6 remote 2008:café:babe:2 local
2008:café:babe::NN
root@RT ~# ip link set dev ip6tnl1 up
root@RT ~# ip -6 route add ::/0 dev ip6tnl1 metric 1
```

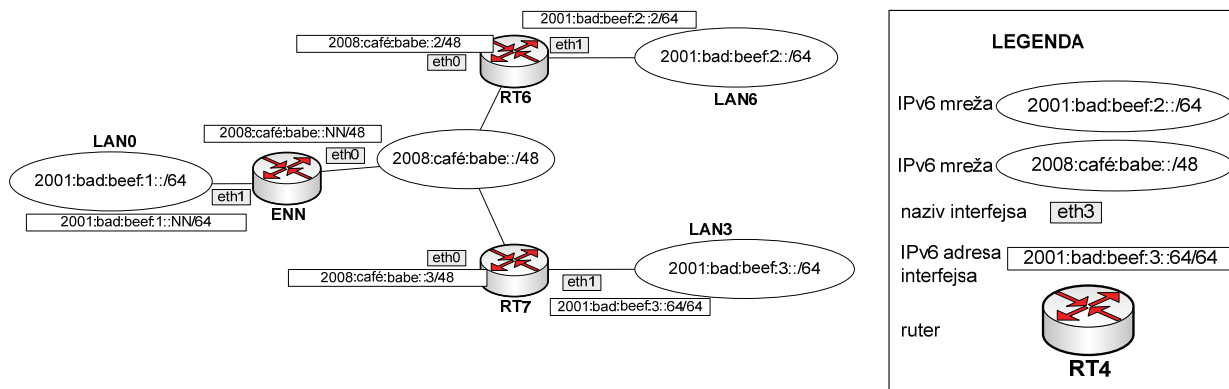
**NAPOMENA:** Ovaj naćin nije moguće primeniti u trenutnom scenariju mreže

## Zadatak 1.

Postaviti rutu na ruteru ENN za mrežu 2001:bad:beef:2::/64 koja je prikazana na slici 2.

Rutu je potrebno:

- postaviti unosom komandi na ruteru ENN
- predstaviti u pismenom obliku na papiru



Slika 2.

## Zadatak 2.

Uneti sve potrebne rute na ruteru RT6 (slika 2.).

Rute je potrebno predstaviti u pismenom obliku na papiru

## Završna razmatranja

U vežbi je demonstrirano kako se vrši konfiguracija rutera dodelom IPv6 adresa i postavljanjem statičkih ruta.

## BGP (Border Gateway Protocol)

### Cilj vežbe

The Border Gateway Protocol (BGP) je jedan od najvažnijih routing protokola Interneta. Radi tako što održava tabelu IP mreža ili prefiksa koji označavaju dostupnost između autonomnih sistema. BGP ne koristi uobičajenu IGP metriku.

Od 1994., četiri verzije ovog protokola su definisane. Osnovna verzija je verzija 4 koja ima podršku za besklasno rutiranje (*Classless Inter-Domain Routing*) i koristi agregaciju ruta (*route aggregation*) za povećanje veličine routing tabela. Od januara 2006, verzija četiri je definisana dokumentom RFC 4271, tako da je starija verzija dokumenta RFC 1771 koja definiše ovaj protokol napuštena.

BGP je dizajniran za EGP (Exterior Gateway Protocol) rutiranje, tj rutiranje između autonomnih sistema.

### Literatura

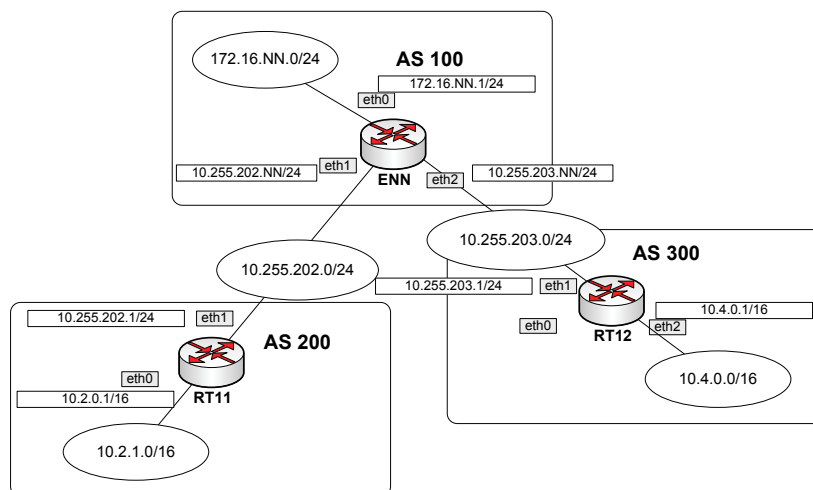
[1] James F. Kurose, Keith W. Ross, Umrežavanje računara, Pearson Education Inc, CET Beograd (prevod), ISBN 86-7991-267-0, Beograd, 2005.

[2] Brent D. Stewart, Clare Gough, CCNP BSCI – zvanični udžbenik za polaganje ispita, Cisco Press, Kompjuterska Biblioteka Čačak, ISBN 1-58720-147-9, Beograd 2008.

### Zadatak

Na slici je prikazana topologija mreže sa 3 rutera i 7 mreža koje su međusobno povezane preko njih. U vežbi je potrebno izvršiti konfigurisanje rutera za protokol BGP, da bi se tabele rutiranja dinamički ažurirale.

