



UNIVERZITET SINGIDUNUM

Departman za posleddiplomske studije

-MASTER STUDIJSKI PROGRAM-

SAVREMENE INFORMACIONE TEHNOLOGIJE

Dunja Adžić

BEZBEDNOST RAČUNARSKE MREŽE

- Master rad -



Beograd, 2010.



UNIVERZITET SINGIDUNUM

Departman za posleddiplomske studije

-MASTER STUDIJSKI PROGRAM-

SAVREMENE INFORMACIONE TEHNOLOGIJE

Dunja Adžić

BEZBEDNOST RAČUNARSKE MREŽE

- Master rad -

Mentor:

Prof. dr Mladen Veinović

Kandidat:

Adžić Dunja 410267/2009



Beograd, 2010.

Sadržaj

1. Uvod	7
2. Povezivanje udaljenih lokacija	9
2.1. Adresiranje	11
2.2. Tehnologija LAN povezivanja	12
3. Rutiranje, Napadi na OSPF i protivmere	15
3.1. Link – state advertisement (LSA)	17
3.1.1. Kriptografsko rešenje za Link State	19
3.1.2. Specijalni tretman LS-a starosnog polja	19
3.1.3. Prednosti i mane	20
3.2. Hash Chains za Stable Link State (SLS)	20
3.2.1. Prednosti i mane	24
3.3. Hash Chains za Fluctuating Link State	24
3.3.1. Prednosti i mane	29
4. Context-Based Access Control (CBAC) - Cisco IOS firewall set funkcija	29
4.1. Glavne funkcije CBAC-a	30
4.2. Proces CBAC-a	31
4.3. Podržani protokoli	32
4.4. Prednosti i ograničenja CBAC-a	34
1. CBAC se oslanja na efektivnost konfiguracije access-lista na ruteru i	35
2. CBAC ne može pregledati saobraćaj sa izvorne ili odredišne adrese rutera	35
To znači da se ne može osloniti na CBAC da obezbedi bilo kakvu sigurnost za sam ruter.	35
5. Bezbednost kontrole pristupa	36
5.1. RADIUS (Remote Authentication Dial In User Service)	36
5.2. Protokol RADIUS	38
5.2. Problemi sa protokolom RADIUS	39
6. IPSec	40
6.1. IPSec protokoli	45
6.1.1. Protokol AH	45
6.1.2. Protokol ESP	47
6.2. Režimi rada	48
6.2.1. Transportni režim rada	48

6.2.2. Tunelovanje.....	51
6.3. Uspostavljanje IPSec komunikacije	53
6.3.1. IKE.....	54
6.4. Resursi koje IPSec zahteva i problemi u implementaciji	55
6.4.1. Konfiguracija NAT-a	57
7. Secure Shell (SSH).....	58
8. L2 security i konfiguracija ostalih protokola.....	60
8.1. Storm Control.....	60
8.2. DHCP snooping.....	61
8.3. RSPAN	61
8.4. Menadžment.....	62
8.5. Kvalitet saobraćaja	63
9. IP telefonija.....	65
10. Konfiguracije rutera i switcheva	68
Cisco Ruter u Novom Sadu	73
Cisco Ruter u Nišu	77
Beograd Switch 1	80
Beograd Switch 2	83
Novi Sad Switch 1.....	87
Novi Sad Switch 2.....	90
Niš Switch 1.....	94
Niš Switch 2.....	97
11. Zaključak.....	101
Literatura	103

Slike

Slika.2.1. Šema WAN povezivanja.....	10
Slika.2.2. Šema LAN povezivanja	13
Slika.3.1. OSPF LSA zaglavlja.....	18
Slika.3.2. Operacija SLS-a	23
Slika.3.3. Tabela hash chain-a.....	25
Slika.3.4. FLS operacije	28
Slika.5.1. RADIUS	37
Slika.6.1. Mesto protokola IPSec u skupu protokola	43

Slika.6.2. Standardni oblik IP paketa.....	45
Slika.6.3. AH zaglavlje.....	45
Slika.6.4. ESP paket	47
Slika.6.5. AH u transportnom režimu rada	49
Slika.6.6. ESP u transportnom režimu rada	49
Slika.6.7. ESP + AH u transportnom režimu rada.....	50
Slika.6.8. Tunelovanje	51
Slika.6.9. AH tunelovanje	52
Slika.6.10. ESP tunelovanje	53
Slika.9.1. Multisite WAN with Centralized Call-Processing Deployment	66

Sažetak:

Master rad na temu bezbednosti računarske mreže pisan je sa idejom da prikaže visoku zaštitu mrežne opreme i komunikacije na svim nivoima implementacijom bezbednosnih protokola i mehanizama. Korišćenjem najnovije Cisco mrežne opreme, zbog svoje pouzdanosti i performansi uređaja, omogućava se visok stepen zaštite mreže bez smanjenja brzine protoka podataka u mreži.

Ključne reči: Bezbednost mreže, Cisco, protokol.

Abstract:

This Master's thesis on computer network security is written with the idea to show the high protection of network equipment and communication at all levels of implementation of security protocols and mechanisms. Using the latest Cisco network equipment, for its reliability and performance of the devices allows that a high level of network security does not affect the flow rate of traffic in the network.

Key words: Network security, Cisco, protocol.

1. Uvod

U ovom master radu izrađen je projekat zaštite podataka celokupne mreže zamišljenog preduzeća DunjaInc Beograd d.o.o. U daljem tekstu koristi se reč projekat, zato što bolje opisuje kompletan sadržaj rada.

Preduzeće DunjaInc Beograd d.o.o. iz Beograda pruža usluge konsaltinga, projektovanja, implementacije i tehničke podrške u oblasti informacionih i komunikacionih tehnologija i proizvoda.

U datom projektu potrebno je povezati tri lokacije (Beograd, Novi Sad, Niš), omogućiti pristup Internetu svim uređajima u mreži, zaštititi mrežnu opremu i komunikaciju, instalirati mrežne servise i administrirati ih, kao i obezbediti VOIP komunikaciju. Na svakoj od lokacija, biće 50 internih korisnika mreže, potrebni serveri za mrežne servise, uređaji koji omogućavaju rad IP telefonije (VOIP).

U ovom projektu odlučeno je da proizvođač mrežne opreme bude Cisco, zbog svoje opšte poznate pouzdanosti i performansi uređaja. Konektivnost između lokacija omogućena je iznajmljivanjem L3 MPLS VPN usluga od internet servis provajdera, u ovom slučaju Telekom Srbija. Saobraćaj između lokacija će biti zaštićen IPSec-om, koji će se realizovati podizanjem virtuelnih tunel interfejsa. Cisco ruteri predviđeni ovim rešenjem, podržavaju firewall, IPSEC, i ostale bezbednosne protokole i mehanizme.

Rutiranje saobraćaja unutar mreže će biti realizovano podizanjem OSPF rutinog protokola. Predviđeno je da izlaz na internet bude obezbeđen instalacijom linka na glavnoj lokaciji (Beograd), a ovim dizajnom je omogućeno centralizovano rešenje, u kojem mrežni administrator (IT odeljenje) lako može pratiti i vršiti nadzor celokupnog izlaznog i ulaznog eksternog saobraćaja (saobraćaja ka i od interneta).

U okviru ovog rešenja biće podignuti svi potrebni mrežni servisi i protokoli neophodni za funkcionisanje mreže zadatih uslova. Kao što je već spomenuto, oprema će biti Cisco proizvođača. Planirana je instalacija rutera na svakoj lokaciji, koji će biti povezani međusobno L3 MPLS VPN rešenjem (povezani sa provajderom optičkim linkovima), a na svakoj lokaciji biće instalirana po dva L3 Cisco switcha.

U projektu koristimo sledeću opremu:

C2911-VSEC/K9 --- Cisco ruter C2911, serije 2900 Cisco rutera, baziranih na pređašnjem iskustvu Cisco-a kao proizvođača u pružanju integrisanih usluga. Cisco 2900 serija nudi hardversku enkripciju, voice i video digitalne signalne procesore (DSP moduli), opcioni firewall, sistem preventive napada (IPS – Intrusion Prevention System), procesiranje poziva, glasovni e-mail sistem (voicemail), aplikativne servise, itd... Ova serija, takodje podržava širok spektar opcionih modula i interfejsa za LAN, kao i za WAN/MAN arhitekture. Ono po čemu se proizvodi ove serije izdvajaju su integrisani servisi (voice, wireless, security, data center tehnologije, mobilnost uređaja, itd...), visoke performanse, visok stepen modularnosti i mrežne agilnosti, efikasno korišćenje električne energije, mogućnost redundantnog napajanja, podrška za PoE (Power Of Ethernet, prenos napajanja preko mrežne infrastrukture,

etherneta)...

Cisco C2911-VSEC/K9 ruter je izabran, pored gore navedenog, zbog svojih mogućnosti da podrži sve protokole i servise potrebne za dizajn i konfiguraciju ovog mrežnog rešenja. Cisco C2911 podržava tri integrisana 10/100/100 Ethernet porta (RJ-45 konektori), jedan servis modul slot, četiri WAN portova visoke brzine, dva DSP modula, jedan interni slot za aplikativne servise. Pruža punu podršku za PoE rešenje, integrisana bezbednosna rešenja (IPsec sa hardverskom enkrijpcijom , CBAC Firewall, IPS, ZBFW Firewall, Content Filtering rešenje) i kompletne servise i protokole potrebne za instalaciju IP telefonije.

Cisco 3560-48PS je izabrani model switcha u ovom projektu. Cisco 3560 serija switcheva je serija fixne konfiguracije, namenjena firmama koje imaju potrebu za inteligentnom mrežnom infrastrukturom, svoje mešto odlično pronalazi u pristupnom delu mreže. Ima 48 10/100/100 Ethernet portova sa potpunom podrškom za PoE (802.3af). Cisco 3560 ima podršku za L3 rutiranje saobraćaja, implementaciju mehanizama za uspostavljanje kvaliteta servisa, multicast menadžment kao i veliki broj bezbednosnih protokola i mehanizama. Kao ovakav Cisco 3560-48PS se idealno pokazao kao rešenje za pristupni deo mreže.

U glavi dva razmatra se povezivanje udaljenih lokacija Beograd, Niš i Novi sad, adresiranje unutrašnje mreže i tehnologija LAN povezivanja.

U glavi tri razmatra se rutiranje internog saobraćaja, koji je sproveden OSPF ruting protokolom kao i mane i prednosti kriptografskih rešenja za protokol.

U glavi četiri razmatra se Contex-Based Access Control-Cisco IOS firewall set funkcija, njihova implementacija, konfiguracija, prednosti i ograničenja.

U glavi pet razmatra se bezbednost kontrole pristupa, RADIUS server i protokol, njihova implementacija i problemi sa protokolom RADIUS.

U glavi šest razmatra se IPSec tehnologija, konfiguracija, režimi rada, uspostavljanje komunikacije i konfiguracija NAT-a.

U glavi sedam razmatra se zaštita udaljenog pristupa uređajima SSh protokolom, konfiguracija protokola.

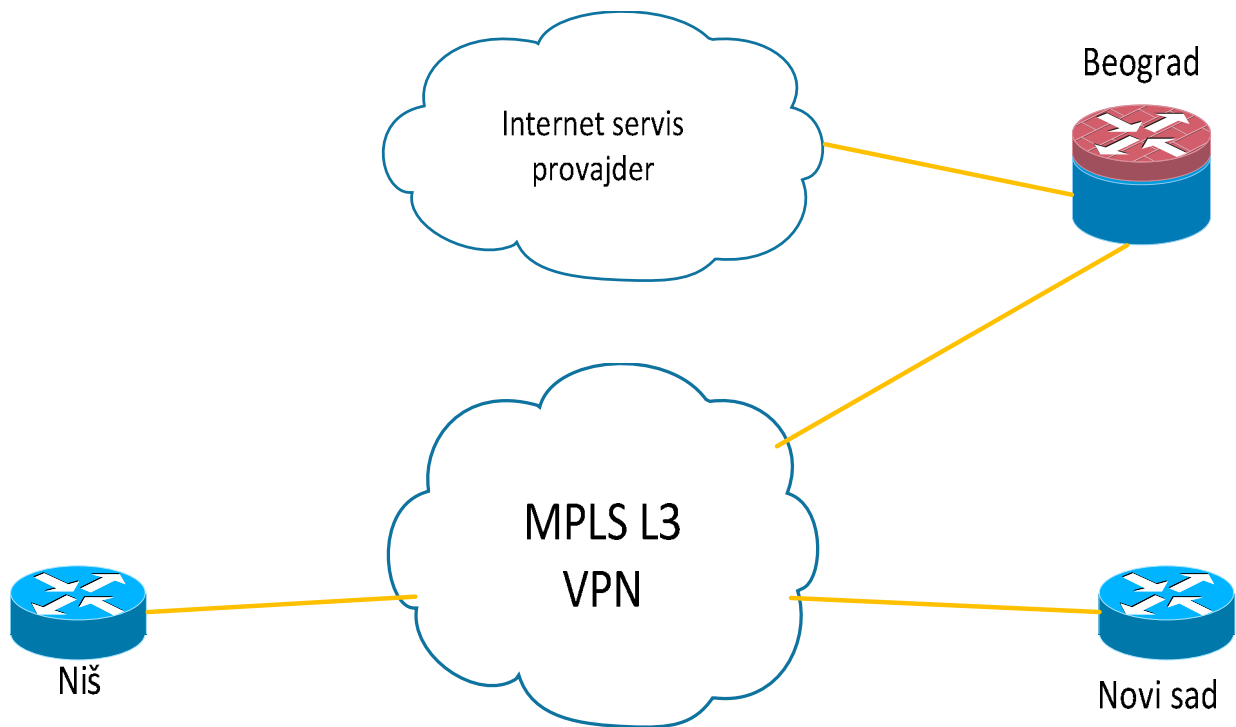
U glavi osam razmatra se L2 bezbednost, kao i konfiguracije ostalih protokola koji su konfiigurisani u projektu: Storm Control, DHCP snooping itd.

U glavi devet razmatra se implementacija i konfiguracija IP telefonije.

U glavi deset navedene su konfiguracije rutera i switcheva na udaljenim lokacijama (Beograd, Niš i Novi Sad).

2. Povezivanje udaljenih lokacija

Povezivanje između udaljenih lokacija (Beograd, Niš, Novi Sad), uspostavljena je uz pomoć L3 MPLS VPN-a koji će se iznajmiti od Telekoma Srbije, prikazano na slici 2.1. L3 MPLS VPN kao servis, predstavlja rešenje uz pomoć kojeg klijenti (u ovom slučaju udaljene lokacije) dobijaju direktan link prema internet servis provajderu, objavljuju svoje interne mreže (koje žele da objave ka ostalim lokacijama), zadržavaju privatne opsege i izbegavaju potrebu za NAT-om (Network Address Translation), a u konfiguraciji rutiranja učestvuju potpuno samostalno ili uz pomoć telekoma. MPLS kao tehnologija, pruža mogućnost internet servis provajderima da unificiranom mrežnom infrastrukturu nude broj servisa svojim klijentima i korisnicima. MPLS je potpuno transparentan za krajne korisnike (CE-Customer Equipment rutere).



SI.2.1. Šema WAN povezivanja

2.1. Adresiranje

Unutrašnja mreža će biti adresirana privatnim opsegom 10.0.0.0/8. Na svakoj lokaciji će biti 3 vlana (Virtual Local Area Network), i to jedan vlan namenjen za data saobraćaj unutar same lokacije, jedan za potrebe internet telefonije i jedan za menadžment saobraćaja (mrežni servisi podignuti na serverima).

Beograd

Vlan 11 – data vlan -10.1.0.0/24

Vlan 12 –voice vlan 10.1.1.0/24

Vlan 13 –menagment vlan 10.1.2.0/24

Novi Sad

Vlan 21 –data vlan – 10.2.0.0/24

Vlan 22 –voice vlan 10.2.1.0/24

Vlan 23 –menagment vlan 10.2.3.0/24

Niš

Vlan 31 –data vlan – 10.3.0.0/24

Vlan 32 –voice vlan – 10.3.1.0/24

Vlan 33 – menagment vlan 10.3.0.0/24

Obzirom da će između lokacija biti podignuti virtuelni tunel interfejsi, radi uspostavljanja IPsec konekcije, potrebno je i njih adresirati.

Beograd - Novi Sad – 10.12.0.0/24

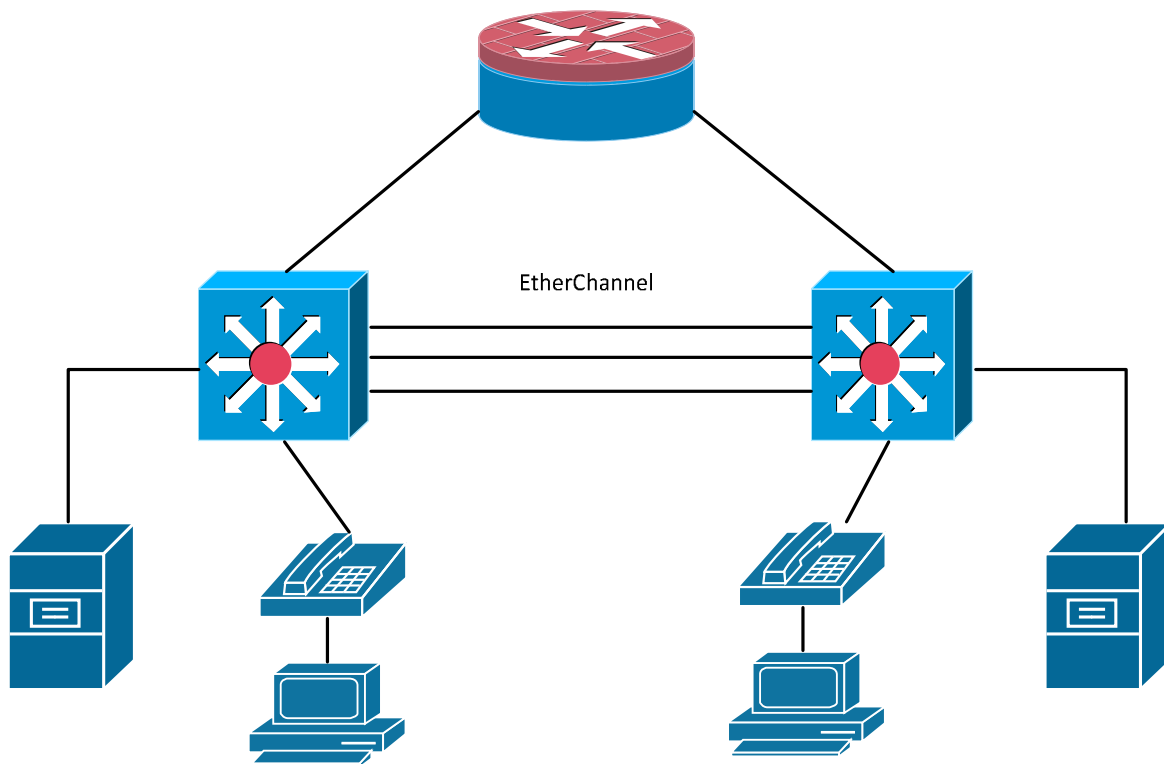
Beograd - Niš – 10.13.0.0/24

Novi Sad - Niš-- 10.23.0.0/24

2.2. Tehnologija LAN povezivanja

LAN svake lokacije će biti izdizajniran na sledeći način. Dva switcha će biti konektovani sa ruterom, i između sebe uz pomoć L2 EtherChannel tehnologije. EtherChannel tehnologija omogućava switch-evima da više fizičkih linkova spoje u jedan logički (u ovom slučaju će biti iskorišćen maximum od 8 ethernet linkova brzine 100 Mbit/s), čime se postiže ravnomerna raspodela saobraćaja i potpuno iskorišćenje protoka linkova. Krajnje stanice će 100 megabitnim ethernetom biti povezane na switch, i ti portovi funkcionišu u access modu. I pored naizgled jednostavne fizičke i logičke topologije, gde ne postoji mogućnost da se napravi petlja, spanning tree će biti implementiran. 802.1w ili Rapid Spanning Tree je protokol koji omogućava loop-free mrežnu topologiju tako što jedan uređaj u mreži postavi kao centralnu tačku saobraćaja, i topologiju bez redundantnih putanja čime se dobija logička mrežna topologija gde je mogućnost petlje svedena na nulu. U slučaju promene, 802.1w konvergira i veoma brzo reaguje na promene. Za potrebe inter-vlan rutiranja (preusmeravanje saobraćaja između lokalnih vlan-ova) biće implementirani virtualni SVI interfejsi (Switch Virtual Interface). Svi interfejsi su logički interfejsi koji se implementiraju u okviru VLAN-ova i koji su zaduženi za inter-vlan komunikaciju.

Na jednom od switcheva u svakoj od LAN mreža, biće konfigurisani VTP Server mod. VTP (Vlan Trunking Protokol), je protokol koji služi sa objavljivanje konfigurisanih vlanova kroz lokalnu mrežu. VTP funkcioniše u tri moda: server, klijent i transparent. Server mod objavljuje lokalno konfigurisane vlan-ove kroz mrežu, klijent je sposoban samo da uči vlan konfiguracije koje server objavio, dok transparentan mod funkcioniše sam za sebe u potpunosti, bez mogućnosti da uči ili objavljuje lokalno konfigurisane vlanove. U svakoj LAN mreži na svakoj od lokacija, jedan switch će da radi u VTP server modu dok će drugi da radi u VTP klijent modu.