Na ruterima, koji su Linux računari sa Trustix Secure Linux 2.2 operativnim sistemom, dodatno je instaliran software za rutiranje - Zebra Free Routing Software. Konfigurisanje Zebra software vrši se komandama koje su slične komandama Cisco rutera, tako da se pomoću ovog softvera može naučiti i rad sa Cisco operativnim sistemom – Cisco IOS, a Cisco predstavlja najrasprostranjeniju rutersku platformu na svetu.

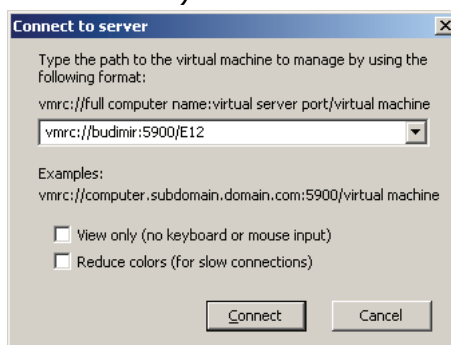


Slika 1. Topologija mreže

U vežbi je potrebno izvršiti konfiguraciju rutera ENN

## Logovanje na ruter

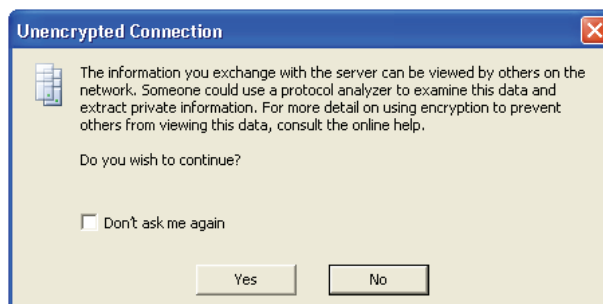
Logovanje na ruter koji je potrebno konfigurisati se vrši programom vmrc.exe koji se nalazi na studentskom disku (Z disk - folder KRM).



Kada se program startuje potrebno je upisati `vmrc://budimir:5900/ENN`

*NAPOMENA: Svako se loguje na drugi ruter, npr. E11, E12, E13 ... E20 itd.*

U sledećem koraku bi trebalo da se otvore dijalog prozori koji upozoravaju na sigurnost. Na ovakve i slične dijalog prozore, ukoliko se otvore, odgovoriti sa OK. Može se čekirati box "Don't ask me again" ukoliko ne želite da se ti prozori ponovo otvore pri sledećem pokušaju pristupa.



Kada se otvori prozor za logovanje potrebno je uneti

username: itguru

password: akademik



Kada se pojavi terminal rutera potrebno je prijaviti se na sistem. Prijava na sistem vrši se tako što se na mestu predviđenom za upis korisničkog imena (npr. **RT login:** ) upisuje korisničko ime **root** (administratorski nalog na Unix/Linux sistemima ) i pritiska taster **[Enter]**. Zatim se na mestu predviđenom za upis lozinke (**Password:** ) unosi lozinka **123456** i pritiska taster **[ Enter ]**. Unešena lozinka se ne vidi na ekranu radi sigurnosti.

## Konfigurisanje rutera

### Korak 1.

Pre početka rada potrebno je proveriti da li su dodeljene odgovarajuće IP adrese interfejsima na rutera. Provera IP adresa vrši se komandom **ifconfig | more**. Komanda se izvršava do kraja tasterom **[ Space ]**, a prekida tasterom **Q**.

Ako adrese nisu ispravno dodeljene mogu se promeniti komandom **netconfig**, npr.