SL.2.2. Šema LAN povezivanja

Lan konfiguracija:

Kreiranje VLAN-ova:

```
Switch3560(config)#vlan 11
Switch3560(config-vlan)#name Data vlan
```

Podesavanje VTP protokola i imena VTP domena (parameter potreban za VTP komunikaciju):

```
Switch3560(config)#vtp mode server|client|transparent
Switch3560(config)#vtp domain name Dunjainc
```

Konfiguracija 802.1w Spanning Tree parametara. Na jednom switch-u će biti spusten prioritet (početna vrednost prioriteta je 32768) kako bi preuzeo ulogu RSTP root-switch-a.

```
Switch3560(config)#spanning-tree mode rstp
Switch3560(config)#spanning-tree vlan 11,12,13 priority 0
```

Interfejsi ka krajnjim hostovima će biti konfigurisani u access modu, i na njima će biti iskonfigurisan port-fast. Port-fast je način da se linkovima koji vode ka krajnjim hostovima omogući brz pristup i učestvovanje u mreži, i da se zaobiđe konveregencija 802.1w Spanning Tree protokola.

```
Switch3560(config)#interface fa0/1
Switch3560(config-if)#switchport mode access
Switch3560(config-if)#switchport access vlan 11
Switch3560(config-if)#spanning tree port-fast
Switch3560(config-if)#no shutdown
```

Između dva switcha, iskonfigurisan je Layer 2 EtherChannel. EtherChannel će biti trunk interfejs. Trunk interfejsi su interfejsi koji mogu prenositi saobraćaj različitih VLAN-ova. Enkapsulacija na ovim interfejsima će biti industrijski standardizovana 802.1q enkapsulacija.

```
Switch3560(config)#interface port-channel 1
Switch3560(config-if)#no shutdown
Switch3560(config-if)#channel-group 1 mode on
Switch3560(config-if)#switchport mode trunk
Switch3560(config-if)#switchport trunk encapsulation dot1q
Switch3560(config)#interface range fa0/1 - fa0/8
Switch3560(config-if)#no shut
Switch3560(config-if)#channel-group 1 mode on
```

Konfiguracija SVI interfejsa radi inter-vlan komunikacije.

```
Switch3560(config)#interface vlan 10
Switch3560(config-if)#ip address 10.1.0.254 255.255.255.0
Switch3560(config-if)#no shutdown
```

3. Rutiranje, Napadi na OSPF i protivmere

Rutiranje internog saobraćaja će biti sprovedeno OSPF ruting protokolom. OSPF (Open Shortest Path First) je standardizovan ruting protokol, široko korišćen u IP mrežama. OSPF je link-state ruting protokol koji rutiranje bazira na ceni linkova (propusna moć, brzina linkova). Arhitektura samog protokola iziskuje potrebu za segmentiranjem ruting domena, zbog optimizacije celokupnog rutiranja i iskorišćenja hardverskih resursa samih uređaja. OSPF ovo postiže konceptom area. Area je logički domen rutiranja. Još jedna prednost aree kao OSPF koncepta je ta što je tako omogućena adresna sumarizacija. Neophodno je implementirati jednu *backbone areu* (Area 0), koja će u ovom slučaju biti implementirana od strane telekoma u okviru L3 MPLS VPN oblaka. Ostale lokacije će biti u zasebnim *areama* (*area* 1,2,3). Obzirom da interna mreža ima jedan izlaz na internet, preko rutera u Beogradu, na njemu će biti konfigurisana default ruta, koja će potom biti objavljena kroz OSPF domen. Konfiguracija OSPF:

Beograd :

```
Router2911(config)#router ospf 1
Router2911(config-r)#router-id x.x.x.x /definisanje OSPF identifikatora koji mora biti unikatan za
svaki OSPF uređaj u mreži
Router2911(config-r)#network 10.0.0.0 0.255.255.255.255 area 1
Router2911(config-r)#area 1 range 10.1.0.0 255.255.0.0
Router2911(config-r)#default-information originate always /objavljivanje default rute kroz domen
Router2911(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0
```

Novi Sad :

```
Router2911(config)#router ospf 1
Router2911(config-r)#router-id x.x.x.x /definisanje OSPF identifikatora koji mora biti unikatan
za svaki OSPF uredjaj u mreži
Router2911(config-r)#network 10.0.0.0 0.255.255.255.255 area 1
Router2911(config-r)#area 2 range 10.2.0.0 255.255.0.0
```

Niš :

```
Router2911(config)#router ospf 1
Router2911(config-r)#router-id x.x.x.x
Router2911(config-r)#network 10.0.0.0 0.255.255.255.255 area 1
Router2911(config-r)#area 3 range 10.3.0.0 255.255.0.0
```

Switch

```
Router2911(config)#router ospf 1
Router2911(config-r)#router-id x.x.x.x
Router2911(config-r)#network 0.0.0.0 255.255.255.255.255 area x
Router2911(config-r)#area x range 10.1.0.0 255.255.0.0
```

Zbog brze konvergenije u slučaju pada linka, tajmeri kojima ospf prepoznaje pad linka biće smanjeni kako bi se ceo process ubrzao. U slučaju potrebe, može se i modifikovati OSPF cena linka.

```
Router2911(config)#interface fa0/0
Router2911(config-if)#ip ospf hello-interval 1
Router2911(config-if)#ip ospf dead-interval 3
Router2911(config-if)#ip ospf cost 10
```

Kada koristimo link state protokole, svaki ruter radi *fluding* stanja svoje veze sa svojim susedima koji zauzvrat prosleđuju informacije drugim ruterima u mreži. Rezultat toga je da svaki ruter ima informacije o vezama svih rutera u mreži, i na kraju ima sliku cele topologije mreže. Open Shortest Path First (OSPF) je najpopularniji link state protokol.

Operacije OSPF protokola se sastoje iz tri faze:

1. Sastati se sa susedima
2. Razmena informacija veze
3. Izračunavanje najkraće rute

Nakon sastanka OSPF suseda u fazi jedan, svaki OSPF ruter stavlja informacije stanja svog interfejsa i susedstva u link state oglašavanje (Link State Advertisement (LSA)) i radi *fluding* ruting domena u fazi 2. Fluding u LSA može biti periodičan i kad god informacija nosi promene (na primer kada se ukloni link). Nakon sakupljanja LSA-ova od suseda, ruter može da gradi link state bazu koja predstavlja topologiju mreže. U trećoj fazi baza se koristi da izračuna najkraći put do destinacije. Faza dva je od ključnog značaja za pravilnost ruting operacija. LSA je primljen, sadržaj link state baze i izračunata najkraća putanja će postati netačne.

3.1. Link – state advertisement (LSA)

Polje starosti link-state-a odnosi se na starost LSA u sekundama. Vodeći ruter i svi ruteri koji propagiraju LSA će povećati starost LSA. U vodećem ruteru prvobitno je postavljena na 0, i dodata nekom vrednošću (definisana InfTransDelay-om) svakim skokom u *flooding* proceduri.

LS starost je takođe uvećana, kako se LSA nalazi u ruterovoj bazi podataka. Kada starost dostigne maksimalnu vrednost definisanu od MaxAge-a, smatra se da je zastareo i treba da bude obrisani iz baze rutera. Za Cisco rutere MaxAge je podešen na 3600 sekundi, mada to nije maksimalna vrednost na koju se polje može podesiti.

Kada ruter primi dve LSA instance koje imaju indentičnu vrednost LS senkvencijalnog broja i kontrolnog zbira, LS polje starosti će biti ispitano. Instanca koja ima manju starosnu vrednost je prihvaćena kao najnovija, osim u posebnom slučaju MaxAge-a. Instanca koja je dostigla starost MaxAge-a je uvek prihvaćena. Da bi dozvolio starim LSA-ovima da budu brzo izbrisani iz domena rutera. Polje kontrolnog zbira LS-a koristi se da detektuje korupciju podataka LSA.

Proračun kontrolnog zbira se zasniva na kompletnom sadržaju, osim polja starosti LSA. On omogućava da starost LSA može biti uvećana bez ažuriranja kontrolnog zbira.

Postoji razlika između LSA i LS instance. LSA je povezan sa određenom vezom. Na primer ako ruter A ima link L do rutera B, ruter A mora da pokrene LSA opisujući taj link i *flood*

rutiranja domena. LSA može biti jedinstveno indentifikovan od vrste LS-a, LS ID-a, i oglašavanja ruterovih ID polja. Prikazano na slici 3.1.

0	8	16	24	31
Starost LS-a		Opcije		Tip LS-a
LS ID				
Matični Ruter				
LS sekvencionalni broj				
LS kontrolni zbir			Dužina	

SI.3.1. OSPF LSA zaglavlja

LSA instanca se odnosi na stanje određenog LSA na određeno vreme. Na primer, u vreme t_1 , ruter radi *flooding* LSA instance opisujući da je izdatak linka L 100; međutim kasnije, u vreme t_2 ruter A menja izdatak linka L na 200, onda ruter A mora da radi *flood* druge LSA instance, navodeći novi izdatak. Dakle u ruting domenu, može biti nekoliko instanci LSA. Kako LSA instanca može biti jedinstveno indentifikovana LS sekvencom, LS kontrolnim zbirom i Poljem starosti LS-a, da bi odredili koja je instanca skorija, ova tri polja moraju biti ispitana.

3.1.1. Kriptografsko rešenje za Link State

Kriptografija je uobičajen pristup da se izbegne ubacivanje lažnih LSA-ova od strane napadača. U ovom odeljku su predstavljene neke od tehnika za odbranu OSPF mreže koje su zasnovane na kriptografiji. U distance vector protokolima ruter rezimira primljene informacije od susednog rutera pre nego sto pošalje svoje. U ovom slučaju nije moguće da se indentifikuje izvorni ruter na datoj mreži. Pošto se izvor ne može utvrditi, kriptografske tehnike se ne mogu koristiti da odrede autentičnost izvora. Međutim u link state protokolima, ruter radi *flooding* svojih link state informacija svakom ruteru u mreži. Što znači da svaki ruter u mreži dobija link state informacije od izvornog rutera. U ovom slučaju izvorni ruter može biti indentifikovan, tako da se kriptografske tehnike mogu koristiti. Osnovna ideja obezbeđenja link state protokola koristeći kriptografske tehnike je da se doda digitalni potpis na svaki OSPF LSA. Ruter koji počne oglašavanje označava link state informacije (LSA) koristeći privatni ključ, i radi *flood* ostalim ruterima u mreži. Ruteri koji primaju informacije tada verifikuju potpis potpisanog LSA koristeći privatni ključ rutera koji oglašava. Ako verifikacija potpisa ne uspe, LSA će biti odbačen. Ako je verifikacija uspešna, garantuje da dobijeni podaci nisu menjani tokom tranzita u mreži i da dolaze od rutera koji oglašava.

Primer konfiguracije MD5 kriptografije u OSPF-u:

```
Cisco2911(config)#interface serial0/0
Cisco2911(config-if)#ip ospf authentication message-digest
Cisco2911(config-if)#ip ospf message-digest key 1 md5 CISCO
```

3.1.2. Specijalni tretman LS-a starosnog polja

LS starosno polje u OSPF LSA zaglavlju biće povećano svakim skokom tokom *flooding* procedure. Posto polje treba da se ažurira od strane intermedijarnih rutera, generalno ne može biti pokriveno u potpisu. Međutim ako polje nije zaštićeno, napadači mogu falsifikovati polje da sruše ruting protokol.

Jedan od primera ove vrste napada je MaxAge napad. Procedura napada MaxAge-a je sledeća:

Kada napadač primi LSA, promeni njegovu starost na MaxAge koji je po standardu 1 sat i radi *flood* modifikovanog LSA u mreži. Kada ostali ruteri prime modifikovan LSA, obrisaće odgovarajući LSA iz svoje baze kao da je maksimalna starost dostignuta. Međutim u isto

vreme, matični ruter takođe prima modifikovani LSA od napadača. Prema specifikacijama OSPFv2, matični ruter će uzvratiti tako što će generisati novi LSA sa tačnim link informacijama i novim sekvencionalnim brojem. Kao rezultat toga, ruteri koji su izbrisali link iz svoje baze primaju novi LSA koji im govori da je link zapravo dostupan. Ako napadač nastavi da generiše LSA-ove sa MaxAge-om, mreža će postati nestabilna. Da bi zaštitili starosno polje i sprečili napad MaxAge-a, potpis može zaštititi starosno polje samo onda kada je starosna vrednost MaxAge. Drugim rečima, MaxAge LSA-ovi moraju biti zaštićeni digitalnim potpisom.

3.1.3. Prednosti i mane

Upotreba digitalnog potpisa ima dve velike prednosti. Pored toga što osigurava da podaci stvarno dolaze samo od strane matičnog rutera, osigurava i da podaci nisu modifikovani u tranzitu. Prema tome, naročito je koristan da oslabi razne načine napada na link. Sa druge strane, ako su primljeni netačni podaci rutiranja, potpis pripojen podacima rutiranja može se koristiti da nađe izvor problema. Iako korišćenje digitalnog potpisa sprečava napade na link, ne može sprečiti kompromitovan ili neispravan ruter od potpisivanja netačnih routing informacija koje će biti *flood-ovane* u mreži. Ipak šteta bi bila minimalna. Kompromitovani ruter bi mogao da objavi sledeće:

1. Link sa netačnom metrikom
2. Link sa netačnom tvrdnjom
3. Link koji zapravo ne postoji

Ne postoji način da se zaštite od prva dva slučaja, ali neželjeni efekti bi bili lokalizovani. U trećem slučaju OSPF Dijkstra izračunavanje najkraće rute neće uzeti u obzor objavljeni nepostojeći link, zato što nema sličnih objava od rutera na kraju tog linka. Druga ograničenja korišćenja digitalnog potpisa uključujući opterećenje sistema, je zahtev PKI-a, kao i potrebne modifikacije protokola.

3.2. Hash Chains za Stable Link State (SLS)

Kriptografija sa javnim ključem može se koristiti u OSPF protokolu da obezbedi sigurnost mreže. Bez obzira na efektivnosti, to zahteva skupe generacije i verifikacije digitalnih potpisa. To je više problem za protokole rutiranja uključujući mnogo ažuriranja rutiranja, pošto digitalni potpisi moraju biti generisani i verifikovani u realnom vremenu. Primećeno je da su u OSPF saobraćaju, nakon ažuriranja rutiranja za promenu link, mnoge naknadne ispravke jednostavno ponovo izveštavaju o ažuriranju, i ne sadrže nove informacije. Na osnovu ovog

posmatranja, rešenje pod nazivom Stable Link State (SLS) je predloženo da smanji kriptografske troškove zajedno sa ažuriranjem rutiranja, procesiranjem i distribucijom.

Ključna ideja SLS-a je da se koristi jedan lanac heševa kao tokeni za autentifikaciju. Da bi generisali lanac heša, prvo mora biti kreiran slučajan tajni kvantitet R . Kvantitet R je hešovan n puta koristeći jaku jednosmernu heš funkciju H , kao što su SHA i MD5. Uzastopne heš vrednosti mogu se lančati na sledeći način: $H^1(R)$, $H^2(R)$, ..., $H^i(R)$, ..., $H^n(R)$ gde je $H^i(R) = H(H^{i-1}(R))$.

SLS rešenje kreira dve nove ažurirane link-state poruke: Anchor Link State Update (ALSU) i Chained Link State Update (CLSU). ALSU je potpisan i koristi se kad god je link state izmenjen ili je trenutni lanac heša iscrpljen, dok je CLSU nepotpisan, i koristi se kada nema izmene link state-a ali CLSU mora biti poslat.

ALSU uglavnom sadrži:

- n – dužina lanca
- $H^n(R)$ - Anchor vrednost
- Vremenska oznaka- za vremenske intervale
- LSU- uobičajene informacije sadržane u OSPF LSU koje zapravo opisuju link state
- Digitalni potpis- za upotrebu autentifikacije

CLSU uglavnom sadrži:

- i – indeks koji poziva broj ispravki posle ALSU
- $H^{n-i}(R)$ – odgovarajuća vrednost heša
- Vremenska oznaka – za vremenske intervale

CLSU ne sadrži potpis. Svrha toga je da se smanji dodatno procesiranje potpisa.

Generisanje i verifikacija ALSU-a:

ALSU je generisan kada čvor detektuje promenu link state-a ili kada je trenutni lanac heša prazan. ALSU je digitalno potpisan od strane matičnog rutera tako da je integritet skoka osiguran. Nakon uspešne verifikacije potpisa, prijemni ruteri skladište u lokalni anchor vrednost, $H^n(R)$, i ostale parametre nošene u primljenom ALSU za buduću upotrebu verifikacije.

Generisanje i verifikacija CLSU-a:

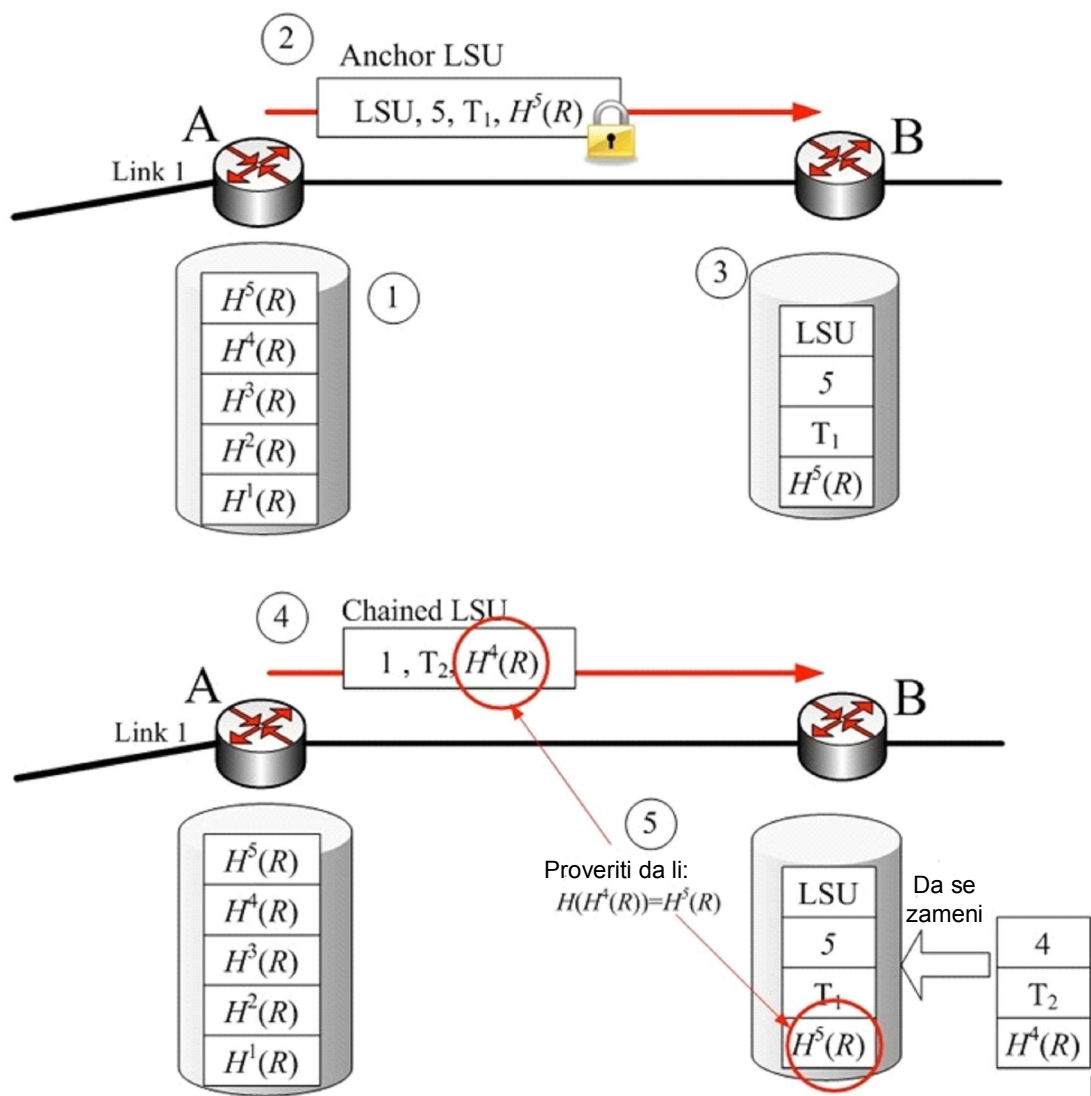
Nakon slanja ALSU-a, CLSU je generisan ako nema promene link state-a, ali novi LSU treba biti poslat. Svaki CLSU sadrži indeks i i odgovarajući $H^{n-i}(R)$. Indeks ukazuje na broj ispravki posle originalno potpisanog ALSU-a, i $H^{n-i}(R)$ se koristi za autentifikaciju.

Na primer ako pretpostavimo da je $n = 5$, prvi CLSU sadrži $i = 1$ i $H(R)$, drugi CLSU sadrži $i = 2$ i $H^2(R)$, i tako dalje.

Ako je i veće od n , heš lanac se smatra osiromašenim, i novi ALSU mora biti generisan. Kada ruter primi CLSU, nema potrebe za verifikacijom potpisa (pošto CLSU nije digitalno potpisan). Umesto toga prijemni ruter verifikuje CLSU tako što proverava $H(H^{n-i}(R)) = H^{n-i+1}(R)$. Na primer ako pretpostavimo da je $n = 5$, kada ruter primi CLSU koji sadrži $i = 1$ i $H^4(R)$, on proverava $H(H^4(R)) = H^5(R)$.

Primer

Predpostavimo da je dužina hash chain-a 5, matični ruter A, izračunava hash chain na osnovu slučajne količine R. U vreme T_1 , ruter A detektuje promenu stanja Linka 1, on šalje ALSU (za link 1) koji sadrži vrednost $H^5(R)$ svom susedu, ruteru B. Nakon uspešne verifikacije potpisa ruter B beleži parametre koje nosi ALSU. Kasnije u vreme T_2 ruter A mora da pošalje novi LSU opisujući stanje Linka 1. Pošto nema promene linka, ruter A generiše CLSU koji sadrži: indeks $i = 1$ (nagoveštavajući da je to prva promena od poslenjeg ALSU-a) $H^4(R)$ (odgovarajuća heš vrednost). Nakon što je primio CLSU, ruter B ga verifikuje tako što računa heš od primljenog $H^4(R)$ i poredi rezultat sa uskladištenim $H^5(R)$. Ako je verifikacija uspešna, stari parametri će biti zamenjeni onim koji su sadržani u ovom CLSU za buduću upotrebu verifikacije. Istovetno, sledeći CLSU sadrži $H^3(R)$ i ruter B ga verifikuje proveravanjem $H(H^3(R)) = H^4(R)$. Ako pretpostavimo da stanje linka nije promenjeno od vremena T_1 i da su poslate četiri CLSU poruke koristeći $H^4(R)$, $H^3(R)$, $H^2(R)$, i $H^1(R)$, naizmenično. Sada je dostignuta *hash chain* dužina $n = 5$, i smatra se da je lanac istekao. Tako da ruter A mora da kreira novi *hash chain* koristeći nova svojstva na primer 'R' i da generiše novi ALSU sa sledeći izmenjeni interval. Na slici 3.2. je prikazan primer koji ilustruje operaciju SLS-a.



SI.3.2. Operacija SLS-a

3.2.1. Prednosti i mane

Glavna prednost ovog metoda je nizak stepen upotrebe dodatnih operacija. Dodatne operacije koje se koriste za generisanje CLSU-a su zanemarljive. To je zbog toga što su vrednosti heša, $H^d(R)$, su unapred izračunate. Sa druge strane, verifikacija CLSU-a je minimalna zato što se izračunava samo jedna vrednost heša. Samo treba proveriti ako je $H(H^{n-l}(R)) = H^{n-l+1}(R)$, koji se može tumačiti kao $H(\text{ova primljena vrednost heša}) = \text{poslednjoj uskladištenoj vrednosti heša}$. Pošto upotreba CLSU-a ne zahteva skup proračun potpisa ili verifikaciju potpisa, to značajno smanjuje dodatne operacije koje se koriste u tradicionalnoj šemi digitlanog potpisa.

Međutim kad kod ima promene stanja linka, novi ALSU mora biti generisan. Pošto su ALSU-ovi potpisani, ako je link promenljiv mnogo ALSU-ova će biti generisano. U tom slučaju, biće visok stepen proračuna potpisa, i prednost SLS-a će biti umanjena.

3.3. Hash Chains za Fluctuating Link State

SLS šema koristi prednost situacije, pošto mnogo LSU-ova su jednostavno ponovljeni iskaz, tako da ne nose nove informacije. Međutim efektivno je samo ako je stanje veze stabilno. U mrežama gde se link često menja predložena je šema Fluctating Link State (FLS).

Koncept i operacije FLS-a su slične onima od SLS-a. Glavna razlika je ta što SLS koristi jedan *hash chain*, a FLS koristi dva *hash chain-a* za svaki incidentni link. Jedan lanac je za gornje stanje a drugi je za donje stanje linka. De različite funkcije heša se koriste za ova dva lanca. Na primer heš funkcija F je za donji lanac pošto je heš funkcija H za gornji lanac. Sa druge strane svaki link koristi jedinstven i slučajno generisan kvantitet, R_1 . Tako da ako ruter ima k incidentnih linkova, a dužina lanca je n , tada ruter generiše heš tabelu koja sadrži $(n \times k \times 2)$ heš vrednosti. Raspored tabele je prikazan na slici 3.3.

Link l			Link j			Link k	
Gornji	Donji		Gornji	Donji		Gornji	Donji
$H^1(R_l)$	$F^1(R_l)$	...	$H^1(R_j)$	$F^1(R_j)$	...	$H^1(R_k)$	$F^1(R_k)$
$\vdots$	$\vdots$		$\vdots$	$\vdots$		$\vdots$	$\vdots$
$H^j(R_l)$	$F^j(R_l)$		$H^j(R_j)$	$F^j(R_j)$		$H^j(R_k)$	$F^j(R_k)$
$\vdots$	$\vdots$		$\vdots$	$\vdots$		$\vdots$	$\vdots$
$H^n(R_l)$	$F^n(R_l)$		$H^n(R_j)$	$F^n(R_j)$		$H^n(R_k)$	$F^n(R_k)$

SI.3.3. Tabela hash chain-a

Generisanje i verifikacija ALSU-a:

ALSU koji se koristi u FLS-u uglavnom sadrži:

- *nodeID* - jedinstven ID za svaki čvor rutera
- vremenska oznaka - za vremenske intervale
- $H^n(R_l), F^n(R_l), \dots$, - Set anker vrednosti za sva stanja i sve linkove
 $H^n(R_j), F^n(R_j), \dots$,
 $H^n(R_k), F^n(R_k)$
- LSU - uobičajena informacija nošena u OSPF LSU koja opisuje stanje linka
- Digitalni potpis – za upotrebu autentifikacije

Glavna razlika između ovog ALSU-a i onog koji se koristi u SLS-u je što ovaj sadrži set anker vrednosti za sva stanja i sve linkove.

Po prijemu ALSU-a, prijemni ruter prvo verifikuje potpis. Ako je uspešno i vremenski interval se smatra svežim, informacije nošene u ALSU-u će biti uskladištene.

Generisanje i verifikacija CLSU-a:

Ako matični čvor mora da pošalje novo ažuriranje, on generiše CLSU. CLSU koji se koristi u FLS-u uglavnom sadrži:

- *nodeID* - jedinstven ID za svaki čvor rutera
- vremenska oznaka - za vremenske intervale
- i - indeks koji se odnosi na broj izmena posle ALSU-a
- LSF - *Link State Flag*
- LSV - *Link State Vector*

LSF i LSV su važna polja za prijemne rutere za određivanje promena linka. Pod pretpostavkom da postoje k linkovi, $L_1 \dots L_k$, definisani su kao:

$LSF = [L_1 \text{ stanje}, \dots, L_j \text{ stanje}, \dots, L_k \text{ stanje}]$

$LSV = [LS(1), \dots, LS(j), \dots, LS(k)]$

$$\text{Gde } LS(j) = \begin{cases} H^{n-1}(R_j) & \text{ako je } L_j \text{ stanje} = UP \\ F^{n-1}(R_j) & \text{ako je } L_j \text{ stanje} = DOWN \end{cases}$$

Na primer, ako je $k=3$, $n=4$, $i=2$, i stanja linkova su *up,down*, i *up*, odnosno LSF i LSV postaju:

$LSF[UP, DOWN, UP]$

$LSV[H^2(R_1), F^2(R_2), H^2(R_3)]$

Svaki prijemni čvor skladišti informacije pređašnjeg CLSU-a, označen kao $CLSU_p$, i LSV_p , gde je $p=i-1$. Po prijemu CLSU-a, prijemni ruter obavlja:

1. Traži trenutni pristup za *nodeID*, i potvrđuje svežinu izmene na osnovu vremenske oznake.
2. Proverava d ali su stanja veze koja su nošena u trenutno primljenom CLSU promenjena. To se postiže tako što se poredi sa $CLSU_p$.

Ako je stanje ne promenjeno, izračunava se:

$H^{i-1}(LS(j))$ ako je L_j stanje *UP*

$F^{i-1}(LS(j))$ ako je L_j stanje *DOWN*

i poredi sa predhodno uskladištenom vrednošću u LSV_p ; odbacuje se na osnovu nepodudaranja.

Ako je stanje nepromenjeno, izračunava se:

$H^i(H^{n-1}(R_j))$ ako je L_j stanje *UP*

$F^i(F^{n-i}(R_j))$ ako je L_j stanje *DOWN*

i poredi sa odgovarajućim vrednostima u LSV_n (koji je nošen u ALSU i skladišten je lokalno); odbacuje se na osnovu nepodudaranja

3. Nakon toga, primljeni LSV zamenjuje prednodni *link state vector* (LSV_p)

Primer

Prvo, ruter A izračunava heš vrednosti svojih konetktovanih linkova (Link 1 i Link 2). Predpostavimo da je dužina lanca 4, sa dva linka i dve heš funkcije, postoje $4 \times 2 \times 2 = 16$ heš vrednosti. Onda ruter A generiše ALSU u T_1 , koji sadrži *hash chains* za link 1 i link 2 za svoj susedni ruter B. ALSU takođe sadrži ID i vremensku oznaku matičnog rutera, i ceo proces je zaštićen potpisom matičnog rutera. Nakon uspešne verifikacije potpisa rutera A, ruter B skladišti informacije nošene u ALSU, koje se koriste za verifikaciju sledećeg CLSU-a.

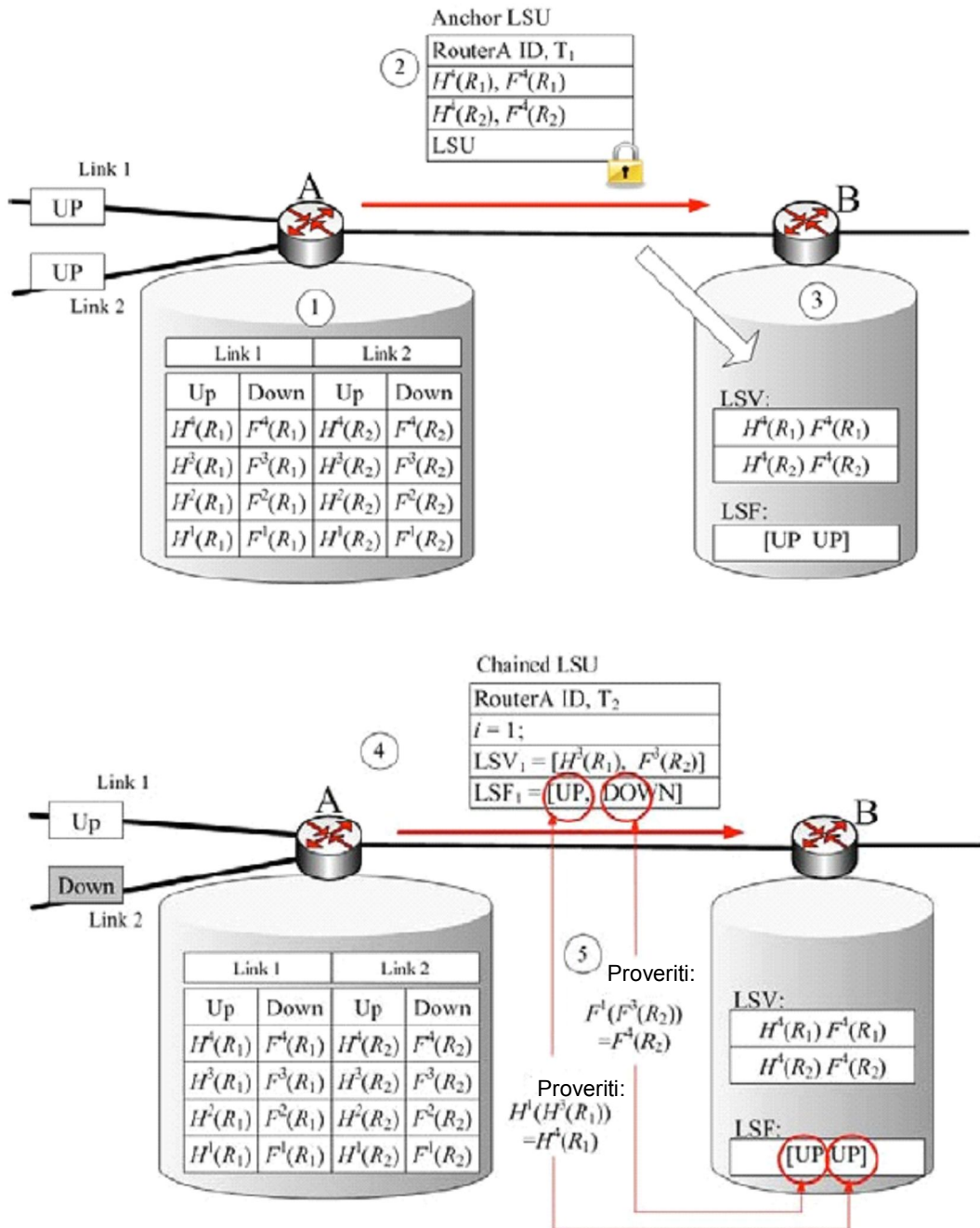
Predpostavimo da u T_2 , Link 2 pada. Ruter A tada generiše CLSU svom susedu za najnovije informacije stanja linka. Pošto su stanja Linka 1 i Linka 2 naizmenično *up* i *down*, LSF i LSV za ovaj CLSU postaju:

$LSF = [UP, DOWN]$

$LSV = [H^3(R_1), F^3(R_2)]$.

Po prijemu CLSU-a, ruter B poredi primljeni LSF i uskladišteni LSF i pronalazi da je stanje Linka 1 nepromenjeno, ali stanje Linka 2 je promenjeno. Dakle, za Link 1 izračunava se $H^1[H^3(R_1)]$ i poredi se sa predhodno uskladištenim $H^4(R_1)$. Za Link 2, izračunava se $F^1[F^3(R_2)]$ i poredi se sa predhodno uskladištenim $F^4(R_2)$.

Nakon ovog proračuna i uspešne verifikacije primljenog CLSU-a, predhodno uskladišteni LSF i LSV će biti zamenjeni ovim upravo primljenim. Na slici 3.3. je prikazan primer FLS operacija.



SI.3.4. FLS operacije

3.3.1. Prednosti i mane

Glavna prednost FLS šeme u odnosu na SLS šemu je to što nema potrebe za skupim proračunima potpisa, ili verifikacije potpisa iako ima promena stanja linka. Prema tome značajno može da smanji dodatnu obradu informacija koja se bavi digitalnim potpisom čak i ako link i čvorišta variraju. Ipak u odnosu na SLS šenu, potrebne su dodatna memorija i snaga obrade heš vrednosti.

4. Context-Based Access Control (CBAC) - Cisco IOS firewall set funkcija

Izabrani uređaji u mreži, pružaju ogromne mogućnosti za uspostavljanje bezbednosnih mehanizama i implementacije bezbednosnih protokola. U mreži će biti implementiran Cisco CBAC router firewall koji će zaštititi ruter u Beogradu od upada i napada na mrežu sa interneta. CBAC, kao firewall rešenja, pruža visok stepen inteligentne zaštite mrežne infrastruktura. Inteligentnom obradom saobraćaja na najvišem nivou (Layer 4 – Layer 7), CBAC pamti i zapisuje uspostavljene konekcije, i u slučaju malicioznog ponašanja određenih paketa i konekcija (iznenadno menjanje portova u TCP konekciji, dug period uspostavljanja konekcije, veliki broj otvaranja konekcije ka specifičnom hostu), CBAC prekida konekciju i zabranjuje sumnjive paketa na odredjeni vremenski period. Pored inteligentne filtracije saobraćaja, pamćenja konekcija, i brzih, pravovremenih reakcija na napade, CBAC takodje odlično prepoznaje razne vrste napada, među kojima su i DDOS napadi. CBAC ima i mogućnost da sarađuje sa aplikativnim proxy serverima. CBAC prati i dozvoljava samo one protokole i samo onaj saobraćaj koji mrežni administrator odredi, ostali su eksplicitno zabranjeni. Kao i svaki statefull firewall (što CBAC i jeste), povratni saobraćaj, koji nema malicioznu tendenciju, se propušta u internu mrežu.

Konfiguracija CBAC:

Router2911(config)#ip inspect name FWALL (protocol) /potrebno je definisati i imenovati CBAC security polisu i odrediti protokole koje će CBAC nadgledati i obezbeđivati. Poželjno je pored UDP, TCP I ICMP saobraćaj, definisati sve specifične protokole koji prolaze kroz mrežu (DNS, HTTP, Citrix, SNMP...).

Router2911(config)#ip inspect tcp synwait-time x/ interval koji CBAC čeka da se uspostavi tcp konekcija pre nego što je zabrani

Router2911(config)#ip inspect tcp finwait-time x /interval koji je potreban da prodje posle primanja TCP FIN paketa pre nego što konekcija bude odbačena

Router2911(config)#ip inspect tcp idle-time x / period neaktivnosti tcp sesije koji je potreban da se ona prekine

Router2911(config)#ip inspect udp idle-time x /period neaktivnosti udp sesije koji je potreban da se ona prekine

Router2911(config)#ip inspect max-incomplete high/low X /maksimalan broj poluotvorenih sesija

4.1. Glavne funkcije CBAC-a

Operacije CBAC-a se sastoje od tri glavne funkcije uključujući inspekciju paketa, održavanje tabele stanja i ažuriranje ulaza access - liste.

Inspekcija paketa odvija pod uslovom da je paket propušten kroz bilo koju relevantnu access-listu. To uključuje:

1. dolazeći ACL iz interne mreže ili;
2. odlazeći ACL u spoljnu mrežu,

CBAC takođe mora biti konfigurisan da bi mogao inspektovati ovaj tip paketa. Treba odrediti interfejs i pravac gde bi inspekcija trebalo da se odvija. Bilo koji paket koji je odbijen od strane access-liste je jednostavno izbačen i neće biti inspektovan.

CBAC koristi inspekciju paketa i održavanje sesije informacija da se poboljša operacija access-liste da bi bila u stanju da uradi sledeće:

- CBAC prati TCP sekvence brojeva i ispušta pakete sa neočekivanim sekvencama brojeva.

- CBAC će prepoznati specifične komande nekih aplikacija koje se mogu koristiti za napade na aplikativnom nivou, i blokirati te pakete.
- CBAC kontroliše UDP sesije približavajući se sesiji informacija korišćenjem UDP podešavanja pasivnog tajmouta.

Tajmout i granične vrednosti se koriste za upravljanje sesijom stanja informacija i važan su deo IOS firewall seta funkcija. CBAC koristi tajmout i granične vrednosti za indentifikovanje sesija koje nisu potpuno uspostavljene, prouzrokujući da te sesije budu odbačene. CBAC može samo da inspektuje protokole koji su specifikovani u skupu pravila za inspekciju, tako da mogućnost za zaštitu od DOS napada se proširuje samo na navedene protokole u setu pravila.

Prateći inspekciju paketa, CBAC kreira ulaz tabele stanja ili ažuriranje postojećih unosa da uključi informacije o stanju sesije. Bilo koji saobraćaj koji se vraća u mrežu iz spoljnog izvora je dozvoljen na osnovu važeće sesije informacija koje se nalaze u tabeli stanja. Informacije sesije u tabeli stanja se konstantno ažuriraju kako saobraćaj prolazi kroz *firewall*.

Ulazi access-liste se dodaju ili brišu dinamički od strane CBAC na osnovu informacija sesije koje su sadržane u tabeli stanja. Ulazi se ubacuju i brišu kako se informacije sesije menjaju. Otvori koji su napravljeni u access-listama su samo privremeni i održavaju se samo dok je sesija validna.

Pre nego što inspekcija saobraćaja počne CBAC pravila inspekcije moraju biti kreirana i dodata na interfejs. CBAC pravila inspekcije se kreiraju i dodaju na interfejs, da li na dolazeći ili odlazeći, na isti način kao i access-liste. Kada paket pokuša da inicira konekciju CBAC će koristiti skup pravila pregleda da utvrdi da li paket treba biti pregledan i sesija stanja praćena.

4.2. Proces CBAC-a

Cisco System istraživanja o CBAC, navodi sled događaja u više detalja, kao što sledi:

- Paket stiže na ruterov spoljni interfejs
- Ruter ispituje paket i proverava ih uz bilo koju dodatnu access-listu na bilo kojim interfejsima kojima paket treba da prođe, i stoga propušta ili odbacuje paket.
- Paket je pregledan od strane CBAC ako pravilo postoji, i informacija o stanju konekcije paketa se snima u novoj tabeli stanja koja je kreirana za ovu konekciju. Ako nema određenog pravila za pregled paketa tog tipa, onda se jednostavno prosleđuje ili odbacuje u skladu sa bilo kojom odgovarajućom access-listom.
- CBAC onda kreira privremene prolaze u ulaznoj access-listi na spoljnom interfejsu da bi dozvolio povratni saobraćaj za ovu konekciju. Ove stavke se održavaju, dodaju ili uklanjaju na osnovu promena stanja konekcije kao u kreiranoj tabeli stanja. Access lista koja je modifikovana mora biti proširena access-lista.

- Odlazni paket se prosleđuje iz spoljnog interfejsa
- Bilo koji povratni paketi za istu konekciju su opet dozvoljeni kroz spoljni interfejs zato što su otvaranja napravljena u unutrašnjosti access-listi od strane CBAC.
- Povratni paketi su pregledani od strane CBAC i stoga se tabela stanja se ažurira. Dalje modifikacije mogu biti napravljene access-listi da odražava trenutno stanje konekcije.
- Tabela stanja je obrisana i b ilo koji ulazi u unutrašnjosti access-liste ove konekcije su obrisani po završetku ili pauze sesije.

4.3. Podržani protokoli

CBAC mora biti konfigurisan za pregled protokola koje želimo da budu pregledani sa liste podržanih protokola. Pregledanje paketa neće početi dok se ne kreira IP lista pregleda, koja uključuje protokole ili sesije koje želimo da budu pregledane. Lista se ona doda na interfejs. Neki paketi ili sesije mogu biti dozvoljene kroz firewall od strane postojeće ruterove access-liste ali nisu pregledani od strane CBAC-a. To može biti ili da nisu navedene na uvid u liste inspekcije ili nisu iz dostupnih lista protokola. I dalje su u stanju da uspostave sesiju i da rade kroz firewall ali neće biti nadgledanja niti revizije nijednog stanja sesije.

Cisco System istraživanja o CBAC navodi podržane protokole kao što sledi; CBAC se može konfigurisati da pregleda sledeće tipove sesije;

- Sve TCP sesije, nezavisno od protokola aplikativnog sloja
- Sve UDP sesije, nezavisno od protokola aplikativnog sloja

Sledeći protokoli aplikativnog sloja mogu biti precizirani za pregled:

- CU-SeeMe (samo *white pine* verzija)
- FTP
- H.323
- HTTP (Java blocking)
- Microsoft NetShow
- Unix R-commands
- RealAudio
- RPC (Sun RPC)
- Microsoft RPC
- SMTP
- SQL*Net
- StreamWorks
- TFTP
- VDOLive

ICMP je često potreba usluga za mnoge mreže ali nije podržan protokol za CBAC pregled. Ovo je primer protokola koji, ukoliko želimo da ga dozvolimo da prođe kroz firewall,

mora da se upravlja od strane tradicionalnih access-lista. CBAC će i dalje dozvoliti protokolu da prođe kroz firewall ali neće pratiti ili pregledati stanje sesije, i prateća revizija koja bi mogla da bude omogućena, će se održati.

Potrebno je napraviti access-listu koja će zabranjivati sav nepotreban saobraćaj osim ako nije prvo prošao kroz firewall inspekciju. Access liste su stateless metod filtracije saobraćaja, koje funkcionišu po principu poređenja određenih polja u paketima sa konfigurisanim parametrima.

Konfiguracija access-liste za ICMP:

```
Router2911(config)# access-list 100 deny tcp any any
Router2911(config)# access-list 100 deny udp any any
Router2911(config)# access-list 100 permit icmp any any echo-reply
Router2911(config)# access-list 100 permit icmp any any time-exceeded
Router2911(config)# access-list 100 permit icmp any any packet-too-big
Router2911(config)# access-list 100 permit icmp any any traceroute
Router2911(config)# access-list 100 permit icmp any any unreachable
Router2911(config)# access-list 100 deny ip any any
```

Na interfejsu koji vodi ka internetu biće implementirani access lista i unapred definisana firewall polisa.

```
Router2911(config)#interface serial0/0
Router2911(config-if)#ip access-group 100 in
Router2911(config-if)#ip inspect FWALL out
```

4.4. Prednosti i ograničenja CBAC-a

CBAC ima niz prednosti nad perimetrim ruterom baziranim na IOS-u koji koristi ACL da kontroliše spoljni pristup unutrašnjoj mreži. *Stateful* inspekcija paketa pruža više sveobuhvatne i fleksibilne alternative za kontrolu protoka saobraćaja nego access-liste, dok se osposobljava Cisco IOS firewall za osnovnu detekciju i prevenciju napada, na primer DoS napadi mogu biti indentifikovani i blokirani.