```
root@RT1 ~# netconfig --ip=10.NN.0.1 --netmask=255.255.0.0 -d eth0
root@RT1 ~# netconfig --ip=10.255.202.NN --netmask=255.255.255.0 -d eth1
root@RT1 ~# netconfig --ip=10.255.203.NN --netmask=255.255.255.0 -d eth2
```

**NAPOMENA:** na rutera E11 IP adrese su 10.11.0.1, 10.255.202.11 I 10.255.203.11, an rutera E30 adrese su 10.30.0.1, 10.255.202.30 i 10.255.203.30. itd.

Primena unetih parametara i dodela IP adresa interfejsima se inicira komandom **service network restart** koja prvo obori interfejse, pa ih kasnije podigne sa novim parametrima.

```
root@RT1 ~# service network restart
Shutting down interface eth0:      [ OK ]
Shutting down interface eth1:      [ OK ]
Shutting down loopback interface:  [ OK ]
Setting network parameters:        [ OK ]
Bringing up loopback interface:     [ OK ]
Bringing up interface eth0:         [ OK ]
Bringing up interface eth1:         [ OK ]
```



```
E11 on budimir - Virtual Machine Remote Control
Remote Control
root@RT ~# netconfig --ip=172.16.1.111 --netmask=255.255.255.0 -d eth0
root@RT ~# netconfig --ip=172.16.2.111 --netmask=255.255.255.0 -d eth1
root@RT ~# service network restart
Shutting down interface eth1:           [ OK ]
Shutting down loopback interface:       [ OK ]
Disabling IPv4 packet forwarding:      [ OK ]
Setting network parameters:            [ OK ]
Bringing up loopback interface:         [ OK ]
Bringing up interface eth0:            [ OK ]
Bringing up interface eth1:            [ OK ]
root@RT ~#
```

## Korak 2.

Startovanje ruting softvera.

Na ruteru ENN potrebno je startovati softver za rutiranje, tj. programe zebra i ripd. To se vrši komandama.

```
root@RT ~# zebra -d
```

```
root@RT ~# bgpd -d
```

## Korak 3.

Konfigurisanje rutera ENN.

Pristup ruting softveru se vrši komandom `telnet localhost 2605` ili `telnet localhost bgpd`. Password je `zebra`.

```
root@RT# telnet localhost 2605
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.

Hello, this is zebra (version 0.94).
Copyright 1996-2002 Kunihiro Ishiguro.
```

```
User Access Verification
```

```
Password:
```

Konfigurisanje se vrši sledećim komandama, prilikom konfigurisanja potrebno je uneti sve mreže koje ruter treba da oglašava tj. sve mreže koje su povezane direktno na ruter (10.NN.0.0/16, 10.255.202.0/24 i 10.255.203.0/24).

```
bgpd > enable # ulazak u komandni mod
bgpd # configure terminal # komanda za ulazak u mod za konfigurisanje
rutera # konfigurisanje
bgpd(config)# router bgp 100 # konfiguracija bgp ruting procesa
```

Ukoliko se pojavi poruka BGP is already running: AS is 7675, to znači da postoji već autonomni sistem sa tim brojem pa je potrebno promeniti id autonomnog sistema (AS)

```
bgpd(config)# no router bgp 7675      # konfiguracija bgp ruting procesa
bgpd(config)# router bgp 100          # konfiguracija bgp ruting procesa

bgpd(config-router)# bgp router-id 10.NN.0.1 # konfiguracija ruter ID-a u formatu
IP adrese. Potrebno je uneti IP adresu eth0 interfejsa

bgpd(config-router)# network 10.NN.0.0/16 # ovde se unose mreže koje ruter oglašava,
tj. sve mreže koje su direktno povezane na ruter. U ovoj vežbi je potrebno uneti
mrežu 10.NN.0.0/16 i mreže 10.255.202.0/24 i 10.255.203.0/24.