CBAC kontrolisane sesije i naknadne izmene na access-listama su dinamične i privremene. To značajno smanjuje period vremena za koje je protok saobraćaja dozvoljen u mreži, smanjujući trajanje bilo kakvih propusta za taj tip sesije.

Funkcija upozorenja i revizorskog ispitivanja omogućava sveobuhvatno evidentiranje statistika sesije i može se koristiti u kombinaciji sa *host-based* i *network-based* za detekciju upada za povezivanje informacija i indentifikaciju pokušaja napada.

U okruženjima gde protok saobraćaja u velikoj meri potiče iz unutrašnjosti mreže sposobnost CBAC-a da se dinamički ažuriraju access-liste omogućava znatno jaču kontrolu, jaču konfiguraciju access-liste i potencijalno manje administrativne troškove za upravljanje spoljnim access-listama.

CBAC ima brojna ograničenja koja mogu ograničiti njegovu efektivnost u određenim okolnostima. CBAC podržava samo saobraćaj IP protokola tako da se samo TCP i UDP paketi pregledaju. ICMP saobraćaj se ne pregleda i mora biti omogućen uobičajenim access-listama.

CBAC neće pregledati pakete gde je izvorna ili odredišna adresa sam *firewall*, pristup ruteru sa spoljne mreže mora biti strogo kontrolisan sa tradicionalnim access-listama.

Kompabilnost CBAC-a sa nekim drugim Ciscovim bezbednosnim funkcijama je ograničena zato što CBAC nije u stanju da precizno ispita teret enkriptovanog saobraćaja koji prolazi kroz *firewall* i podrška za pregled protokola je dodatno ograničena ako su CBAC i enkripcija konfigurisani na samom *firewall-u*. CBAC radi samo sa IPSec-om ako je *firewall* krajnja tačka IPSec-a za protok saobraćaja, CBAC ne može pregledati zaglavlje IPSec paketa koji prolazi kroz *firewall*. Redundantni IOS *firewall-i* nisu podržani jer stanje *firewall* sesije je unutar samog rutera.

Kada su omogućeni, Cisco IOS *Firewall* set funkcija utiču na korišćenje resursa sistema, i mora se osigurati da ruter ima dovoljno memorije i procesorske snage da bi zadovoljio zahteve performansi.

Bez obzira koliko je dobra konfiguracija CBAC-a, ruter je otvoren za napad ako nije instaliran softver za *firewall*. Politika bezbednosti takođe terba da bude napisana i ažurirana, da bi se formirala osnova konfigurisanja bezbednosti na samom ruteru, i takođe treba konfigurisati CBAC politiku za inspekciju. Bez obzira da li konfigurisani Cisco IOS *firewall* radi na perimetnom ruteru ili ima odvojeni *firewall* i dalje treba misliti o tom ruteru kao odvojenom uređaju od firewall-a i primeniti principe “dubinske odbrane”, radi osiguravanja bezbednosti na svim nivoima mreže. Perimetni ruter i *firewall* rade zajedno da bi poboljšali bezbednost mreže i obezbedili odbranu perimetra. Izgradnja Cisco IOS *firewall* instalacije na bezbedno konfigurisanom eksternom ruteru je od suštinske važnosti jer:

1. CBAC se oslanja na efektivnost konfiguracije access-lista na ruteru i
2. CBAC ne može pregledati saobraćaj sa izvorne ili odredišne adrese rutera.

To znači da se ne može osloniti na CBAC da obezbedi bilo kakvu sigurnost za sam ruter.

5. Bezbednost kontrole pristupa

Kontrola pristupa je sistem koji omogućava autoritet koji će da kontroliše pristup oblastima i resursima u datom fizičkom objektu ili IT sistemu.

Postoje tri komponente kontrole pristupa:

- Određivanje kome je dozvoljen pristup mreži
- Određivanje kojim servisima mogu pristupiti
- Pruža detaljne računovodstvene evidencije o servisima kojim se pristupalo

Kontrola pristupa će biti implementirana uz pomoć radius servera i AAA mehanizma. AAA mehanizam je Cisco način da centralizuje kompletno rešenje pristupa mreži. AAA vrši tri funkcije, autentifikaciju, autorizaciju i accounting. Lokalno iskonfigurisane polise daju ruteru mogućnost da komunicira sa eksternim radius serverom postavljenim u mreži. Korisnici koji pristupaju uređajima budu autentifikovani, autorizovani i sav njihov rad na uređajima se beleži na radius serveru. Radius server se instalira na server jedinici, i potrebno je uspostaviti UDP konektivnost između servera i samih Cisco uređaja.

Konfiguracija kontrole pristupa:

```
Router2911(config)#aaa new-model
Router2911(config)#radius-server host x.x.x.x /ip addressa radius servera
Router2911(config)#aaa authentication login default radius local
Router2911(config)#aaa authentication attempts login 3
Router2911(config)#aaa authorization exec default radius
Router2911(config)#aaa authorization commands default radius
Router2911(config)#aaa accounting exec default start-stop
Router2911(config)#aaa accounting commands default start-stop
```

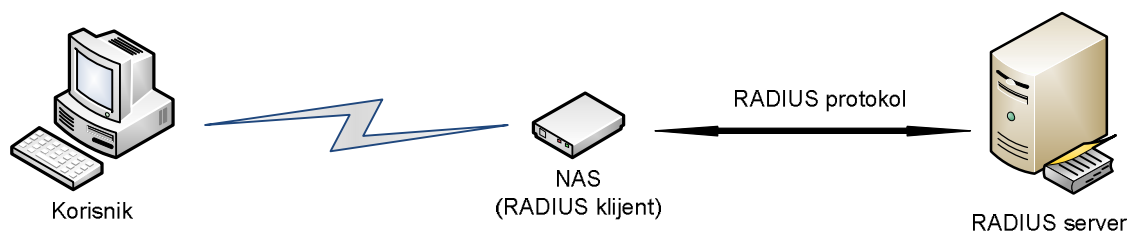
5.1. RADIUS (Remote Authentication Dial In User Service)

RADIUS je AAA protokol tj. protokol koji se koristi za proveru indentiteta korisnika, njihovu autorizaciju i obračun usluga korisnicima. Najčešće se primenjuje za proveru indentiteta korisnika na mrežnim uređajima, ruterima i modemima, koji zbog ograničenih hardverskih resursa kojim raspolažu, ne mogu da čuvaju veliki broj parametara za proveru indentiteta različitih korisnika. RADIUS ima i mehanizam centralizovanog administriranja korisnika, što je jako pogodno u okruženjima gde treba administrirati mnogo korisnika. Ukoliko se uz to u obzir uzme i činjenica da se protokol RADIUS danas smatra *de facto* standardom za daljinsku

proveru identiteta korisnika, većina proizvođača mrežne opreme, zbog toga, u svoje uređaje ugrađuje podršku za RADIUS.

RADIUS se zasniva na troslojnoj klijent/server arhitekturi i koristi usluge UDP protokola transpotnog sloja (portovi 1812 i 1813 za proveru indentiteta). ISP zahteva da se unese korisničko ime i lozinka. Te informacije se šalju preko protokola PPP serveru za pristup mreži tj. NAS uređaju (*Network Access Server*). NAS uređaj šalje te podatke RADIUS serveru preko protokola RADIUS. RADIUS server pomoću PAP, CHAP ili EAP protokola proverava da li su podaci koji su zadati ispravni. Ukoliko su uneti podaci, server će odobriti pristup mreži, tj. posalti odgovor NAS uređaju koji će dalje dodeliti IP adresu i odgovarajuće parametre. RADIUS server takođe može biti obavešten o početku i završetku sesije korišćenja mreže, što omogućava naplaćivanje usluga prema vremenu korišćenja, a podaci o korišćenju mreže mogu se dalje upotrebiti za statističku analizu.

RADIUS je incijalno razvila kompanija Livingstone Enterprises za svoju Port-Master seriju servera za pristup mreži, ali je kasnije (1997. godine) objavljen u dokumentima RFC 2058 i RFC 2059 (Tekuće verzije su RFC 2865 i RFC 2866). Trenutno postoji nekoliko RADIUS serverskih paketa- neki su komercijalni, a neki pripadaju kategoriji softvera sa otvorenim kodom (*open source*). Ovi serverski paketi su slični po tome što je većina sposobna da čuva podatke o korisnicima u obliku tekstualnih datoteka, različitih baza podataka ili na LDAP serverima, kao da i informacije o korišćenju usluga izvozi u tekstualne datoteke ili razne baze podataka. Treba uzeti u obzir da je RADIUS proširiv, što većini proizvođača RADIUS hardvera i softvera omogućava da implementiraju protokol shodno svojim potrebama.



SI.5.1. RADIUS

Protokol RADIUS se koristi u sigurnosnom standardu za bežične mreže 802.1x. Iako mu to nije osnovna namena, u sprezi sa protokolom EAP značajno povećava nivo sigurnosti u odnosu na WEP standard.

5.2.Protokol RADIUS

Klijent i server razmenjuju podatke preko RADIUS paketa enkapsuliranih u UDP paketima protokola transportnog sloja. Jedan RADIUS paket sadrži sledeće polja:

- Kod (*code*). Polje dužine jednog bajta na osnovu vrednosti definiše tip RADIUS paketa- npr, zahtev za pristup mreži (Access- Request) ili prihvatanje pristupa (Access Accept).
- Identifikator (identifier, ID). Polje dužine jednog bajta koji omogućava jednoznačnu identifikaciju parova zahtev - odgovor.
- Dužina (length). Polje dužine dva bajta koje određuje veličinu paketa.
- Autentifikator (Authenticator). Vrednost koja se koristi za proveru ispravnosti odgovora koji šalje RADIUS server i za zaštitu korisničke lozinke.
- Atributi (Attributes). Deo paketa u kome se nalaze proizvoljni atributi koji pripadaju samoj sesiji (zahtevu ili odgovoru). Jedini obavezni atributi su korisničko ime i lozinka u zaštićenom obliku (User-Name i User-Password). Ostali atributi su opcioni.

Primer

Tipičan postupak provere identiteta u protokolu RADIUS

NAS (RADIUS klijent) želi da na RADIUS serveru proveri parametre identiteta (korisničko ime i lozinku) korisnika koji može da pristupi bazi podataka sa opisima korisnika.

Na osnovu korisničkog zahteva, RADIUS klijent šalje RADIUS serveru upit sa navedenim korisničkim imenom i lozinkom. Na osnovu postavljenog upita, server obrađuje korisnički zahtev i - zavisno od projavljenih parametara- šalje klijentu odgovarajući paket. Na osnovu sadržaja primljenog paketa NAS server korisniku dozvoljava ili zabranjuje pristup resursima. Detaljnije, provere identiteta odvija se na sledeći način:

Klijentov zahtev

Klijent započinje sesiju generisanja zahteva (RADIUS Access-Request paket), Koji obavezno sadrži dva atributa korisnika: User- Name i User-Password. Bajt identifikacije (ID) tog paketa klijent bira proizvoljno (algoritam za generisanje ovog broja nije zadat u protokolu). Paket takođe sadrži vrednost Request Authenticator u polju Autentifikator. Ta vrednost je takođe 16-bajtni niz koji generiše generator pseudo slučajnih sekvenci. Algoritam generisanja ovog niza značajan za sigurnost ovog protokola, ali tip generatora nije zadat u protokolu i određuje se pri konkretnoj implementaciji protokola. RADIUS paket nije zaštićen ni na koji način izuzev atributa User-Password koji je šifrovan. Na primer ako je K tajni ključ koji dele klijenti servera, a RA pseudo slučajna vrednost Request Authenticator. Lozinka se deli na 16-bajtna blokov $p_1, \dots, p_n$, pri čemu se poslednji blok po potrebi dopunjuje nulama. Dalje se radi sledeće:

- $c_1 = p_1 \text{ xor MD5}(S + RA)$
- $c_2 = p_2 \text{ xor MD5}(S + c_1)$
- ...
- $c_n = p_n \text{ xor MD5}(S + c_{n-1})$

Zaštićeni atribut User-Password dobija se spajanjem dobijenih blokova $c_1, \dots, c_n$.

Serverov odgovor

Nakon primljenog zahteva, server proverava da li postoji tajni ključ koji deli sa RADIUS klijentom. Ako server ne poseduje tajni ključ tog klijenta, zahtev se odbija. Ukoliko takav ključ postoji, server dešifruje vrednost atributa User-Password i dobija korisničku lozinku. Nakon toga server proverava ispravnost lozinke; ukoliko je lozinka validna, server klijentu vraća paket Access-Accept. Lozinka ne odgovara paru korisničko ime-lozinka koji server čuva u svojoj bazi, klijentu se šalje paket Access-Reject kojim se odbija nastavak provere identiteta. U oba slučaja vrednost polja ID paketa koji server šalje, identična je toj vrednosti u klijentovom zahtevu. Vrednost atributa *Response Authenticator* dobija se primenom MD5 heš funkcije na vrednost *Response Authenticator* klijentovog zahteva.

Obrada serverovog odgovora

Kada primi odgovor, klijent proverava da li su vrednosti u poljima ID zahteva i odgovora identične, i na osnovu toga određuje da li se odgovor zaista odnosi na njegov zahtev. Nakon toga se proverava polje *Response Authenticator* primljenog paketa. Klijent generiše svoju vrednost na isti način i upoređuje je sa dobijenom; ukoliko se vrednosti razlikuju, odgovor se smatra neregularnim i sesija se prekida. Ako je klijentu vraćen Access-Accept RADIUS paket ispravnog sadržaja, korisničko ime i lozinka smatraju se regularnim; Klijent smatra da je identitet korisnika uspešno proveren i odobrava mu pristup mrežnim resursima. Ukoliko je klijentu vraćen *Access- Reject* RADIUS paket ispravnog sadržaja, korisničko ime i lozinka smatraju se neregularnim, pa klijent ne dozvoljava korisniku da pristupi mrežnim resursima.

5.2. Problemi sa protokolom RADIUS

Pri upotrebi protokola RADIUS javlja se nekoliko pitanja o sigurnosti, koje su posledica arhitekture i implementacije protokola. Kao prvo, postavlja se pitanje o adekvatnosti algoritma za zaštitu korisničke lozinke. Za šifrovanje lozinke ne sme se koristiti algoritam koji pripada grupi protočnih algoritama; algoritam korišćen u protokolu RADIUS, nažalost, pripada toj grupi i kao generator pseudo slučajnih sekvenci koristi MD5 heš funkciju koja se, u ovom slučaju, smatra neprikladno odabranim alatom. Dalje, upit *Access-Request* koji se šalje serveru ne sadrži nikakav element koji će omogućiti RADIUS serveru da proveriti identitet klijenta. Dovodi se u pitanje i prediktivnost (predvidivost) generatora slučajnih brojeva za generisanje atributa *Request Authenticator*. U suštini, sigurnost koju protokol RADIUS pruža najviše zavisi od kvaliteta algoritma, za generisanje atributa *Request Authenticator*. Ukoliko želimo da obezbedimo sigurnost pomoću protokola RADIUS, pomenuti atribut mora biti jedinstven i nepredvidljiv. Pošto sama specifikacija protokola ne ukazuje na važnost postupka generisanja ovog atributa, ponekad se koriste loše i površne implementacije mehanizma za generisanje pseudoslučajnih sekvenci. Trenutno ne postoji protokol koji bi potpuno zamenio RADIUS, tj. protokol koji je

kompatibilan sa njim ali je sigurniji i rešava sve pomenute probleme. U skorije vreme zamena bi mogla biti protokol Diameter (za razliku od RADIUS-a, koristi usluge TCP protokola na transportnom sloju). Diameter je još uvek u fazi razvoja.

6. IPsec

IPsec. Saobraćaj između samih lokacija će biti zaštićen IPsec tehnologijom. U trenutnim poslovnim okruženjima, kritično je da korporativne mreže, konektovane na globalne mreže, imaju maksimalno zaštićen saobraćaj. IPsec, kao framework, nudi maksimalnu zaštitu podataka, kao vid kreiranja bezbednih virtuelnih privatnih mreža. Dizajniran je da pruži kriptološku bezbednost visokog kvaliteta i performansi. IPsec je skup protokola koji služe za autentifikaciju i enkripciju IP paketa u dvosmernoj komunikativnoj sesiji. Bezbednost IP saobraćaja je omogućena tako što sam sistem bira skup unapred predefinisanih bezbednosnih protokola i polisa, koji posle obavljaju složene matematičke funkcije koje obezbeđuju selektovan saobraćaj. IPsec može biti upotrebljen da se zaštiti jedan ili više kanala komunikacije između para hostova, para bezbednosih uređaja ili para bezbednosnih uređaja i hosta. U bezbednosne uređaje spadaju razne vrste firewall-ova, VPN koncentratora i svih ostalih uređaja čija je funkcija da omoguće mrežnu i sistemsku bezbednost. Jednostavno rečeno, IPsec omogućava sigurne tunele između dva hosta (u ovom slučaju dva rutera). Definiše se koji paketi se smatraju od kritične važnosti za zaštitu i koji bi trebalo da se pošalju kroz sigurne tunele, i mrežni administrator definiše parametre kojim bi paketi trebalo biti zaštićeni. Onda, kada IPsec host primeti senzitivni paket, konstruiše sigurni kanal komunikacije, tunel, i šalje tako zaštićen paket kroz taj tunel. Ti tuneli se isto tako nazivaju bezbednosne asocijacije (SA – Security Association). SA definiše koji protokoli i algoritmi trebaju biti primenjeni na senzitivne pakete, kao što definišu sistem razmene ključeva kao i same ključeve. SA je skup algoritama i parametara koji moraju da budu isti na obe krajnje tačke tunela. Izbor samih parametara, mehanizama i algoritama je na IPsec administratorima. U cilju odluke koja i koji nivo protekcije se pridodaje selektovanom paketu, IPsec koristi SPI parameter (Security Parameter Index), index koji je pokazivač na SADB (Security Association Database), koji zajedno sa destinacionom IP adresom paketom, kreira jedinstvenu bezbednosnu asocijaciju za taj paket. IPsec u svojoj osnovi koristi tri protokola AH, ESP i IKE.

AH (Authentication Header) se koristi da se omogući connectionless integritet i autentifikacija izvora paketa. AH takodje pruža zaštitu od replay napada. AH je definisan kroz RFC 2402. AH je mehanizam koji omogućava jak integritet i jaku autentifikaciju IP datagrama. AH ne omogućava enkripciju, tako da korisnici koji žele da kriptuju saobraćaj, moraju da se odluče za ESP.

ESP (Encapsulating Security Payload), pored mogućnosti AH, daje mogućnost i enkripcije samih paketa. Treći protokol u okviru IPsec-a je IKE (Internet Key Exchange). IKE je hibridan

protokol koji uzima deo Oakley-a i deo drugog protokol skupa koji se zove Skeme unutar ISAKMP (Internet Security Association and Key Management Protocol) framework-a. IKE se koristi da bi se uspostavila deljena bezbednosna polisa i razmena samih ključeva. IKE radi u dve faze. U fazi jedan dva ISAKMP hosta uspostavljaju siguran, autentifikovan kanal kojim komuniciraju koji se naziva ISAKMP SA. IKE faza dva je zadužena sa samu razmenu ključeva.

U projektu će biti implementiran IPsec između svake od tri lokacije upotrebom predefinisanih ključeva.

Prvo je potrebno uključiti ISAKMP kao protokol na ruteru.

```
Cisco2911(config)#crypto isakmp enable
```

Sledeći korak će biti definisanje ISAKMP polise. U okviru jedne polise, može se definisati više parametarnih mapa za definisanje same ISAKMP SA. ISAKMP polisa se definiše na sledeći način:

```
Cisco2911(config)#crypto isakmp policy 110
Cisco2911(config-isakmp)#encryption des|3des|aes
Cisco2911(config-isakmp)#hash md5|sha
Cisco2911(config-isakmp)#authentication pre-share
Cisco2911(config-isakmp)#group 1|2|5
Cisco2901(config-isakmp)#lifetime 36000
```

Sledeće što je potrebno je konfigurisati predefinisani ključ. Definiše se ime ključa i adresa hosta sa kojim se uspostavlja lokacija. Predefinisani ključ se koristi da se identifikuje i autentifikuje IPsec tunel. Ključ može biti bilo koji string.

```
Cisco2901(config)#crypto isakmp key cisco1234 address 10.12.0.2
```

Dalje se konfiguriše IPsec polisa koja se još i naziva transform-set. Transform-set predstavlja određenu kombinaciju bezbednosnih protokola, algoritama i ostalih opcija za IPsec obezbeđivanje saobraćaja. Transform-set je u stvari skup protokola (AES,MD5,SHA...) koji štite realan saobraćaj. Tokom IPsec SA pregovaranja, hostovi se dogovore o određenom transform-setu koji će da koriste za određeni tok podataka. Moguće je definisati više transform-setova i specificirati jedan ili više transform-set u crypto mapi (mapa koja se koristi

za objedinjavanje kroz konfiguraciju). Da bi se uspostavila pravilna security polisa potrebno je da bar jedna polisa u okviru transform seta bude ista na peer-ovima.
Primer konfigurisanog t-seta :

```
Cisco2911(config)#crypto ipsec transform-set CISCO esp-des esp-md5-hmac
```

Definisanje globalnog perioda vazenja IPsec SA:

```
Cisco2911(config)#crypto ipsec security-association lifetime seconds 36000
```

Potrebno je definisati interesantan saobraćaj koji se određuje access-listom. Interesantan saobraćaj je onaj saobraćaj koji treba biti zaštićen kroz tunele. Poslednje što treba uraditi je kreirati crypto mape i iskonfigurisati ih na virtuelne GRE interfejsse. U okviru crypto mape nalaze se sledeći parametri : interesantan saobraćaj, adresa IPsec peer-a, lokalna adresa koja će biti iskorišćena za IPsec saobraćaj, koji transform-setovi trebaju da se iskoriste za taj saobraćaj, ostali parametri koji mogu biti od manje ili veće vaznosti. Primer.

Definicija interesantnog saobraćaja:

```
Access list 110 permit ip 10.1.0.0 0.0.255.255 10.2.0.0 0.0.255.255

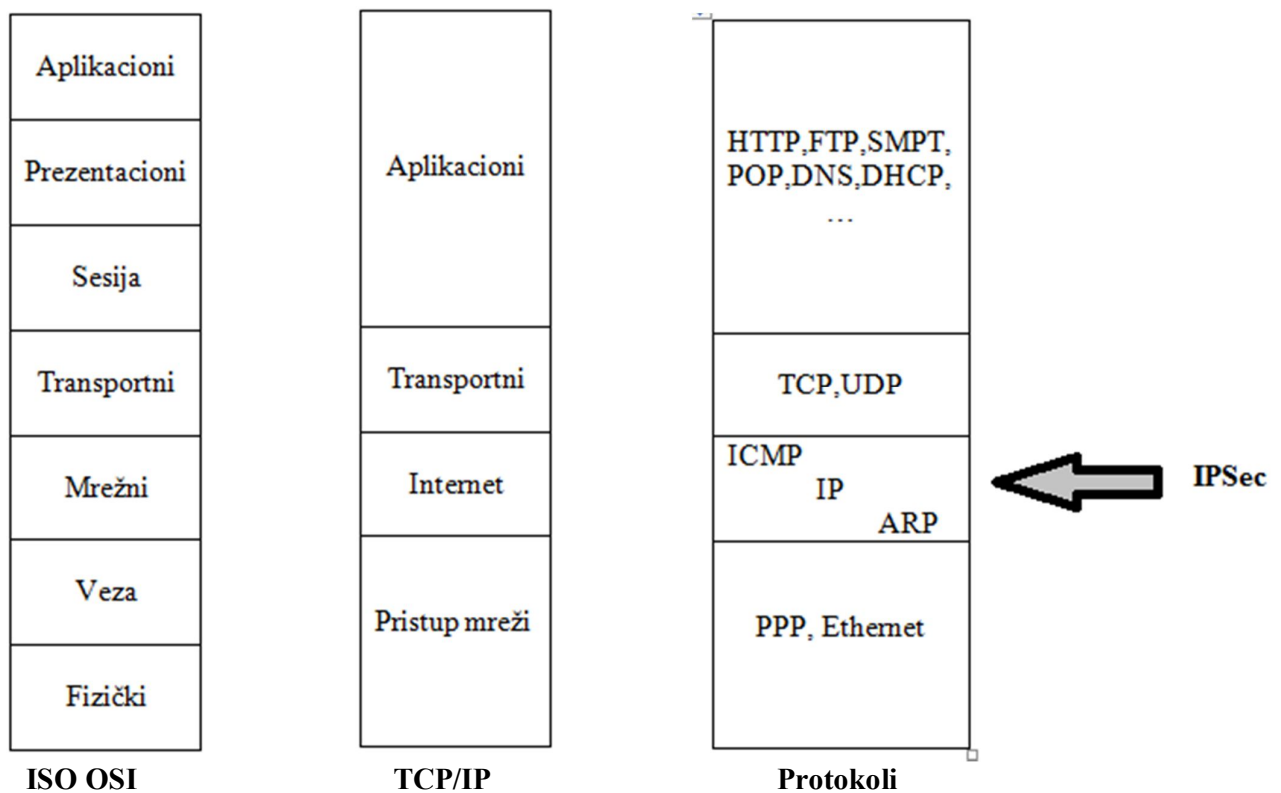
Cisco2911(config)#crypto map MYMAP 110 ipsec-isakmp
Cisco2911(config-crypto-map)#match address 110
Cisco2911(config-crypto-map)#set peer 10.12.0.2
Cisco2911(config-crypto-map)#set transform-set CISCO
Cisco2911(config-crypto-map)#set security-accociation lifetime seconds 36000
```

Zadnja stavka zadatka je povezati mapu sa virtuelnim interfejsom:

```
Cisco2911(config)#interface tunnel 12
Cisco2911(config-if)#crypto map MYMAP
```

TCP/IP je skup protokola koji je prihvaćen kao *de facto* standard za mrežnu komunikaciju u većini današnjih računarskih mreža. Internet, kao globalna mreža, zasnovan je na skupu

protokola TCP/IP, pri čemu je IPv4 protokol Internet sloja. U verziji 6 protokola IP (IPv6) ispravljani su neki nedostaci protokola IPv4. Jedan od osnovnih nedostataka skupa protokola TCP/IP jeste nepostojanje zaštitnih mehanizama kojima bi se osigurao integritet podataka koji se prenose mrežom i proveri identitet učesnika u komunikaciji. IPSec- skup proširenja protokola IPv4 koji obezbeđuje osnovne sigurnosne aspekte mrežne komunikacije su privatnost, integritet, provera identiteta i neporecivost. IPSec pored toga što je proširenje protokola IPv4 je integralni deo protokola IPv6. Pošto se integriše sa IP protokolom, IPSec implementira sigurnosne mehanizme mrežne komunikacije na mrežnom sloju OSI referentnog modela, odnosno na Internet sloju skupa protokola TCP/IP, slika 6.1.



Sl. 6.1. Mesto protokola IPSec u skupu protokola

IP protokol obezbeđuje komunikacioni kanal sa kraja na kraj i ne zavisi od nižih slojeva. IPSec se može koristiti bez obzira na način implementacije fizičkog sloja i sloja veze. Komunikacioni uređaji na putu između dva entiteta ne moraju podržavati IPSec. Protokoli transportnog sloja koriste sigurnosne usluge koje obezbeđuju Ipsec, što znači da svi podaci koji se prenose posredstvom TCP i UDP protokola, kao i ICMP poruke, mogu koristiti sigurni komunikacioni kanal koji obezbeđuje IPSec. Upotreba IPSec protokola transparentna je (nevidljiva) za više slojeve skupa protokola TCP/IP. To znači da aplikacije koriste ove usluge bez obzira na svoju funkcionalnost. IPSec definiše informacije koje se moraju dodati IP paketu kako bi se obezbedili privatnost, integritet, provera identiteta i način šifrovanja sadržaja paketa. Pri radu, IPSec koristi sledeće protokole i standarde:

- Diffie-Hellmanov protokol za razmenu ključeva između dva učesnika u komunikaciji,
- algoritme za digitalno potpisivanje komunikacije pri Diffie-Hellmanovoj razmeni ključeva kako bi se potvrdio identitet oba učesnika u komunikaciji i izbegla mogućnost napada tipa čovek u sredini.
- AES, 3DES i DES simetrične algoritme za šifrovanje
- HMAC (*Hashing Message Authentication*) u sprezi sa algoritmima MD5 i SHA,
- digitalne sertifikate koje je potpisao odgovarajući autoritet.

IPSec podržava dva režima rada: prenosni (*transport mode*) i tunelovanje (*tunnel mode*).

- U prenosnom režimu rada šifruju se samo podaci tj. punjenje IP paketa, dok IP zaglavlja ostaju u originalnom obliku (otvoreni tekst). Zaglavlja viših slojeva (npr, sloja aplikacije) šifrovane su, a mogućnost pregledanja paketa je ograničena. Prednost ovog režima rada je to što se svakom paketu dodaje svega nekoliko okteta. U ovom načinu rada, ruteri mogu na javnoj mreži videti adrese izvorišta i odredišta poruka, što potencijalnom napadaču delimično omogućava da analizira mrežni saobraćaj.
- Drugi režim rada IPSec je IP tunelovanje, pri kome se koristi poseban oblik IP paketa. Tunel čine klijent i server koji su konfigurisani da koriste IPSec tunelovanje i unapred dogovorene mehanizme za enkapsulaciju i šifrovanje kompletnih IP paketa, što obezbeđuje potpuno siguran prenos preko javnih ili privatnih mreža. Šifrovani podaci se spajaju sa odgovarajućim nešifrovanim IP zaglavlјima, formirajući tako IP pakete koji se na kraju tunela dešifruju i obliku u IP pakete namenjen krajnjem odredištu.

6.1. IPSec protokoli

IPSec se implementira pomoću dva međusobno nesavisna protokola. AH (*Authentication Header*) obezbeđuje usluge integriteta, provere identita i neporecivost, dok ESP (*Encapsulated Security Payload*) osim toga obezbeđuje i privatnost podataka koji se prenose. Oba protokola AH i ESP, modifikuju standardni oblik IP paketa ko je prikazan na slici 6.2.

IP zaglavlje	TCP zaglavlje	Podaci
--------------	---------------	--------

SI.6.2. Standardni oblik IP paketa

6.1.1. Protokol AH

Protokol AH Definisan je u dokumentu RFC 2402. AH obezbeđuje sigurnosne usluge provere identiteta, integriteta i neporecivosti IP paketa, ali ne može obezbediti privatnost. Protokol je definisano AH zaglavlje koje se smešta između IP zaglavlja i podataka koji slede. Specifičnost protokola AH je to što on, za razliku od ostalih TCP/IP protokola, ne enkapsulira podatke iz protokola kojima pruža uslugu. Na slici 6.3. prikazano je zaglavlje; sva polja ovog zaglavlja su obavezna.

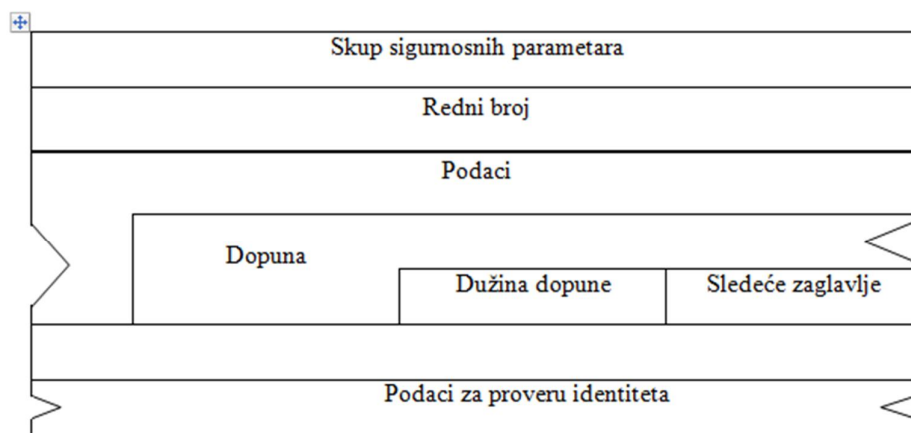
Sledeće zaglavlje	Dužina punjenja	Rezervisano
Skup sigurnosnih parametara		
Redni broj		
Podaci za proveru identiteta		

SI.6.3. AH zaglavlje

- Sledeće zaglavlje (next header). Osmobitno polje za identifikaciju tipa podataka koji slede nakon AH zaglavlja. Polje sadrži vrednost koja označava IP protokole (npr, 6-TCP, 17-UDP, 50-ESP).
- Dužina punjenja (*Payload Length*). Dužina punjenja izražena brojem 32-bitnih reči, umanjenim za vrednost 2.
- Rezervisano (*Reserved*). Polje dužine 16 bitova rezervisano za buduće potrebe postavlja se na vrednost 0.
- Skup sigurnosnih parametara (*Security Parameters Index*). Ovo polje dužine 32 bita sadrži proizvoljnu vrednost, koja uz IP adresu i sigurnosni protokol (u ovom slučaju AH) definiše jedinstven skup sigurnosnih parametara (Security Association- SA) koji se koristi u sigurnoj komunikaciji između dva entiteta. ESA skup sigurnosnih parametara definiše se prilikom uspostavljanja IPSec veze. Vrednost iz intervala 1-255 rezervisala je IANA za buduću upotrebu..
- Redni broj (*sequence number*). Polje dužine 32 bita koje služi kao zaštita od napada ponavljanjem paketa. Povećava se prilikom svakog slanja paketa koji ima identičan SA skup sigurnosnih parametara. Pošiljalac mora da generiše ovo polje, a primalac može, ali ne mora da ga interpretira. Na početku komunikacije ovo polje se postavlja na vrednost 1.
- Podaci za proveru identiteta (*Authentication Data*). U polju koje sadrži podatke za proveru identiteta nalazi se ICV vrednost (*Integrity Check Value*) na osnovu koje se proverava integritet i autentičnost poruke. Dužina tog polja je promenljiva ali mora biti celobrojni umnožak 32-bitne reči. Ukoliko polje samo po sebi ne ispunjava taj uslov proizvoljnim nizom bitova dopunjava se do dužine $n \times 32$ bita. Vrednost ICV-a se računa na osnovu svih polja IP zaglavlja koja se ne menjaju prilikom prenosa, čitavog AH zaglavlja (koje je za tu potrebu postavljeno na vrednost 0), i svih podataka protokola višeg sloja. ICV može biti kod (*code*) za proveru identiteta poruke (*Message Authentication Code*), izračunat pomoću simetričnih algoritama za šifrovanje ili rezultat heš funkcija. Algoritam koji se upotrebljava za računanje ICV-a definiše se prilikom uspostavljanja komunikacije i deo je SA skupa sigurnosnih parametara.

6.1.2. Protokol ESP

Protokol ESP definisan je u dokumentu RFC 2406. ESP obezbeđuje sigurnosne usluge provere identiteta, integriteta, neporecivosti i privatnosti podataka. Protokol definiše ESP zaglavlje koje se u IP paket smešta posle IP zaglavlja, enkapsulira sve podatke iz protokola višeg sloja i dodaje završni slog u koji se mogu smestiti podaci za proveru identiteta. Na slici 6.4. prikazan je ESP paket sa odgovarajućim poljima u zaglavlju.



SI. 6.4. ESP paket

- Skup sigurnosnih parametara (*Security Parameters Index*). Polje dužine 32 bita u kome se, isto kao i u zaglavlju protokola AH, definiše jedinstven SA skup sigurnosnih parametara (određene prilikom uspostavljanja komunikacije) koji se koristi u komunikaciji između dva entiteta. Kao i u zaglavlju protokola AH, vrednost od 1 do 255 rezervisane su za buduću upotrebu.
- Redni broj (*Sequence Number*). Ovo polje dužine 32 bita služi, kao i u AH zaglavlju, za zaštitu od našada ponavljanjem paketa, a povećava se prilikom svakog stanja paketa koji ima identičan SA skup sigurnosnih parametara. Pošiljalac mora da generiše ovo polje, a primalac može, ali ne mora da ga interpretira. Na početku komunikacije ovo polje se postavlja na vrednost 0 (Što se razlikuje od AH zaglavlja u kome se koristi inicijalna vrednost 1).
- Podaci i dopuna (*Payload Data*). Ovo polje proizvoljne dužine sadrži podatke IP paketa i dopunu. U polju mogu biti smešteni i podaci koji su nužni za kriptografsku sinhronizaciju (kao što je inicijalizacioni vektor- IV), ukoliko to zahteva kriptografski algoritam koji se koristi. Dopuna se koristi iz dva razloga, prvi je da neki algoritmi šifruju blokove fiksne dužine, što znači da deo ESP paketa u kome su smešteni podaci treba dopuniti do odgovarajuće dužine, a drugi je da razlozi implementacije- ukupna

dužina polja podaci, dopuna, dužina dopune i sledeće zaglavlje treba da bude celobrojni umnožak 32-bitne reči.

- Dužina dopune (*Padding Length*). Ovo 8-bitno polje određuje dužinu (izraženu u broju okteta) predhodno korišćene dopune. Dozvoljene vrednosti su od 0 do 255; vrednost 0 označava da dopuna ne postoji.
- Sledeće zaglavlje (*Next Header*). Sledeće zaglavlje je kao i u protokolu AH 8-bitno polje koje identifikuje tip podataka koji sledi posle ESP zaglavlja. Polje sadrži vrednosti iz definisanog skupa brojeva koji označavaju IP protokole.
- Podaci za proveru identiteta (*Authentication Data*). Ovo polje proizvoljne dužine nije obavezno, a koristi se samo kada je u SA skupu sigurnosnih parametara specificirana usluga provere identiteta. U tom slučaju, ovo polje sadrži ICV koji se izračunava za ceo ESP paket (ESP zaglavlje, podaci i dopuna) izuzev polja namenjenog podacima za proveru identiteta. Dužina ovog polja zavisi od primenjenog algoritma za proveru identiteta.

6.2. Režimi rada

IPSec definiše dva osnovna režima rada: transportni režim i tunelovanje. Oba protokola, AH i ESP, mogu se koristiti u transportnom režimu ili za tunelovanje. Ukoliko je potrebno dodatno podići nivo sigurnosti, može se koristiti i kombinacija oba protokola.

6.2.1. Transportni režim rada

Transportni režim rada namenjen je prvenstveno za uspostavljanje sigurne komunikacije između dva entiteta, tj. za komunikaciju između računara u privatnim LAN ili WAN mrežama. U transportnom načinu rada potrebno je da obe krajnje tačke komunikacije (izvor i odredište) podržavaju IPSec. Korišćenjem protokola AH i ESP mogu se postići različiti aspekti sigurne komunikacije.

- Protokol AH. AH zaglavlje se dodaje odmah iza IP zaglavlja (slika 6.5.) . U tom slučaju polje “protokol” u IP zaglavlju sadrži vrednost 51 (AH), dok polje “sledeće zaglavlje” u AH zaglavlju sadrži vrednost koja odgovara enkapsuliranom datagramu iz višeg sloja (npr. 6 za TCP segment). Kao što se na osnovu slike može zapaziti AH u transportnom režimu obezbeđuje proveru identiteta, integriteta i neporecivost celog IP paketa.

IP Zaglavlje	IPSec AH zaglavlje	TCP zaglavlje	podaci
-----------------	-----------------------	------------------	--------

proveren identitet



Sl. 6.5. AH u transportnom režimu rada

- Protokol ESP. U transportnom režimu rada ESP osigurava integritet, proveru identiteta, neporecivost i privatnost podataka koji se prenose. Ukoliko se za IPSec koristi ESP, polje “protokol” u IP zaglavlju sadržiće vrednost 50 (ESP), a polje ”sledeće zaglavlje” - vrednost koja odgovara enkapsuliranim podacima iz višeg sloja, isto kao u protokolu AH. Iza enkapsuliranih podataka ESP dodaje dopunu, a opciono (ukoliko je u SA skupu sigurnosnih parametara specificirana i provera identiteta) polje “podaci za proveru identiteta”. Na slici 6.6. prikazan je ESP u transportnom režimu rada.

IP zaglavlje	IPSec ESP zaglavlje	TCP zaglavlje	podaci	ESP dopuna	ESP aut. podaci
-----------------	------------------------	------------------	--------	---------------	--------------------

šifrovano



proveren identitet

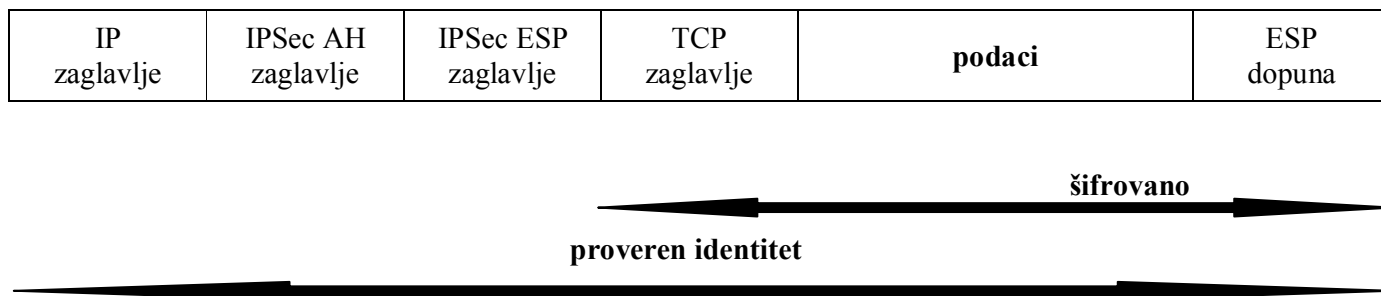


Sl. 6.6. ESP u transportnom režimu rada

Kao što se vidi sa slike, svi podaci iz višeg sloja (uključujući i ESP dopunu) šifrovani su. Takođe, može se uočiti da za razliku od protokola AH, koji proverava identitet celog IP paketa (uključujući i IP zaglavlje) - ESP proverava identiteta vlastitog zaglavlja i podataka, ali

ne i identitet IP zaglavlja, čime je teorijski ostavljena mogućnost neovlašćene izmene IP zaglavlja.

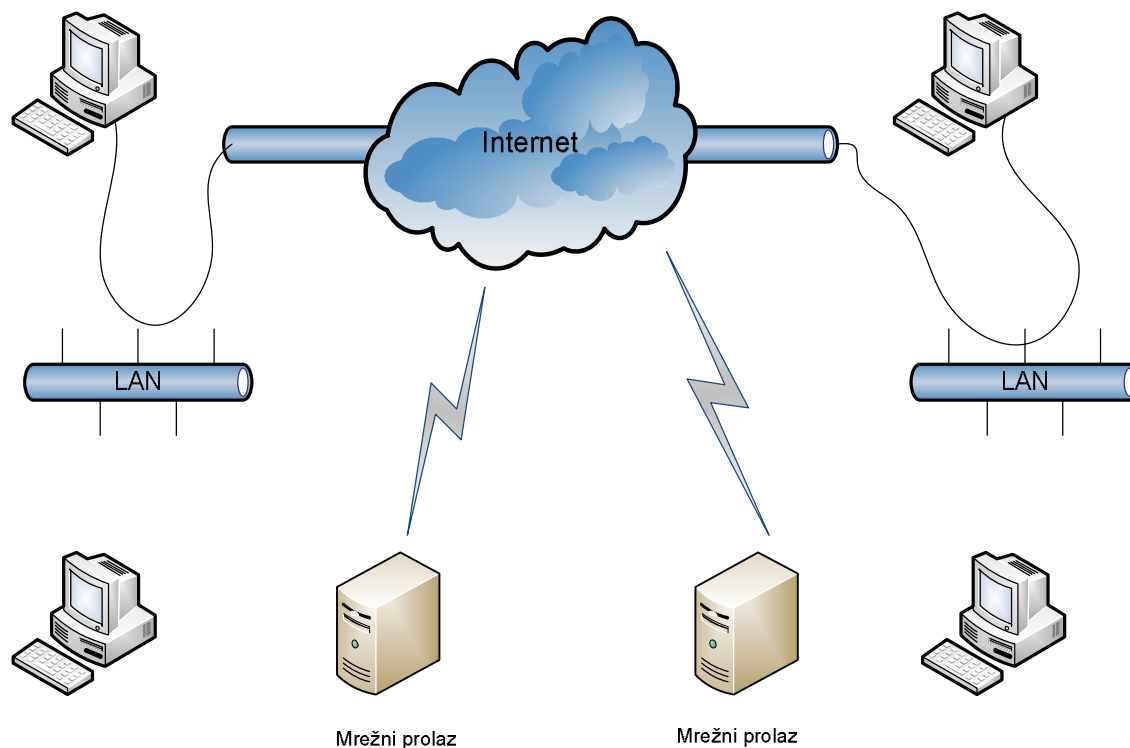
- ESP + AH. AH obezbeđuje integritet, proveru identiteta i neporecivost celog IP paketa, a ESP privatnost podataka, i opciono integritet, proveru identiteta i neporecivost podataka i ESP zaglavlja. Ukoliko je potrebno da se dostigne veoma visok nivo zaštite, to jest da se osigura privatnost podataka i proveru identiteta, integritet i neporecivost celog IP paketa, koriste se ESP i AH zajedno. U tom slučaju polje “protokol” u IP zaglavlju sadržaće vrednost 51 (AH). Nakon toga slede: AH zaglavlje, čije polje “sledeće zaglavlje” sadrži vrednost 50 (ESP) i ESP zaglavlje čije polje “sledeće zaglavlje” sadrži vrednost koja označava onaj protokol višeg sloja iz kog su podaci enkapsulirani u tako formiranom paketu. 6.7. prikazuje IPSec u transportnom režimu rada kada se istovremeno koriste ESP i AH. Treba napomenuti da se prvo formira ESP deo paketa, to jest šifruju se podaci sa transportnog sloja i formira odgovarajuće ESP zaglavlje. Posle toga se računa vrednost AH zaglavlja i formira se zaglavlje. U ovom slučaju, ESP deo paketa ne sadrži opciono polje “podaci za proveru identiteta”, pošto proveru identiteta, integritet i neporecivost obezbeđuje AH.