bgpd(config-router)# network 10.255.202.0/24
bgpd(config-router)# network 10.255.203.0/24

bgpd(config-router)# neighbor 10.255.202.1 remote-as 200      # identifikacija
suseda (rutera) i njegovog autonomnog sistema
bgpd(config-router)# neighbor 10.255.203.1 remote-as 300      # snimanje konfiguracije

bgpd(config-router)# write      # snimanje konfiguracije
Configuration saved to /usr/local/etc/bgpd.conf

bgpd(config-router)# exit      # izlazak iz zebre
bgpd (config)# exit            # izlazak iz zebre
```

Konfiguracija se može pogledati komandom **show running-config** (dok se nalazite u modu za konfigurisanje - enable) u ruting softveru, tj. pre izlaska iz zebra softvera.

Izlazak is zebra softvera se vrši komandom

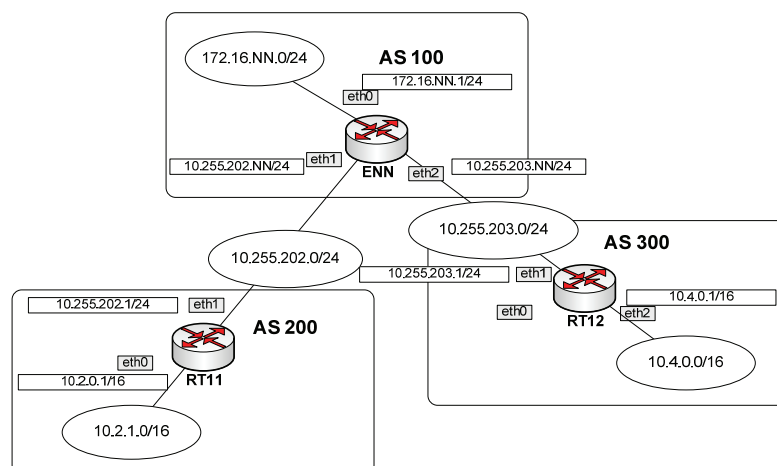
```
bgpd# exit
Connection closed by foreign host.
root@RT #
```

Proveriti tabelu rutiranja komandom route -n

## Analiza BGP podataka – raditi posle završenog zadatka

```
ospfd> show ip bgp
ospfd> show ip bgp neighbors
```

### Zadatak 1.



Odgovoriti kojim komandama zebra rutera bi se trebao konfigurirati ruter **RT11** na slici da bi mogao da funkcioniše ispravno.

Da bi se ruter ispravno konfigurirao potrebno je:

- Definirati AS i router-ID
- navesti sve mreže koje se oglašavaju, tj. one koje su direktno povezane na ruter
- definirati susede

## **Zadatak 2.**

Odgovoriti kojim komandama zebra rutera bi se trebao konfigurirati ruter **RT12** na slici da bi mogao da funkcioniše ispravno.

Da bi se ruter ispravno konfigurirao potrebno je:

- Definirati AS i router-ID
- navesti sve mreže koje se oglašavaju, tj. one koje su direktno povezane na ruter
- definirati susede

## Zaštita računarskih mreža - Firewall sistemi

### Zaštita računarskih mreža - Firewall sistemi

U današnje vreme, sigurnost mreža i zaštita podataka predstavljaju veoma važan faktor. Povezivanje mreže na Internet i direktna veza sa svetom, predstavljaju stalnu pretnju za sistem i pruža mogućnost napadačima da, koristeći različite bezbednosne propuste, neovlašćeno pristupe sistemu.

Firewall predstavlja mehanizam zaštite u računarskim mrežama. To je softverski ili hardverski proizvod koji proverava pakete koji dolaze do njega i na osnovu unetih pravila, propušta ili odbija te pakete. Firewall se obično nalazi na ulazu u mrežu, tj. između unutrašnje i spoljašnje mreže, tako da se celokupan saobraćaj mora odvijati preko njega. On štiti mrežu na svim softverskim slojevima OSI modela – od sloja veze podataka, do aplikacionog sloja.

U firewall sistemima se koriste tri osnovna metoda zaštite:

- filtriranje paketa (packet filtering) – metod pomoću koga se odbija pristup TCP/IP paketa koji dolaze sa neovlašćenih hostova i na neovlašćene servise.
- translacija adresa ili NAT (Network Address Translation) – metod koji prevodi IP adrese unutrašnjih hostova u neku od spoljašnjih (javnih) adresa i tako omogućava hostovima iz unutrašnje mreže pristup Internetu. Za ovaj metod se često koristi i termin IP
- masquerading.
- proxy servisi – metod koji radi pomoću aplikacija višeg sloja, koje omogućavaju hostovima iz interne mreže pristup servisima na javnim serverima.

### IPTABLES

Na Linux sistemima IPTABLES predstavlja popularnu firewall platformu. IPTABLES koristi pravila za filtriranje konekcija i paketa i grupiše ih u lanac (eng. chain). Postoje nekoliko različitih lanaca, npr. INPUT lanac se koristi za pakete koji dolaze do hosta, OUTPUT lanac za pakete koji odlaze sa hosta, a koji su nastali na samom hostu i FORWARD lanac koji se koristi za pakete koji prolaze kroz host. Svaka linija koja se unese u lanac je pravilo. To pravilo koristi kernel sistema da bi znao šta treba da radi sa odgovarajućim paketom. Pravila se pišu na sledeći način:

```
iptables [-t table] command [match] [target/jump]
```

- **t [table]** se koristi kada se želi koristiti određena tabela. Ako se ova opcija izostavi podrazumevana tabela je FILTER tabela. Postoje tri tabele: NAT, MANGLE i FILTER. Filter tabela se koristi za filtriranje paketa i ujedno je najvažnija tabela. NAT tabela se koristi za adresnu translaciju.

- **command** se koristi da bi se naznačilo da li pravilo treba da se doda ili obriše.

- A dodaje pravilo (append)
- D briše pravilo (delete)
- I ubacuje pravilo na početku lanca (insert)
- L lista pravila i
- F briše pravila.

- **match** govori kernelu koji se paketi proveravaju. To se radi upisom IP adrese sa koje dolazi paket ili interfejsa itd.

-p, -protocol iptables -A INPUT -p tcp

komanda koristi za proveru određenog protokola TCP, UDP ili ICMP.

-s, -src, -source iptables -A INPUT -s 192.168.1.1

komanda se koristi za proveru paketa koji dolaze sa navedenom izvorišnom IP adresom. Može se navesti samo IP adresa kao u primeru bez ili sa mrežnom maskom (192.168.0.0/255.255.255.0) ili sa mrežnim prefiksom (192.168.0.0/24). Ako se

-d, -dst, -destination iptables -A INPUT -d 192.168.1.1

komanda se koristi za proveru paketa koji dolaze sa navedenom odredišnom IP adresom. Mrežna maska ili mrežni prefiks se takođe mogu navesti kao u prethodnom slučaju. Ako se navede -destination ! 192.168.0.1 proveravaju se samo paketi koji ne odlaze na navedeno odredište.

-i, -in-interface iptables -A INPUT -i eth0

komanda se koristi za proveru paketa koji dolaze sa naznačenog interfejsa. Upotreba komande je dozvoljena u INPUT, FORWARD i PREROUTING lancima.

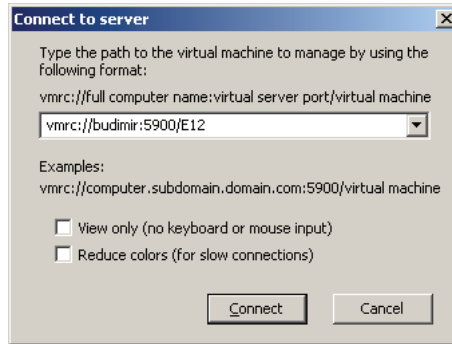
-o, -out-interface iptables -A FORWARD -o eth0

komadna se koristi za proveru paketa koji odlaze preko određenog interfejsa. Koristi se samo sa OUTPUT, FORWARD i POSTROUTING lancima. Može se koristiti i znak za inverziju - !, kao u slučaju -in-interface komande.

- **[target/jump]** ove opcije govore koja akcija treba da se sprovede nad paketom koji odgovara pravilu, tj. da li se paket odbacuje, blokira ili propušta. Dve osnovne TARGET opcije su ACCEPT, REJECT i DROP.

## Logovanje na ruter

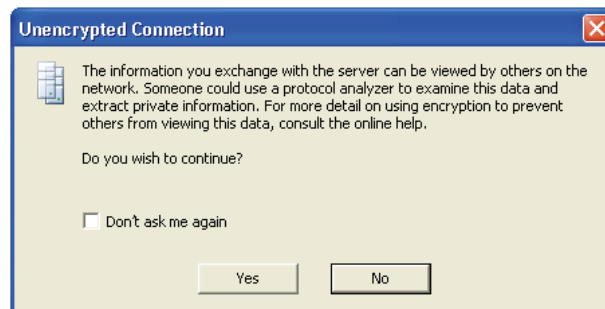
Logovanje na ruter koji je potrebno konfigurisati se vrši programom vmrc.exe koji se nalazi na studentskom disku (Z disk - folder KRM).



Kada se program startuje potrebno je upisati `vmrc://budimir:5900/ENN`

*NAPOMENA: Svako se loguje na drugi ruter, npr. E11, E12, E13 ... E20 itd.*

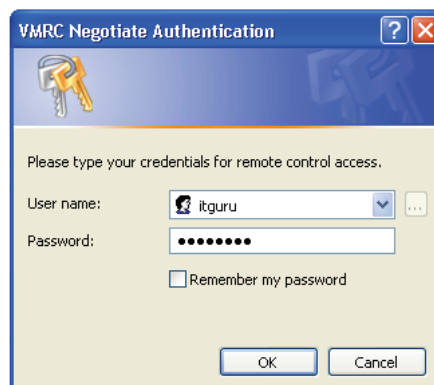
U sledećem koraku bi trebalo da se otvore dijalog prozori koji upozoravaju na sigurnost. Na ovakve i slične dijalog prozore, ukoliko se otvore, odgovoriti sa OK. Može se čekirati box "Don't ask me again" ukoliko ne želite da se prozori ponovo otvore pri sledećem pokušaju pristupa.



Kada se otvori prozor za logovanje potrebno je uneti

username: itguru

password: akademik



Kada se pojavi terminal rutera potrebno je prijaviti se na sistem. Prijava na sistem vrši se tako što se na mestu predviđenom za upis korisničkog imena (npr. **RT login:** ) upisuje korisničko ime **root** (administratorski nalog na Unix/Linux sistemima) i pritiska taster **[Enter]**. Zatim se na mestu predviđenom za upis lozinke (**Password:** ) unosi lozinka **123456** i pritiska taster **[ Enter ]**. Unešena lozinka se ne vidi na ekranu radi sigurnosti.

## Konfigurisanje rutera

### Korak 1.

Pre početka rada potrebno je proveriti da li su dodeljene odgovarajuće IP adrese interfejsima na ruteru. Provera IP adresa vrši se komandom `ifconfig | more`. Komanda se izvršava do kraja tasterom [ `Space` ], a prekida tasterom `Q`.

IP adrese se dodeljuju interfejsima komandom `netconfig`, npr