Sl. 6.7. ESP + AH u transportnom režimu rada

6.2.2. Tunelovanje

Tunelovanje je drugi režim rada IPSec protokola, u kome IPSec služi za uspostavljanje sigurne komunikacije između mrežnih prolaza (*Gateway*) na udaljenim mrežama (*Gateway-to-Gateway*), obezbeđujući tako virtuelnu privatnu komunikaciju, to jest uspostavljajući VPN mrežu između udaljenih lokacija. U ovom slučaju krajnji entiteti u komunikaciji ne moraju da podržavaju IPSec. Za njih je čitava komunikacija transparentna (nevidljiva) jer se sve operacije neophodne za sigurnu komunikaciju obavljaju u mrežnim prolazima. Mrežni prolazi na udaljenim mrežama predstvljaju ulaznu, odnosno izlaznu tačku sigurnog komunikacionog kanala. Oni formiraju siguran tunel kroz nesiguran medijum (internet) - zbog toga se taj način rada i zove tunelovanje slika (6.8.).



SI.6.8. Tunelovanje

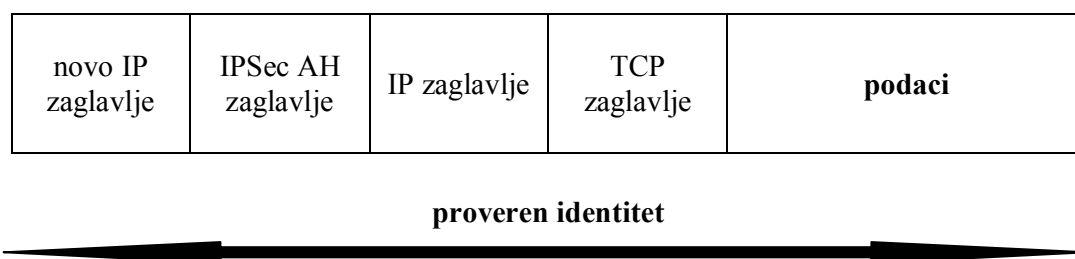
Tunelski način rada moguće je i u komunikaciji računar-računar ili računar-mrežni prolaz, ali tada krajnji entiteti (odnosno entitet) moraju podržavati IPSec.

Za razliku od transparentnog načina rada, gde se AH, odnosno ESP zaglavlja dodaju u postojeći IP paket, pri tunelovanju se formira potpuno nov IP paket koji enkapsulira kompletan originalni IP paket. Dva entiteta komuniciraju na sledeći način:

1. Pošiljalac formira IP paket i šalje ga preko lokalne mreže lokalnom mrežnom prolazu.
2. Mrežni prolaz enkapsulira originalni IP paket nov paket i formira odgovarajuća AH, odnosno ESP zaglavlja.
3. Tako formirani paket šalje se preko uspostavljenog tunela do mrežnog prolaza na udaljenoj mreži, koji uklanja dodatna zaglavlja, i po potrebi dešifruje paket i proverava njegov integritet.
4. Nakon toga se originalni IP paket isporučuje odredištu.

Kao i u transportnom načinu rada, moguća je i implementacija IPSec-a korišćenjem protokola AH i ESP.

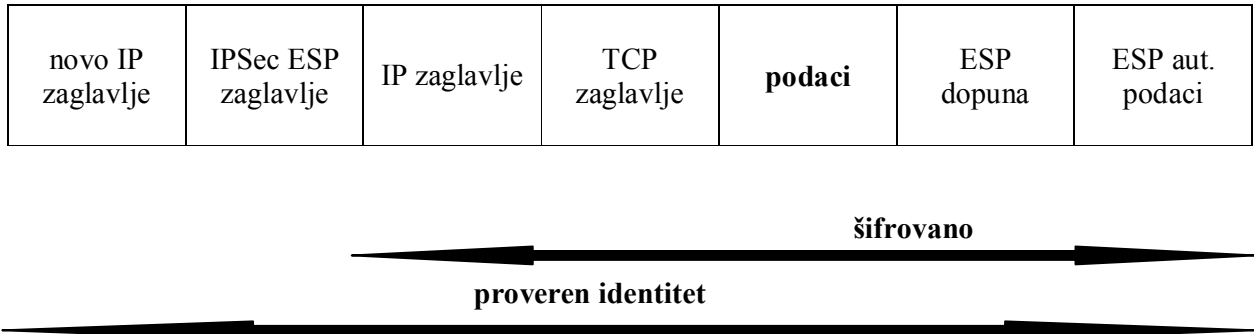
- AH. Ukoliko je potrebno da se obezbede samo integritet, provera identiteta i neporecivost poruka, a privatnost nije nužna, koristi se protokol AH. U tom slučaju originalni IP paket, koji sadrži adresu krajnjeg odredišta, enkapsulira se u nov IP paket kojem se dodaje odgovarajuće AH zaglavlje (slika 6.9.) Tada polje "protokol" novog IP zaglavlja, koje sadrži adresu krajnje tačke IPSec tunela, ima vrednost 51 (AH), dok polje "sledeće zaglavlje" iz AH zaglavlja ima vrednost 4 (IP paket enkapsuliran u IP paket).



Sl.6.9. AH tunelovanje

- ESP. Ako treba obezbediti i privatnost komunikacije, koristi se protokol ESP. U tunelskom režimu rada, za razliku od transportnog ESP šifruje i obezbeđuje proveru identiteta (slika 4.10.) , integritet i neporecivost celog originalnog IP paketa, pošto je sam paket enkapsuliran u novi IP paket. U ovom slučaju, polje "protokol" novog Ip

zaglavlja koje, isto kao i u AH zaglavlju, sadrži gde su krajnje tačke IPSec tunela, ima vrednost 50 (ESP), dok polje "sledeće zaglavlje" u ESP zaglavlju ima vrednost 4(IP paket enkapsuliran u IP paket).



Sl.6.10. ESP tunelovanje

- AH+ESP. Kombinacija AH i ESP u tunelskom režimu rada nije predviđena (RFC dokument 2401).

6.3. Uspostavljanje IPSec komunikacije

IPSec ne sadrži mehanizam za uspostavljanje komunikacije i ne specificira konkretne kriptografske algoritme koji će se koristiti u IPSec komunikaciji. Za korišćenje bilo kakvih kriptografskih metoda nužno je da se entiteti koji žele da komuniciraju pomoću IPSec protokola dogovore o skupu sigurnosnih parametara komunikacije SA. Taj skup sigurnosnih parametara obuhvata dogovor o kriptografskim metodama koje će se koristiti, način provere identiteta strana u komunikaciji i razmeni kriptografskih ključeva potrebnih za tako dogovorenu komunikaciju. Postoji nekoliko načina za uspostavljanje IPSec komunikacije. Teoretski je moguće ručno podešavanje skupa sigurnosnih parametara, ali to nije prihvatljivo za bilo kakvu ozbiljniju primenu. Postoji nekoliko formalnih metoda koje se već koriste ili su predložene za uspostavljanje IPSec komunikacije. Protokoli *Photuris* i *SKIP (Simple Key Management For Internet Protocols)* zasnovani su na Diffie-Hellmanovom protokolu za razmenu ključeva i mogu se koristiti u tu svrhu. Kao opšteprihvaćene metode za uspostavljanje IPSec komunikacije koriste se *ISAKMP (Internet Security Association And Key Management Protocol)* i *IKE (Internet Key Exchange)*.

6.3.1. IKE

IKE je standardni protokol za uspostavljanje sigurne IPsec komunikacije, definisan u dokumentu RFC 2409. Protokol je implementiran kombinovanjem nekoliko postojećih protokola: ISAKMP, Oakley i SKEME.

Uspostavljanje IPsec komunikacije primenom protokola IKE sastoji se od dve osnovne faze: uspostavljanjem IKE SA skupa sigurnosnih parametara i uspostavljanja IPsec SA skupa sigurnosnih parametara korišćenjem IKE SA. Uobičajeno je da se za IKE komunikaciju koristi UDP priključak (port) 500.

- Uspostavljanje IKE SA. Osnovna funkcija IKE SA skupa sigurnosnih parametara jeste osiguravanje provere identiteta i sigurnosti IKE saobraćaja, a unutar tako uspostavljene komunikacije može se definisati više IPsec SA. Uspostavljeni IKE SA mora da sadrži sledeće attribute: algoritam za šifrovanje, heš funkciju, metodu provere identiteta i Oakley grupu koja definiše Diffie-Hellmanovu razmenu ključeva (RSA, eliptičke krive).

Metoda provere identiteta je način provere identiteta entiteta u predstojećoj komunikaciji. IKE podržava sledeće metode provere identiteta: Korišćenje digitalnih potpisa (RSA ili DSS), korišćenje tajnog ključa (*preshared key*) i korišćenje kriptografije sa javnim ključevima. Metoda provere identiteta, zasnovana na tajnom ključu, ostljiva je na napade tipa "čovek u sredini", što je inherentno svojstvo Diffie-Hellmanovog protokola za razmenu ključeva. Za sigurnu proveru identiteta preporučuje se korišćenje infrastrukture javnih ključeva.

Za uspostavljanje IKE SA koriste se dva načina: glavni način (*Main Mode*) i agresivni način (*Aggressive Mode*). Glavni način se koristi kada je neophodna zaštita identiteta entiteta u komunikaciji. U glavnom načinu rada, entiteti razmenjuju 6 poruka kako bi uspostavili IKE SA. Agresivni način se može upotrebiti kada nije neophodna zaštita identiteta, ali je poželjno što brže uspostavljanje komunikacije. Pri ovom načinu rada, entiteti razmenjuju samo tri poruke. Potrebno je napomenuti da će, ukoliko se u agresivnom načinu rada koristi kriptografija sa javnim ključevima, zaštita identiteta takođe biti osigurana.

Za održavanje IKE komunikacije potrebno je generisati četiri različita ključa: glavni ključ koji se koristi za generisanje ostalih ključeva, ključ koji IKE SA koristi za šifrovanje poruka, ključ koji IKE SA koristi da obezbedi integritet i proveru identiteta poruka i ključ koji služi za generisanje IPsec SA. Pri generisanju ključeva koriste se i *cookies* koje generišu entiteti u komunikaciji i koji predstvljaju heš vrednosti izračunate na osnovu identifikatora (IP adresa entiteta, port, protokol), vremenske oznake i tajne vrednosti poznate samo entitetu koji je generisao *cookies*

- Uspostavljanje IPsec SA. Druga faza protokola IKE služi za uspostavljanje IPsec SA skupa sigurnosnih parametara. Ova faza se izvodi u takozvanom brzom načinu rada (*Quick Mode*). Cela komunikacija koje se odvija u drugoj fazi zaštićena je jer se koristi predhodno uspostavljen IKE SA skup sigurnosnih parametara. Ova faza u

stvari nije zasebna, već se koristi samo za generisanje IPSec SA na temelju ranije uspostavljenog IKE SA.

Nakon druge faze protokola IKE, dva entiteta u komunikaciji definisala su IPSec SA skup sigurnosnih parametara, i mogu uspostaviti siguran kanal za razmenu poruka.

6.4. Resursi koje IPSec zahteva i problemi u implementaciji

Zbog kriptografskih operacija koje su matematički zahtevne, za korišćenje IPSec-a potrebni su dodatni procesorski resursi. Osim toga, IPSec povećava ukupan mrežni saobraćaj, što je po sebi razumljivo ukoliko se IPSec paketi uporede sa standardnim IP paketima. Postoje dva razloga za povećanje mrežnog saobraćaja, to jeste premašenje (*overhead*) koji IPSec unosi (koje može rezultovati degradacijom performansi mreže). To su :

- Dodatna zaglavlja koja se mogu pojaviti u različitim načinima IPSec rada,
- Dopuna (*Padding*) koja je neophodna za ispravno funkcionisanje primenjenih kriptografskih algoritama.

Premašenje zaglavlja zavisi od načina rada IPSec-a, kao i od IPSec protokola koji se koristi. Upotreba AH protokola (ukoliko se koriste propisane heš funkcije MD5 ili SHA1) unosi premašenje od 24 okteta, od kojih 12 otpada na zaglavlja bez polja “podaci za proveru identiteta”, dok preostalih 12 okteta (96 bitova) otpada na polje sa ICV vrednošću koju je generisala heš funkcija. Treba napomenuti sledeće: iako MD5 daje izlazni rezultat dužine 128 bitova, a SHA1 160 bitova, te vrednosti se za potrebe IPSece svode na 96 bitova.

Ukoliko se koristi protokol ESP, premašenje zavisi od toga da li se ESP upotrebljava samo za obezbeđivanje privatnosti ili služi za obezbeđivanje integriteta, neporecivosti i proveru identiteta prouke. Premašenje zavisi i od primljenog kriptografskog protokola. Esp zaglavlje samo po sebi dodaje 8 okteta. Dalje, za kriptografske algoritme u režimu rada CBC neophodan je inicijalizacioni vektor čija dužina može biti do 16 okteta (8 okteta za DES i 3DES ili 16 okteta za AES). Tu su još četiri okteta koji se odnose na polja “dužina dopune” i “sledeće zaglavlje” (dva okteta za polja i dva za dopunjavanje do 32-bitne reči), a ukoliko se ESP koristi za osiguranje integriteta, neporecivosti i provere identiteta, potrebno je dodati i 12 okteta i za ICV prenos iz polja “podaci za proveru identiteta” na kraju ESP paketa. Ukoliko se IPSec koristi u tunnelskom načinu rada, treba dodati 20 okteta za novo IP zaglavlje.

Premašenje dopune zavisi i od IPSec protokola koji se koriste, to jeste direktno od algoritama za šifrovanje i heš funkcija koje su navedene u SA skupu sigurnosnih parametara. Dopuna je nužna zato što algoritmi za šifrovanje i heš funkcije kao ulaz koriste blokove fiksne dužine, a ta dužina zavisi od algoritama koji će se koristiti. Za kriptografske algoritme (DES, 3DES, AES) to konkretno znači da će svaki paket imati dopunu koja obezbeđuje da IP paket bude dužine $n \cdot 64$ bita, odnosno $n \cdot 128$ bitova. Pri upotrebi heš funkcija (MD5 i SHA1), zbog implementacijske specifičnosti, paket će biti dopunjen do umnoška od 448 bitova.

Razlog je to što oba algoritma ulaznim podacima implicitno dodaju 64 -bitni blok podataka, pa se skraćuje ulazni blok koji se može obraditi u heš funkciji. Premašenje ima veći uticaj na opadanje performansi mreže ukoliko se uglavnom šalju manji paketi.

IPSec protokol, sam po sebi, donosi neke probleme koje je ponekada teško ili nemoguće rešiti u specifičnim mrežnim okruženjima. To se prvenstveno odnosi na korišćenje NAT-a i IP fragmentaciju koja se može javiti prilikom IPSec komunikacije.

Ukoliko se bilo gde između entiteta koji žele da uspostave IPSec komunikaciju koristi NAT, ni u transportnom ni u tunelskom režimu rada ne može se upotrebiti AH, jer će NAT narušiti integritet IP paketa pa će ga primalac odbaciti.

Ukoliko se za IPSec koristi samo ESP, situacija nije bezizlazna. U transportnom načinu rada upotreba NAT-a takođe onemogućuje IPSec komunikaciju, ali u tunelskom režimu ESP može funkcionisati. Prilikom korišćenja NAT-a, treba obratiti pažnju i na IKE / ISAKMP, jer se pri proveru identiteta zasnovanoj na tajnom ključu koriste i *cookies* koji se generišu, zavisno od IP adrese entiteta tada se takođe gubi integritet i ne može se uspostaviti komunikacija. Ovaj nedostatak, za razliku od problema sa AH i ESP, može se otkloniti tako što će se primeniti druge IKE metode za proveru identiteta (korišćenje digitalnih potpisa ili kriptografije sa javnim ključevima).

IPSec NAT-T (NAT *Traversal*) tehnologija unapređuje IPSec tako da, uz određena ograničenja, omogućava uspostavljanje IPSec komunikacije i između entiteta koji se nalaze iza NAT uređaja. U tehnologiji IPSec NAT-T osnovna ideja je korišćenje UDP paketa za enkapsulaciju IPSec ESP i IKE paketa. IPSec AH paketi ne mogu se enkapsulirati korišćenjem NAT-T tehnologijom. Standardni port za IPSec NAT-T komunikaciju je UDP port 4500. Korišćenje istog UDP porta za NAT-T enkapsulaciju pojednostaljuje konkretnu implementaciju i konfiguraciju (npr. mrežnih barijera).

6.4.1. Konfiguracija NAT-a

NAT (Network Address Translation) je servis koji omogućava organizacijama da reše problem nedostatka dovoljno javnih IP adresa kako bi saobraćaj imao potpunu konverzaciju sa internetom i eksternim mrežama. NAT funkcioniše tako što izvorne privatne IP adrese pretvara u određene javne IP adrese (koje se zadaju konfiguracijom) i tako modifikovane pakete šalje na internet. U povratku saobraćaja, paketi su opet modifikovani. PAT (Port Address Translation) je oblik NATa koji dozvoljava da replikaciju više broja privatnih adresa bude preslikan u manji opseg javnih. Ovo se postiže uz pomoć TCP/UDP protokola i otvaranja više TCP/UDP portova radi prepoznavanje konekcija. Potrebno je definisati dva tipa NAT zone : outside i inside. Inside zona je okrenuta lokalnoj mreži, outside zone je okrenuta ka eksternoj mreži/internetu. Access-listom se definiše koji saobraćaj treba biti preveden. U koliko se javi potreba da korisnici sa interneta trebaju da direktno pristupe servisima u internoj mreži, na primer WEB serveru, konfiguriše se statički NAT. Statički NAT omogućava translaciju u 1:1 odnosu. Takodje se može postići relacija od više prema jedan, uz modifikaciju portova. Primer :

```
Cisco2911(config)#access list 10 permit 10.0.0.0 0.255.255.255
Cisco2911(config)#ip nat inside source list 10 interface serial 0/1 overload
Cisco2911(config)#ip nat inside source static 10.1.3.3 80 1.1.3.3 80
Cisco2911(config)#ip nat inside source static 10.1.3.3 25 1.1.3.3 25
Cisco2911(config)#interface s0/1
Cisco2911(config-if)#ip nat outside
Cisco2911(config)#interface fa0/1
Cisco2911(config-if)#ip nat inside
```

Core Dump. Core Dump sistem omogućava da uređaj usled otkaza pošalje presek sistema (memorije, procesora) u slučaju otkaza bilo koje vrste. Uz presek sistema, šalje se i razlog otkaza uređaja.

```
Cisco2911(config)#exception protocol ftp
Cisco2911(config)#exception dump 10.1.3.15
Cisco2911(config)#exception corefile dunjacoredump
```

7. Secure Shell (SSH)

Zaštita udaljenog pristupa uređajima vršiće se upotrebom SSH protokola. SSH je kriptovana verzija telnet protokola, koji omogućava podacima da budu zaštićeni bezbednosnim kanalima između dva uređaja. Koristi kriptografiju javnih ključeva da autentifikuje udaljene korisnike. SSH se obično koristi kao pandan telnet protokolu, mada može i da se koristi u druge svrhe. Standardni TCP port za SSH je 22. SSH u trenutnom postojanju podržava više verzija, a u ovom projektu biće korišćena verzija 2.0 .

```
Router2911(config)#ip domain-name dunjainc.com
Router2911(config)#crypto key generate rsa modulus 1024
Router2911(config)#ip ssh version 2
Router2911(config)#line vty 0 4
Router2911(config-line )#transport input ssh
Router2911(config-line)#transport output ssh
```

SSH je popularan protokol za šifrovanje komunikacionih kanala, koji se najčešće koristi za obezbeđivanje sigurnih sesija udaljenog prijavljivanja na sistem. Arhitektura protokola SSH je dvoslojna (two - tier) klijent/server arhitektura. SSH-server je softver koji prima ili odbija veze koje stižu računaru. SSH klijentski softver instaliran je na udeljenim računarima; klijenti šalju SSH serveru zahteve tipa “Molim da me prijavite na sistem”, “molim, pošaljite mi datoteku” ili “molim izvršite ovu komandu “. Pri tome, SSH šifrjuje sve podatke koji se prenose preko mreže, a samo šifrovanje je korisniku transparentno. SSH je protokol a ne proizvod; to je specifikacija koja predlaže jedan od načina za sigurnu mrežnu komunikaciju. Trenutno se koriste dve nekompatibilne verzije ovog protokola – verzija 1 i verzija 2. Protokol SSH obezbeđuje sigurnosne mehanizme provere identiteta, šifrovanja i obezbeđivanja integriteta podataka koji se šalju preko mreže.

Osnovne kriptografski zaštićene funkcije koje SSH obezbeđuje su:

- Daljinsko upravljanje na sistem. Prilikom prijavljivanja na udaljeni računar, telnet klijent preko mreže šalje korisničko ime i lozinku kao otvoren tekst, koji napadač može lako da presretne. Da situacija bude još gora, cela telnet sesija putuje preko mrežekao otvoren tekst, tako da napadač bez većeg napora može videti šta je neki korisnik radio. SSH rešava ovaj problem. SSH klijent će formirati šifrovan kanal preko koga će poslati vaše korisničko ime i lozinku udaljnog serveru. Nakon toga, identitet se proverava na osnovu podataka koji se čuvaju u bazi podataka na udaljenom računaru.
- Siguran prenos podataka. Tradicionalni programi za prenos podataka, kao što je ftp, nisu rešenje za siguran prenos datoteka preko mreže. Datoteke se doduše mogu

šifrovati u programu kao što je PGP i poslati na uobičajen način, a zatim dešifrovati na drugom računaru, ali takav način slanja nije transparentan za korisnika. Zahvaljujući protokolu SSH, datoteka se može preneti između dva računara preko šifrovanog kanala, primenom komande **scp** (*secure copy*). Prilikom slanja datoteka ovom komandom, datoteka se šifrjuje pre nego što napusti izvorišni računar dešifrjuje kada stigne na odredište.

- Daljinsko izvršavanje komandi. Ukoliko administrator sistema pokrene komandu na udaljenom računaru (na primer pomoću komande **rsh**), rezultati komande se šalju kao otvoren tekst preko mreže. Ako je rezultat izvršenja ove komande osetljiva informacija, onda se **rsh** mehanizam smatra neprihvatljivim i umesto njega se koristi mehanizam za daljinsko izvršavanje komandi koji obezbeđuje **ssh**. Na primer umesto:

\$ rsh racunar /usr/komanda

administrator će zadati komandu:

\$ssh racunar /usr/komanda

Sintaksa je približno ista, a vidljivi izlaz na korisnikovom ekranu identičan. Međutim, za razliku od **rsh** mehanizma, **ssh** šifrjuje podatke koje prenosi preko mreže, tako da napadač ne može doći do osetljivih informacija koje nastaju kao rezultat izvršenja komandi na udaljenom računaru.

SSH takođe omogućava da se za prijavljivanje na udaljene računare koriste ključevi umesto lozinki. U ovu svrhu se koristi SSH agent za proveru identiteta koji radi na lokalnom računaru. Ova SSH funkcionalnost je naročito zgodna kada korisnik daljinski pristupa većem broju računara čiji su korisnički nalozi zaštićeni različitim dugačkim lozinkama koje se teško pamte.

8. L2 security i konfiguracija ostalih protokola

Statistika pokazuje da je najveći broj napada došao iz same unutrašnje mreže. Layer 2 security je bitan factor bezbednosti cele mreže. U ovom projektu planirana je implementacija više bezbednosnih mehanizama layer 2 nivoa.

Port-security je način da se ograniči broj mac adresa koje mogu prisupiti internoj LAN mreži. Port-security se konfiguriše na interfejsu. Ukoliko se na konfigurisanom interfejsu pojave paketi sa nedozvoljenom, interfejs reaguje u skladu sa jednim od 3 moda port-security mehanizma. Shutdown – interfejs se administrativno gasi za sav saobraćaj, Protect – saobraćaj je zabranjen za pakete sa diskutabilnom mac adresom, Restrict – saobraćaj je zabranjen za pakete sa diskutabilnom mac adresom i SNMP poruka se generiše. Pošto je tri moda kojima port-security uči mac adrese. Static – učenje je statičko, definisano od strane administratora, Dynamic – učenje je dinamičko, Sticky – učenje je dinamično sa dodatnom opcijom pamćenja mac adresa u konfigurativnom fajlu uređaja. Primer :

```
Cisco3560(config)#interface fa0/0
Cisco3560(config-if)#switchport port-security
Cisco3560(config-if)#switchport port-security maximum 3
Cisco3560(config-if)#switchport port-security violation shutdown|restrict|protect
Cisco3560(config-if)#switchport port-security mac-address sticky
```

8.1. Storm Control

Storm control. Storm-control je jednostavan način da se ograniči protok saobraćaja na određenom interfejsu. Jednostavnim ograničenjem i merenjem saobraćaja, u slučaju da se definisana granica prekorači, interfejs se administrativno i manuelno gasi ili se generiše SNMP poruka. Primer :

```
Cisco3560(config)#interface fa0/0
Cisco3560(config-if)#storm-control unicast level {threshold values } pps|bps
Cisco3560(config-if)#storm-control action shutdown|trap
```

8.2. DHCP snooping

DHCP snooping. DHCP snooping je mehanizam koji obezbeđuje internu mrežu od takozvanih man-in-the-middle napada. Interfejsi koji vode ka DHCP serveru se stavljaju u trust mod, i to su jedini interfejsi koji mogu primati dhcp-reply poruke. Ostali interfejsi će zabranjivati dhcp-reply poruke ukoliko se one pojave. Primer:

```
Cisco3560(config)#ip dhcp snooping vlan {vlan range}  
Cisco3560(config)#ip dhcp snooping information option  
Cisco3560(config)#interface fa0/0  
Cisco3560(config-if)#ip dhcp snooping trust
```

8.3. RSPAN

RSPAN (Remote Switched Port Analyzer) je sistem koji se konfiguriše u Ethernet mreži radi sakupljanja podataka o saobraćaju i slanja na software za analizu saobraćaja mreže. Funkcioniše tako što se na uređaju (switch) konfigurišu interfejsi ili čak i vlan-ovi koji se žele pratiti/analizirati i taj saobraćaj se replicira u jedan poseban vlan (konfigurisan samo za potrebe RSPAN) koji saobraćaj šalje na server na kome je instaliran software za analizu saobraćaja. RSPAN je u suštini napredna verzija SPAN protokola. Najveće unapređenje u odnosu na SPAN je to što se može implementirati kroz celu Ethernet mrežu, a ne na ograničen uređaj. Primer:

```
Cisco3560(config)#vlan 900  
Cisco3560(config-v)#name Vlan za rspan  
Cisco3560(config-v)#remote-span  
Cisco3560(config-v)#exit  
Cisco3560(config)#monitor session 1 source vlan 11 both  
Cisco3560(config)#monitor session 1 destination remote vlan 900  
Cisco3560(config)#monitor session 1 destination interface fastEthernet 0/31
```

8.4. Menadžment

Network menadžment protokoli će biti implementirani na Cisco uređajima. Serveri za network menadžment će biti centralizovani na serverima u Beogradu radi bolje i preciznije administracije. Funkcija ovog sistema je da analiza da upravljanje projektovanom mrežom budu pravovremeno, centralizovane i efikasne za administraciju od strane mrežnog administrativnog tima.

Logging. Jako je bitno pratiti događaje u mreži. Log poruke mogu biti od ključne važnosti za celokupni mrežni sistem, pogotovo u slučaju otkaza delova sistema ili upada u mrežu. Log poruke zapisuju vreme određenog događaja u mreži, stepen njegove važnosti za sistem, i kratak opis događaja. Poruke se šalju na definisani server u mreži, odakle mrežni administrator može pratiti mrežna dešavanja. Primer:

```
Cisco2911(config)#logging on
Cisco2911(config)#logging host 10.1.2.10 / adresa servera
Cisco 2911(config)#logging trap 7
```

NTP (Network time Protocol) je protokol za sinhronizaciju vremena u mrežnom sistemu. Funkcioniše po UDP protokolu port 123. Potrebno je definisati referentnu tačku po kojoj ostali uređaji definišu vreme. U ovom slučaju to će biti Cisco2911 ruter u Beogradu, koji će preuzeti NTP Master ulogu u mreži. Ostali uređaji prave klijent-server konekciju sa njim i modifikuju svoje vreme.

```
Cisco2911#clock set 21:00:00 23 Sept 2010
Cisco2911(config)#ntp master
Cisco2911(config)#ntp logging
```

SNMP (Simple Network Management Protocol) je protokol aplikativnog sloja koji omogućava komunikaciju između SNMP servera i agenta. SNMP arhitektura se sastoji od 3 dela : menadžer, agent i MIB. SNMP menadžer je sistem koji kontroliše i analizira mrežne aktivnosti agenata koristeći SNMP protokol. Na menadžeru se instalira software koji vrši analizu, upravljanje i monitoring.

SNMP Agent je uređaj koji komunicira sa SNMP Menadžerom, u ovom slučaju Cisco uređaji. MIB (Management Information Base) je virtuelna baza informacija koja se sastoji od skupa objekata kojima se manipuliše. SNMP Agent sadrži MIB varijable čije vrednosti SNMP Menadžer može da zahteva radi upravljanja ili promene. SNMP agent i menadžer komuniciraju posredstvom trap poruka. SNMP trenutno podržava 3 verzije rada : 1, 2c i 3. Razlika između verzija se najviše ogleda u bezbednosti i autentifikacije SNMP uređaja. U ovom projektu biće korišćena verzija 2c. Razmena poruka u v2c funkcionse pomoću community parametra koji služi kao osnovno sredstvo bezbednosti komunikacije (nešto

slično kao password). Na uređajima se konfigurisu community stringovi, varijable koje se žele pratiti i kojima se želi manipulirati, i lokacija SNMP menadžera. Primer:

```
Cisco2911(config)#snmp-server location Beograd
Cisco2911(config)#snmp-server contact Dunja Adzic
Cisco2911(config)#snmp-server community SNMP_STRING rw
Cisco2911(config)#snmp-server enable traps
Cisco2911(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
Cisco2911(config)#snmp-server system-shutdown
```

8.5. Kvalitet saobraćaja

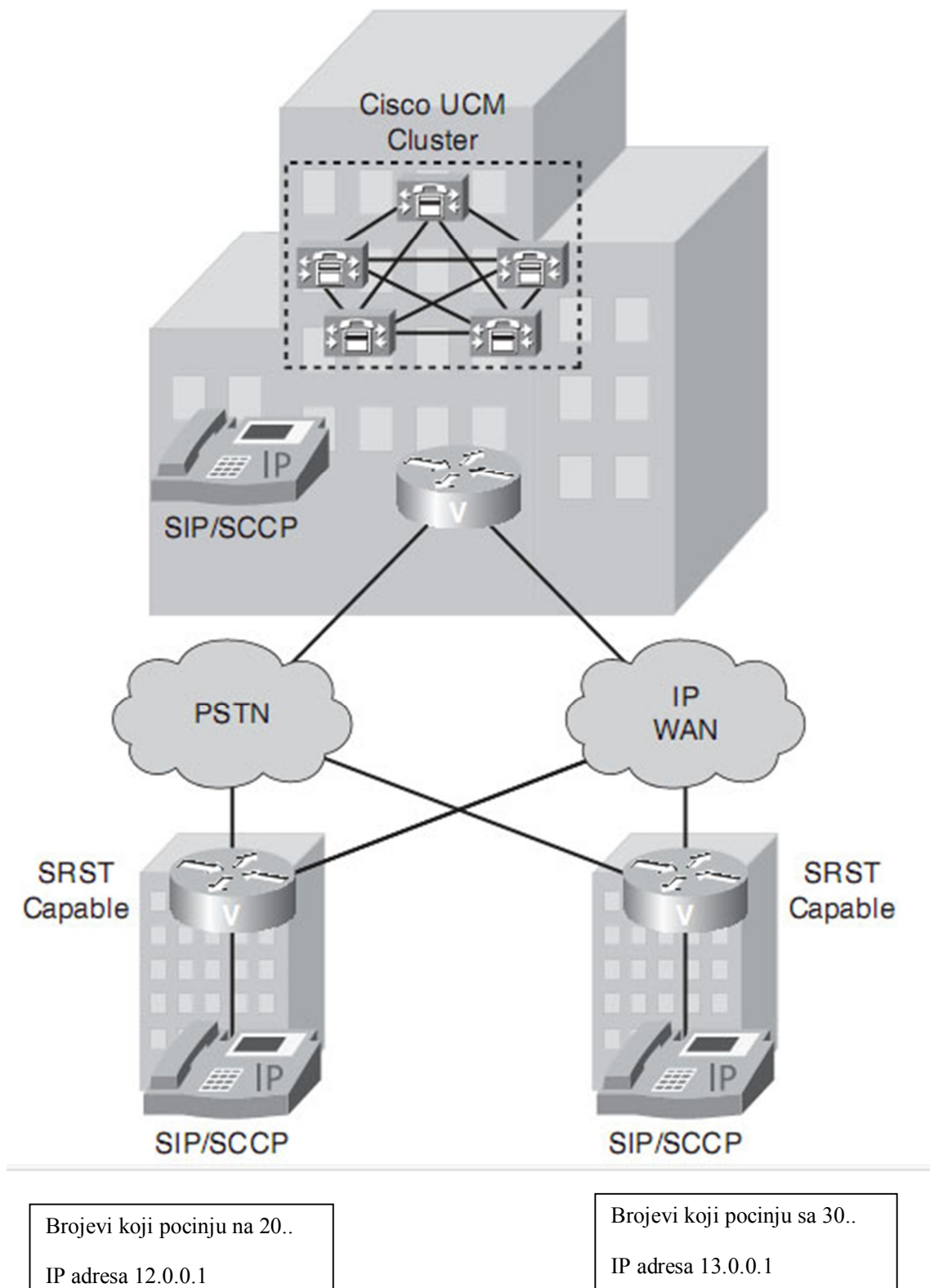
Kvalitet saobraćaja, QOS. QOS (Quality of Service) je skup mehanizama i protokola koji obezbeđuju mreži kvalitetan i nesmetan rad. Mehanizmi kvaliteta servisa obezbeđuju određeni protok saobraćaja, fragmentiranje i kompresiju paketa, sredstva za borbu protiv kašnjenja paketa, zagušenja linkova, prepoznavanje određenih aplikativnih protokola itd. . Cisco kao proizvođač, ima poseban način za implementaciju kvaliteta servisa, MQC (Modular Quality Of Service Command line). MQC je framework, u kome se prvo definišu klase saobraćaja, a na definisanim klasama se konfigurišu željeni protokoli. U ovom projektu koristiće se sledeći QOS protokoli : LLQ, WRED, Traffic policing i NBAR. NBAR (Network Based Application Recognition) je Ciscoov mehanizam za inteligentnu detekciju protokola aplikativnog nivoa (ali i nizih nivoa). NBAR vrši detekciju na razne načine : prepoznavanjem ospega portova aplikacije, aplikativnih potpisa, ponašanja paketa saobraćaja itd. LLQ (Low Latency Queueing) je mehanizam koji saobraćaju određene klase garantuje određeni propusni opseg u vremenu najvećeg zagušenja linka. U klasama se definiše željeni procenat koji je garantovan, a u koliko je u trenutku moguća veća propusna moć linka (link nije zagažen) određena klasa može dobiti i više od zagarantovanog bandwidtha. U LLQ je moguće definisati jednu prioritetsku klasu (u ovom slučaju to će biti klasa gde je definisan VOIP saobraćaj) kojoj se daje fiksni procenat zagarantovanosti linka. To je sav bandwidth koji je dat određenoj klasi, tako da se ona ne može “boriti” za više u slučaju neopterećenosti linka. WRED (Weighted Random Early Detection) je mehanizam koji služi za izbegavanje zagušenja linka. Princip rada WRED-a se zasniva na inteligentnom odabiru paketa koji se uništava definisanim granicama. Ovime se izbegava masivno odbacivanje paketa u slučaju zagaženosti linka i restartovanja ili uništavanja TCP sesija. Traffic policing je način da se saobraćaj ograniči na određenu vrednost. Definiše se bandwidth za sav saobraćaj koji je dozvoljen. Zahtev mreže je da povratni saobraćaj bude ograničen na 4 Mbit/sec. Jedan od zahteva mreže je da nekoristan saobraćaj (torrenti, online video igre...) bude zabranjen, što će se učiniti upotrebom NBAR-a. U ovom projektu biće definisane četiri klase : VOIP klasa (saobraćaj interne telefonije), Klasa-korisnog-saobraćaja (http,sql,ftp...), Klasa-beskorisnog saobraćaja (torrenti, igre...) i class-default (sav ostali saobraćaj). VOIP klasa će LLQ-om biti ograničena na 20% propusne moći internet linka. Klasi korisnog saobraćaja će biti obezbeđeno 40% propusne moći linka ka internetu, dok će ostali saobraćaj zauzeti ostatak. Saobraćaj definisati u klasi Klasa-beskorisnog-saobraćaja će biti zabranjen. WRED će biti implementiran u klasi Klasa-korisnog-saobraćaja. Povratni Saobraćaj u mrežu će biti ograničen na 4Mbit/sec.

```
Cisco2911(config)#class -map match-any VOIP
Cisco2911(config-cmap)#match protocol voice
Cisco2911(config)#class-map match-any Klasa-korisnog-saobracaja
Cisco2911(config-cmap)#match protocol http
Cisco2911(config-cmap)#match protocol ftp
Cisco2911(config-cmap)#match protocol sql
Cisco2911(config-cmap)#match protocol snmp
Cisco2911(config-cmap)#match protocol smtp
Cisco2911(config)#class-map match-any Klasa-beskorisnog-saobracaja
Cisco2911(config-cmap)#match protocol bittorrent
Cisco2911(config-cmap)#match protocol kazaa
Cisco2911(config-cmap)#match protocol gnutella edonkey
```

```
Cisco2911(config)#policy-map PMAP
Cisco2911(config-pmap)#class VOIP
Cisco2911(config-pmap-c)#bandwidth percent 20
Cisco2911(config-pmap)#class Klasa-korisnog-saobracaja
Cisco2911(config-pmap-c)#bandwidth percent 40
Cisco2911(config-pmap-c)#random-detect
Cisco2911(config-pmap)#class Klasa-beskorisnog-saobracaja
Cisco2911(config-pmap-c)#drop
Cisco2911(config-pmap)#class class-default
Cisco2911(config-pmap-c)#exit
Cisco2911(config)#policy-map POLICE
Cisco2911(config-pmap)#class class-default
Cisco2911(config-pmap-c)#police 4000000 conform-action transmit exceed-action
transmit violate-action drop
Cisco2911(config-pmap)#exit
Cisco2911(config)#interface ser0/1
Cisco2911(config-if)#service-policy output PMAP
Cisco2911(config-if)#service-policy input POLICE
```

9. IP telefonija

Model koji je predstavljen na slici 9.1. označava tehnologiju poznatu pod nazivom „Multisite WAN with Centralized Call-Processing Deployment“. Ideja koja leži u osnovi ovakvog dizajna jeste da se u svim razgovorima koji se obavljaju između udaljenih lokacija preskace javna telefonska mreža i na taj način ostvari znatna ušteda na telefonskom racunu. Međutim, svi ruteri moraju biti podešeni tako da u koliko dodje do pada IP mreže, svaki poziv može biti obavljen preko javne telefonske mreže. Ovaj dizajn podrazumeva podizanje Cisco Call Manager klastera na centralnoj lokaciji koji uključuje *Publisher* (glavni server) i *Subscriber* (redundantni server koji svoju bazu na određenom interval vremena osvežava podacima sa glavnog servera). Ovaj klaster bi preko H323 protokola bio povezan na Cisco 2811 ruter koji bi ujedno služio i kao *voice gateway*. Cisco 2811 tip rutera bi bio postavljen i na udaljenim lokacijama iz razloga što se na njemu može instalirati Cisco Call Manager Express softver, koji omogućava rutiranje voice saobraćaja kroz IP mrežu, a i može raditi kao nezavisna telefonska centrala. Vazno je napomenuti da *Cisco Call Manager* i *Cisco Call Manager Express* služe za konfigurisanje telefona i njihovih mogućnosti ali se u slučaju koriscenja H323 protokola celokupne telefonske routing liste konfigurisu na ruterima!



SI.9.1. Multisite WAN with Centralized Call-Processing Deployment

Da bi telefoni mogli da funkcionišu, potrebno im je napajanje. Postoje dva opšte-poznata načina kako prevazići taj problem. Prvi je da svaki telefon ima svoj adapter a drugi način podrazumeva da telefon preko ethernet kablova vuče struju sa POE (power over ethernet) switch-a. Cisco telefoni podržavaju dva protokola za napajanje telefona strujom sa switch-a, to su Cisco Inline koji je zaštićen Cisco-ovim patentom i drugi je protokol pod nazivom 802.3 af koji je opšte prihvaćeni standard. Prilikom paljenja (bootovanja) telefona, telefoni dobijaju IP adresu od lokalnog DHCP servera koji im pored ostalih informacija pružaju i informaciju o adresi TFTP servera, koji mora imati javnu statičku IP adresu, a koji se nalazi u okviru Call Manager klastera. Od tog servera telefoni dobijaju svoju putu konfiguraciju kao i brojeve telefona koje će se koristiti u lokalnom saobraćaju. Ono što je neophodno za bezbednost cele mreže jeste da se ruteri registruju na klaster u SRST modu, što praktično znači to da u koliko klaster prestane sa radom, ruter postaje glavna centrala i on rutira pozive za određeni broj telefona. Besplatni razgovori su mogući isključivo između rutera unutar mreže dok se za sve druge pozive mora obezbediti dovoljan broj javnih telefonskih priključaka da bi cela mreža funkcionisala normalno. Sa obzirom na to da se na svakoj lokaciji nalazi po 50 telefona, najbolje rešenje predstavlja E1 link ka javnoj telefonskoj mreži koji omogućava do 30 simultanih telefonskih poziva u isto vreme.

Primeri konfiguracija na ruteru

<u>Preko IP-a :</u>	<u>Preko PSTN-a</u>
Dial-peer voice 1000 voip	Dial-peer voice 1001 pots
Destination-pattern 10..	Destination-pattern 10..
Session target ipv4:11.0.0.1	port 1/0:23
Preference 1	Preference 2
	Prefix 01145510

U koliko korisnik sa udaljenih lokacija želi da kontaktira neki od brojeva iz Beogradskog sektora on okreće lokalni broj. Ruter gleda koji dial-peer ima manji Preference i pokušava da ostvari poziv preko njega, u našem slučaju je to preko IP-a. U koliko ta putanja nije moguća iz nekog razloga on na okrenuti broj dodaje definisane cifre i takav, sabran, broj šalje javnoj telefonskoj mreži koja će dalje taj poziv preuslediti do krajnje destinacije. Cisco ruteri koji rade rutiranje voice saobraćaja u slučaju definisanja POTS dial-peer-ova odklanjaju sve tačno

definisane cifre sto se moze videti i na prikazu gore, gde je u okviru prefiksa koji ce biti dodan broju nalaze i 10.