```
root@RT1 ~# netconfig --ip=172.16.1.NN --netmask=255.255.255.0 -d eth0
root@RT1 ~# netconfig --ip=192.168.NN.1 --netmask=255.255.255.0 -d eth1
```

*NAPOMENA: na ruteru E11 IP adrese su 172.16.1.11 i 172.16.2.11, na ruteru E12 IP adrese su 172.16.1.12 i 172.16.2.12, tj. 172.16.1.NN i 172.16.1.NN za svaki ENN.*

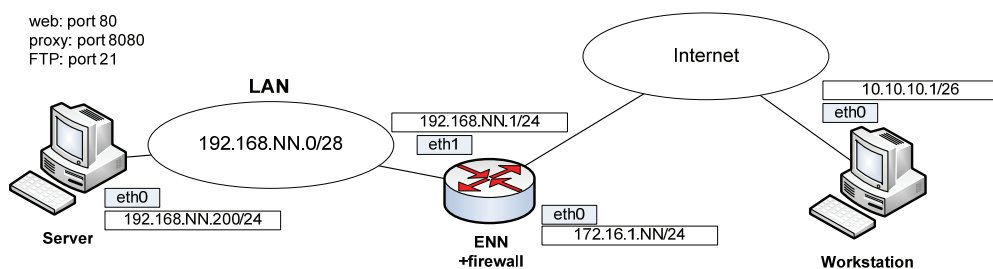
Primena unetih parametara i dodela IP adresa interfejsima se inicira komandom `service network restart` koja prvo obori interfejse, pa ih kasnije podigne sa novim parametrima.

```
root@RT1 ~# service network restart
Shutting down interface eth0:      [ OK ]
Shutting down interface eth1:      [ OK ]
Shutting down loopback interface:  [ OK ]
Setting network parameters:        [ OK ]
Bringing up loopback interface:    [ OK ]
Bringing up interface eth0:        [ OK ]
Bringing up interface eth1:        [ OK ]
```

### Korak 2.

#### Konfiguracija firewall-a

U mreži prikazanoj na slici ruter ENN štiti mrežu 192.168.NN.0/24 i server 192.168.NN.200 od neovlašćenog pristupa sa Interneta.



Sadržaj IPTABLES pravila možemo videti komandom:

```
iptables -L -n
```

Na početku, da bi smo izbegli pojavu grešaka, potrebno je obrisati sva pravila. To se radi komandom:

```
iptables -F
```

Da bi se **dozvolio pristup sa radne stanice (workstation)** sa adresom 10.10.10.1 na FTP server na računaru 192.168.NN.200 (FTP server je na portu 21) potrebno je otkucati komandu

```
iptables -A FORWARD -p tcp -s 10.10.10.1/32 -d 192.168.NN.200/32 --dport 21 -j ACCEPT
```

Budući da se pravilo odnosi na pakete koji dolaze spolja sa Interneta, prolaze kroz firewall i odlaze u mrežu 192.168.NN.0/24 koristi se FORWARD lanac.

Da bi se **dozvolio pristup sa radne stanice (workstation)** sa adresom 10.10.10.1 na Proxy serveru koji se nalazi na računaru 192.168.NN.200 (Proxy server je na portu 8080), ali i celoj mreži 192.168.NN.0/24 potrebno je otkucati komandu:

```
iptables -A FORWARD -p tcp -s 10.10.10.1/32 -d 192.168.NN.0/24 --dport 8080 -j ACCEPT
```

Da bi se **zabranio pristup svim ostalim računarima sa Interneta serveru na računaru 192.168.NN.200**, potrebno je otkucati komandu

```
iptables -A FORWARD -s 0.0.0.0/0 -d 192.168.NN.200/32 -j REJECT
```

U ovom slučaju nije potrebno uneti komandu `-p tcp`, jer je potrebno zabraniti saobraćaj za sve protokole (tcp, udp i icmp, a ne samo tcp).

Takođe, u ovom slučaju nije potrebno uneti odredišni port, npr. `--dport 8080` budući da je potrebno zabraniti ulaz na sve portove, a ne samo na 8080.

Adresa **0.0.0.0/0** menja sve računare na Internetu, tj. sve moguće adrese. Ako se source (-s) ili destination (-d) adresa izostavi, npr. `iptables -A FORWARD -d 192.168.NN.200/32 -j REJECT`, izostavljene adrese će se zameniti sa 0.0.0.0/0.

Kada bi trebalo da se zabrani pristup ENN + firewallu sa svih ostalih računara, tada bi se koristio lanac INPUT, pa je potrebno otkucati komandu

```
iptables -A INPUT -s 0.0.0.0/0 -d 172.16.1.NN/32 -j REJECT
```

Lista unetih pravila se dobija komandom:

```
iptables -L -n
```

### **Korak 3.**

#### **Brisanje komandi**

Ako treba da se ukine pristup proxy serveru spolja, potrebno je obrisati uneto pravilo. Brisanje se vrši komandom

```
iptables -D FORWARD -p tcp -s 10.10.10.1/32 -d 192.168.NN.0/24 --dport 8080 -j ACCEPT
```

Proveriti da li je pravilo obrisano komandom

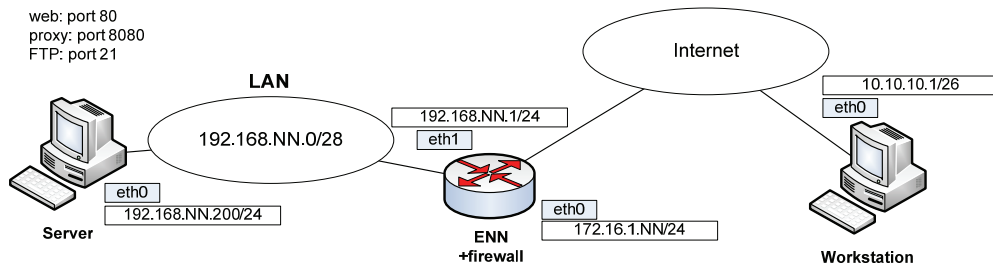
```
iptables -L -n
```



## Korak 4.

### Adresna translacija

Za mrežnu konfiguraciju prikazanu na slici, potrebno je izvršiti konfiguraciju firewall-a tako da se omogući adresna translacija, tj. pozajmljivanje ili dodela "javne IP" adrese rutera (172.16.1.NN) hostovima iz unutrašnje mreže (192.168.NN.0/24). Na ovaj način će se omogućiti svim hostovima da pristupaju Internetu i pored toga što nemaju javnu IP adresu i sami ne mogu da komuniciraju sa ostalim hostovima na Internetu.



Napomena: U ovom slučaju potrebno je koristiti nat tabelu (-t nat) i POSTROUTING lanac i izlazni interfejs eth0 (-eth0)

```
iptables -A POSTROUTING -t nat -s 192.168.NN.0/24 -o eth0 -j SNAT --to-source 172.16.1.NN
```

## Korak 5.

### Redirekcija na unutrašnju mrežu celokupnog saobraćaja

Na istoj slici je predstavljena mrežna konfiguracija u kojoj je web server postavljen u unutrašnju LAN mrežu. Web serveru je dodeljena privatna IP adresa. Ovo je učinjeno radi dodatne sigurnosti servera. Problem je što spoljni računari ne mogu da pristupe web serveru zbog privatne IP adrese.

Da bi se omogućio pristup spoljnim računarima potrebno je izvršiti redirekciju web zahteva, koji dolaze sa spoljašnje mreže na ruter sa „javnom“ IP adresom (172.16.1.NN), na privatnu IP adresu web servera (192.168.NN.200).

Napomena: U ovom slučaju potrebno je koristiti nat tabelu (-t nat) i PREROUTING lanac

### OTKUCATI JEDNU OD SLEDEĆE DVE KOMANDE

Celokupan saobraćaj se preusmerava komandom

```
iptables -A PREROUTING -t nat -d 172.16.1.NN -j DNAT --to-destination 192.168.NN.200
```

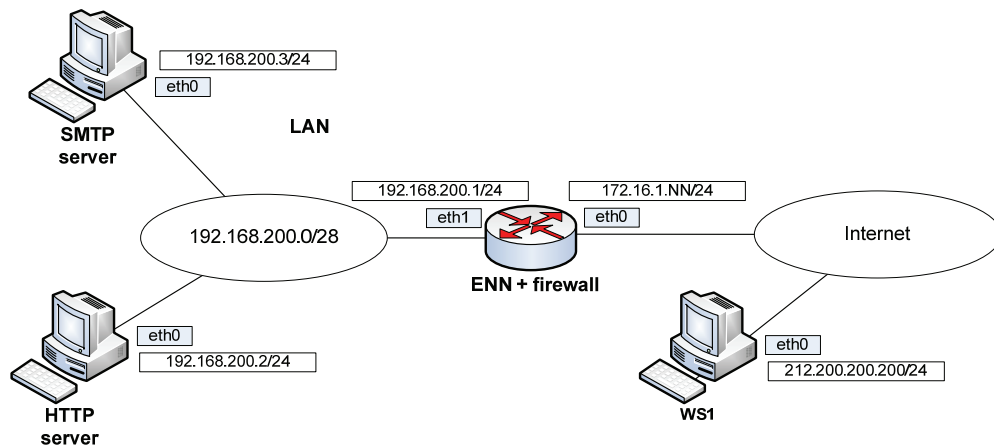
a samo web saobraćaj se presumeava komandom

```
iptables -A PREROUTING -p tcp -t nat -d 172.16.1.NN -dport 80 -j DNAT --to-destination 192.168.NN.200
```

Da bi se izlistao sadržaj NAT tabele otkucati komandu

```
iptables -L -t nat -n
```

### Zadatak 1.



Za mrežnu konfiguraciju prikazanu na slici izvršiti konfiguraciju firewall-a tako da se:

- dozvoli pristup sa računara 212.200.200.200 na SMTP server 192.168.200.3. SMTP ili mail protokol radi na portu 25.
- zabrani pristup za sve računare sa Interneta na mrežu 192.168.200.0/24
- zabrani pristup za sve računare sa Interneta na ENN ruter + firewall
- izvrši preusmeravanje SMTP zahteva, koji dolaze sa spoljašnje mreže na ruter sa javnom IP adresom (212.200.200.1), na privatnu IP adresu SMTP servera (192.168.200.3). SMTP je na portu 25.
- izvrši preusmeravanje HTTP zahteva, koji dolaze sa spoljašnje mreže na ruter sa javnom IP adresom (212.200.200.1), na privatnu IP adresu SMTP servera (192.168.200.2). HTTP je na portu 80.
- omogući izlazak na Internet mrežu hostovima 192.168.200.2 i 192.168.200.3 (tj. za mrežu 192.168.200.0/24) pomoću NAT servera