10. Konfiguracije rutera i switcheva

Konfiguracije rutera:

Cisco Ruter u Beogradu

```
Router2911>enable
Router2911#configure terminal
Router2911(config)#hostname Beograd2911
Beograd2911(config)#interface s0/1
Beograd2911(config-if)#no shutdown
Beograd2911(config-if)#ip address x.x.x.x / adresa dobijena od Internet provajdera
Beograd2911(config-if)#description Interfejs ka intenetu
Beograd2911(config-if)#exit
Beograd2911(config)#interface ser0/0
Beograd2911(config-if)#ip address x.x.x.x /adresa dobijena od MPLS servis provajdera
Beograd2911(config-if)#description MPLS/VPN interfejs
Beograd2911(config-if)#no shut
Beograd2911(config)#interface fa0/0
Beograd2911(config-if)#description Interfes ka SW1
Beograd2911(config-if)#ip address 10.1.12.1 255.255.255.0
Beograd2911(config-if)#ip ospf hello-interval 1
Beograd2911(config-if)#ip ospf dead-interval 3
Beograd2911(config-if)#no shutdown
Beograd2911(config)#interface fa0/1
Beograd2911(config-if)#ip ospf hello-intreval 1
Beograd2911(config-if)#ip ospf dead-intervaln 3
Beograd2911(config-if)#description Interface ka SW2
Beograd2911(config-if)#ip add 10.1.13.1 255.255.255.0
Beograd2911(config-if)#no shut
Beograd2911(config)#router ospf 1
Beograd2911(config-r)#router-id 1.1.1.1
Beograd2911(config-if)#network 10.1.0.0 0.0.255.255 area 1
```

```
Beograd2911(config-r)#area 1 range 10.1.0.0 255.255.0.0
Beograd2911(config-r)#default-information originate
Beograd2911(config-r)#exit
Beograd2911(config)#ip route 0.0.0.0 0.0.0.0 serial 0/1
Beograd2911(config)#ip access-list standard NAT-ACL
Beograd2911(config-acl)#permit ip 10.0.0.0 0.255.255.255
Beograd2911(config-acl)#exit
Beograd2911(config)#ip nat inside source list NAT interface s0/1 overload
Beograd2911(config)#inter s0/1
Beograd2911(config-if)#ip nat outside
Beograd2911(config-if)#exit
Beograd2911(config)#interface fa0/0
Beograd2911(config-if)#ip nat inside
Beograd2911(config-if)#exit
Beograd2911(config)#interface fa0/1
Beograd2911(config-if)#ip nat inside
Beograd2911(config-if)#exit
Beograd2911(config)#ip inspect name FWRULE tcp
Beograd2911(config)#ip inspect name FWRULE udp
Beograd2911(config)#ip inspect name FWRULE icmp
Beograd2911(config)#ip inspect name FWRULE http
Beograd2911(config)#ip inspect name FWRULE snmp
Beograd2911(config)#ip inspect name FWRULE smtp
Beograd2911(config)#ip inspect name FWRULE ssh
Beograd2911(config)#ip inspect name FWRULE ftp
Beograd2911(config)#ip inspect name FWRULE syslog
Beograd2911(config)#ip inspect tcp synwait-time 30
Beograd2911(config)#ip inspect tcp finwait-time 30
Beograd2911(config)#ip inspect tcp idle-time 30
Beograd2911(config)#ip inspect tcp mac-incomplete host 100
Beograd2911(config)#ip inspect udp idle-time 30
Beograd2911(config)#ip inspect audit-trail
Beograd2911(config)#no ip inspect alert-off
Beograd2911(config)#ip inspect max-incomplete high 150
Beograd2911(config)#ip inspect max-incomplete low 75
```

```
Beograd2911(config)# access-list 100 deny tcp any any
Router2911(config)# access-list 100 deny udp any any
Beograd2911(config)# access-list 100 permit icmp any any echo-reply
Beograd2911(config)# access-list 100 permit icmp any any time-exceeded
Beograd2911(config)# access-list 100 permit icmp any any packet-too-big
Beograd2911(config)# access-list 100 permit icmp any any traceroute
Beograd2911(config)# access-list 100 permit icmp any any unreachable
Beograd2911(config)# access-list 100 deny ip any any
Beograd2911(config)#interface s0/1
Beograd2911(config-if)#ip access-group 100 in
Beograd2911(config-if)#ip inspect FWRULE out
Beograd2911(config-if)#exit
Beograd2911(config)#ip domain-name dunjainc.com
Beograd2911(config)#crypto key generate rsa modulus 1024
Beograd2911(config)#ip ssh version 2
Beograd2911(config)#line vty 0 4
Beograd2911(config-line )#transport input ssh
Beograd2911(config-line)#transport output ssh
Beograd2911(config-line)#exit
Beograd2911(config)#aaa new-model
Beograd2911(config)#radius-server host 10.1.2.18
Beograd2911(config)#aaa authentication login default radius local
Beograd2911(config)#aaa authentication attempts login 3
Beograd2911(config)#aaa authorization exec default radius
Beograd2911(config)#aaa authorization commands default radius
Beograd2911(config)#aaa accounting exec default start-stop
Beograd2911(config)#aaa accounting commands default start-stop
Beograd2911(config)#snmp-server location Beograd
Beograd2911(config)#snmp-server contact Dunja Adzic
Beograd2911(config)#snmp-server community SNMP_STRING rw
Beograd2911(config)#snmp-server enable traps
Beograd2911(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
Beograd2911(config)#snmp-server system-shutdown
Beograd2911(config)#logging on
Beograd2911(config)#logging host 10.1.2.10
```

```
Beograd2911(config)#logging trap 7
Beograd2911(config)#exception protocol ftp
Beograd2911(config)#exception dump 10.1.2.15
Beograd2911(config)#exception corefile dunjacoredump
Beograd2911(config)#exit
Beograd2911#clock set 21:00:00 23 Sept 2010
Beograd2911#configure terminal
Beograd2911(config)#ntp master
Beograd2911(config)#ntp logging
Beograd2911(config)#crypto isakmp enable
Beograd2911(config)#crypto isakmp policy 112
Beograd2911(config-isakmp)#encryption aes 256
Beograd2911(config-isakmp)#hash md5
Beograd2911(config-isakmp)#authentication pre-share
Beograd2911(config-isakmp)#group 2
Beograd2911(config-isakmp)#lifetime 36000
Beograd2911(config-isakmp)#crypto isakmp policy 113
Beograd2911(config-isakmp)#encryption aes 256
Beograd2911(config-isakmp)#hash md5
Beograd2911(config-isakmp)#authentication pre-share
Beograd2911(config-isakmp)#group 2
Beograd2911(config-isakmp)#lifetime 36000
Beograd2911(config)#crypto isakmp key cisco12 address 10.12.0.2
Beograd2911(config)#crypto isakmp key cisco13 address 10.13.0.3
Beograd2911(config)#crypto ipsec transform-set CISCO esp-des esp-md5-hmac
Beograd2911(config)#access-list 112 permit ip 10.1.0.0 0.0.255.255 10.2.0.0
0.0.255.255
Beograd2911(config)#access-list 113 permit ip 10.1.0.0 0.0.255.255 10.3.0.0
0.0.255.255
Beograd2911(config)#crypto map BEOGRAD-NOVISAD 112 ipsec-isakmp
Beograd2911(config-crypto-map)#match address 112
Beograd2911(config-crypto-map)#set peer 10.12.0.2
Beograd2911(config-crypto-map)#set transform-set CISCO
Beograd2911(config-crypto-map)#set security-accociation lifetime seconds 36000
Beograd2911(config-crypto-map)#exit
```

```

Beograd2911(config)#crypto map BEOGRAD-NIS 113 ipsec-isakmp
Beograd2911(config-crypto-map)#match address 113
Beograd2911(config-crypto-map)#set peer 10.13.0.3
Beograd2911(config-crypto-map)#set transform-set CISCO
Beograd2911(config-crypto-map)#set security-association lifetime seconds 36000
Beograd2911(config-crypto-map)#exit
Beograd2911(config)#interface tunnel 12
Beograd2911(config-if)#ip ospf 1 area 0
Beograd2911(config-if)#description Tunel interfejs ka Novom Sadu
Beograd2911(config-if)#ip address 10.12.0.1 255.255.255.0
Beograd2911(config-if)#tunnel source serial 0/0
Beograd2911(config-if)#tunnel destination x.x.x.x /adresa interfejsa ka MPLS/VPN
provajderu u Novom Sadu
Beograd2911(config-if)#tunnel mode gre ip
Beograd2911(config-if)#crypto map BEOGRAD-NOVISAD
Beograd2911(config-if)#no shut
Beograd2911(config-if)#exit
Beograd2911(config)#interface tunnel 13
Beograd2911(config-if)#ip ospf 1 area 0
Beograd2911(config-if)#description Tunel interface ka Nisu
Beograd2911(config-if)#ip address 10.13.0.1 255.255.255.0
Beograd2911(config-if)#tunnel mode gre ip
Beograd2911(config-if)#tunnel source serial 0/0
Beograd2911(config-if)#tunnel destination x.x.x.x / adresa interfejsa ka MPLS/VPN
provajderu u Nisu
Beograd2911(config-if)#crypto map BEOGRAD-NIS
Beograd2911(config-if)#no shut
Beograd2911(config-if)#exit
Beograd2911(config)#class-map match-any VOIP
Beograd2911(config-cmap)#match protocol voice
Beograd2911(config)#class-map match-any Klasa-korisnog-saobracaja
Beograd2911(config-cmap)#match protocol http
Beograd2911(config-cmap)#match protocol ftp
Beograd2911(config-cmap)#match protocol sql
Beograd2911(config-cmap)#match protocol snmp

```

```

Beograd2911(config-cmap)#match protocol smtp
Beograd2911(config)#class-map match-any Klasa-beskorisnog-saobracaja
Beograd2911(config-cmap)#match protocol bittorrent
Beograd2911(config-cmap)#match protocol kazaa
Beograd2911(config-cmap)#match protocol gnutella edonkey
Beograd2911(config)#policy-map PMAP
Beograd2911(config-pmap)#class VOIP
Beograd2911(config-pmap-c)#bandwidth percent 20
Beograd2911(config-pmap)#class Klasa-korisnog-saobracaja
Beograd2911(config-pmap-c)#bandwidth percent 40
Beograd2911(config-pmap-c)#random-detect
Beograd2911(config-pmap)#class Klasa-beskorisnog-saobracaja
Beograd2911(config-pmap-c)#drop
Beograd2911(config-pmap)#class class-default
Beograd2911(config-pmap-c)#exit
Beograd2911(config)#policy-map POLICE
Beograd2911(config-pmap)#class class-default
Beograd2911(config-pmap-c)#police 4000000 conform-action transmit exceed-action
transmit violate-action drop
Beograd2911(config-pmap)#exit
Beograd2911(config)#interface ser0/1
Beograd2911(config-if)#service-policy output PMAP
Beograd2911(config-if)#service-policy input POLICE

```

Cisco Ruter u Novom Sadu

```

Router2911>enable
Router2911#configure terminal
Router2911(config)#hostname Beograd2911
Novisad2911(config)#interface ser0/0
Novisad2911(config-if)#ip address x.x.x.x /adresa dobijena od MPLS servis provajdera
Novisad2911(config-if)#description MPLS/VPN interfejs

```

```

Novisad2911(config-if)#no shut
Novisad2911(config)#interface fa0/0
Novisad2911(config-if)#description Interfes ka SW1
Novisad2911(config-if)#ip address 10.2.12.1 255.255.255.0
Novisad2911(config-if)#ip ospf hello-interval 1
Novisad2911(config-if)#ip ospf dead-interval 3
Novisad2911(config-if)#no shutdown
Novisad2911(config)#interface fa0/1
Novisad2911(config-if)#ip ospf hello-interval 1
Novisad2911(config-if)#ip ospf dead-interval 3
Novisad2911(config-if)#description Interface ka SW2
Novisad2911(config-if)#ip add 10.2.13.1 255.255.255.0
Novisad2911(config-if)#no shut
Novisad2911(config)#router ospf 1
Novisad2911(config-r)#router-id 2.2.2.2
Novisad2911(config-if)#network 10.2.0.0 0.0.255.255 area 2
Novisad2911(config-r)#area 2 range 10.2.0.0 255.255.0.0
Novisad2911(config-r)#exit
Novisad2911(config)#ip domain-name dunjainc.com
Novisad2911(config)#crypto key generate rsa modulus 1024
Novisad2911(config)#ip ssh version 2
2911(config)#line vty 0 4
Novisad2911(config-line)#transport input ssh
Novisad2911(config-line)#transport output ssh
Novisad2911(config-line)#exit
Novisad2911(config)#aaa new-model
Novisad2911(config)#radius-server host 10.1.2.18
Novisad2911(config)#aaa authentication login default radius local
Novisad2911(config)#aaa authentication attempts login 3
Novisad2911(config)#aaa authorization exec default radius
Novisad2911(config)#aaa authorization commands default radius
Novisad2911(config)#aaa accounting exec default start-stop
Novisad2911(config)#aaa accounting commands default start-stop
Novisad2911(config)#snmp-server location Beograd
Novisad2911(config)#snmp-server contact Dunja Adzic

```

```
Novisad2911(config)#snmp-server community SNMP_STRING rw
Novisad2911(config)#snmp-server enable traps
Novisad2911(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
Novisad2911(config)#snmp-server system-shutdown
Novisad2911(config)#logging on
Novisad2911(config)#logging host 10.1.2.10
Novisad2911(config)#logging trap 7
Novisad2911(config)#exception protocol ftp
Novisad2911(config)#exception dump 10.1.2.15
Novisad2911(config)#exception corefile dunjacoredump
Novisad2911(config)#ntp server 10.12.0.1
Novisad2911(config)#ntp logging
Novisad2911(config)#crypto isakmp enable
Novisad2911(config)#crypto isakmp policy 112
Novisad2911(config-isakmp)#encryption aes 256
Novisad2911(config-isakmp)#hash md5
Novisad2911(config-isakmp)#authentication pre-share
Novisad2911(config-isakmp)#group 2
Novisad2911(config-isakmp)#lifetime 36000
Novisad2911(config-isakmp)#crypto isakmp policy 113
Novisad2911(config-isakmp)#encryption aes 256
Novisad2911(config-isakmp)#hash md5
Novisad2911(config-isakmp)#authentication pre-share
Novisad2911(config-isakmp)#group 2
Novisad2911(config-isakmp)#lifetime 36000
Novisad2911(config)#crypto isakmp key cisco12 address 10.12.0.1
Novisad2911(config)#crypto isakmp key cisco13 address 10.23.0.3
Novisad2911(config)#crypto ipsec transform-set CISCO esp-des esp-md5-hmac
Novisad2911(config)#access-list 112 permit ip 10.2.0.0 0.0.255.255 10.1.0.0 0.0.255.255
Novisad2911(config)#access-list 123 permit ip 10.2.0.0 0.0.255.255 10.3.0.0 0.0.255.255
Novisad2911(config)#crypto map BEOGRAD-NOVISAD 112 ipsec-isakmp
Novisad2911(config-crypto-map)#match address 112
Novisad2911(config-crypto-map)#set peer 10.12.0.1
Novisad2911(config-crypto-map)#set transform-set CISCO
Novisad2911(config-crypto-map)#set security-accociation lifetime seconds 36000
```

```
Novisad2911(config-crypto-map)#exit
Novisad2911(config)#crypto map NOVISAD-NIS 113 ipsec-isakmp
Novisad2911(config-crypto-map)#match address 123
Novisad2911(config-crypto-map)#set peer 10.23.0.3
Novisad2911(config-crypto-map)#set transform-set CISCO
Novisad2911(config-crypto-map)#set security-association lifetime seconds 36000
Novisad2911(config-crypto-map)#exit
Novisad2911(config)#interface tunnel 12
Novisad2911(config-if)#ip ospf 1 area 0
Novisad2911(config-if)#description Tunel interfejs ka Novom Sadu
Novisad2911(config-if)#ip address 10.12.0.2 255.255.255.0
Novisad2911(config-if)#tunnel source serial 0/0
Novisad2911(config-if)#tunnel destination x.x.x.x /adresa interfejsa ka MPLS/VPN
provajderu u Beogradu
Novisad2911(config-if)#tunnel mode gre ip
Novisad2911(config-if)#crypto map BEOGRAD-NOVISAD
Novisad2911(config-if)#no shut
Novisad2911(config-if)#exit
Novisad2911(config)#interface tunnel 23
Novisad2911(config-if)#ip ospf 1 area 0
Novisad2911(config-if)#description Tunel interface ka Nisu
Novisad2911(config-if)#ip address 10.23.0.2 255.255.255.0
Novisad2911(config-if)#tunnel mode gre ip
Novisad2911(config-if)#tunnel source serial 0/0
Novisad2911(config-if)#tunnel destination x.x.x.x / adresa interfejsa ka MPLS/VPN
provajderu u Nisu
Novisad2911(config-if)#crypto map NOVI-NIS
Novisad2911(config-if)#no shut
Novisad2911(config-if)#exit
```

Cisco Ruter u Nišu

```
Router2911>enable
Router2911#configure terminal
Router2911(config)#hostname Nis2911
Nis 2911(config)#interface ser0/0
Nis 2911(config-if)#ip address x.x.x.x /adresa dobijena od MPLS servis provajdera
Nis 2911(config-if)#description MPLS/VPN interfejs
Nis 2911(config-if)#no shut
Nis 2911(config)#interface fa0/0
Nis 2911(config-if)#description Interfes ka SW1
Nis 2911(config-if)#ip address 10.3.12.1 255.255.255.0
Nis 2911(config-if)#ip ospf hello-interval 1
Nis 2911(config-if)#ip ospf dead-interval 3
Nis 2911(config-if)#no shutdown
Nis 2911(config)#interface fa0/1
Nis 2911(config-if)#ip ospf hello-intreval 1
Nis 2911(config-if)#ip ospf dead-interval 3
Nis 2911(config-if)#description Interface ka SW2
Nis 2911(config-if)#ip add 10.3.13.1 255.255.255.0
Nis 2911(config-if)#no shut
Nis 2911(config)#router ospf 1
Nis 2911(config-r)#router-id 3.3.3.3
Nis 2911(config-if)#network 10.3.0.0 0.0.255.255 area 3
Nis 2911(config-r)#area 2 range 10.3.0.0 255.255.0.0
Nis 2911(config-r)#exit
Nis 2911(config)#ip domain-name dunjainc.com
Nis 2911(config)#crypto key generate rsa modulus 1024
Nis 2911(config)#ip ssh version 2
Nis 2911(config)#line vty 0 4
Nis 2911(config-line )#transport input ssh
```

```

Nis 2911(config-line)#transport output ssh
Nis 2911(config-line)#exit
Nis 2911(config)#aaa new-model
Nis 2911(config)#radius-server host 10.1.2.18
Nis 2911(config)#aaa authentication login default radius local
Nis 2911(config)#aaa authentication attempts login 3
Nis 2911(config)#aaa authorization exec default radius
Nis 2911(config)#aaa authorization commands default radius
Nis 2911(config)#aaa accounting exec default start-stop
Nis 2911(config)#aaa accounting commands default start-stop
Nis 2911(config)#snmp-server location Beograd
Nis 2911(config)#snmp-server contact Dunja Adzic
Nis 2911(config)#snmp-server community SNMP_STRING rw
Nis 2911(config)#snmp-server enable traps
Nis 2911(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
Nis 2911(config)#snmp-server system-shutdown
Nis 2911(config)#logging on
Nis 2911(config)#logging host 10.1.2.10
Nis2911(config)#logging trap 7
Nis2911(config)#exception protocol ftp
Nis2911(config)#exception dump 10.1.2.15
Nis2911(config)#exception corefile dunjacoredump
Nis2911(config)#ntp server 10.12.0.1
Nis 2911(config)#ntp logging
Nis 2911(config)#crypto isakmp enable
Nis 2911(config)#crypto isakmp policy 112
Nis 2911(config-isakmp)#encryption aes 256
Nis 2911(config-isakmp)#hash md5
Nis 2911(config-isakmp)#authentication pre-share
Nis 2911(config-isakmp)#group 2
Nis 2911(config-isakmp)#lifetime 36000
Nis 2911(config-isakmp)#crypto isakmp policy 113
Nis 2911(config-isakmp)#encryption aes 256
Nis 2911(config-isakmp)#hash md5
Nis 2911(config-isakmp)#authentication pre-share

```

```

Nis 2911(config-isakmp)#group 2
Nis 2911(config-isakmp)#lifetime 36000
Nis 2911(config)#crypto isakmp key cisco12 address 10.12.0.1
Nis 2911(config)#crypto isakmp key cisco13 address 10.23.0.2
Nis 2911(config)#crypto ipsec transform-set CISCO esp-des esp-md5-hmac
Nis 2911(config)#access-list 112 permit ip 10.3.0.0 0.0.255.255 10.1.0.0 0.0.255.255
Nis 2911(config)#access-list 123 permit ip 10.3.0.0 0.0.255.255 10.2.0.0 0.0.255.255
Nis 2911(config)#crypto map BEOGRAD-NIŠ 112 ipsec-isakmp
Nis 2911(config-crypto-map)#match address 112
Nis 2911(config-crypto-map)#set peer 10.13.0.1
Nis 2911(config-crypto-map)#set transform-set CISCO
Nis 2911(config-crypto-map)#set security-association lifetime seconds 36000
Nis 2911(config-crypto-map)#exit
Nis 2911(config)#crypto map NOVISAD-NIS 113 ipsec-isakmp
Nis 2911(config-crypto-map)#match address 123
Nis 2911(config-crypto-map)#set peer 10.23.0.2
Nis 2911(config-crypto-map)#set transform-set CISCO
Nis 2911(config-crypto-map)#set security-association lifetime seconds 36000
Nis 2911(config-crypto-map)#exit
Nis 2911(config)#interface tunnel 12
Nis 2911(config-if)#ip ospf 1 area 0
Nis 2911(config-if)#description Tunel interfejs ka Beogradu
Nis 2911(config-if)#ip address 10.13.0.3 255.255.255.0
Nis 2911(config-if)#tunnel source serial 0/0
Nis 2911(config-if)#tunnel destination x.x.x.x /adresa interfejsa ka MPLS/VPN
provajderu u Beogradu
Nis 2911(config-if)#tunnel mode gre ip
Nis 2911(config-if)#crypto map BEOGRAD-NIS
Nis 2911(config-if)#no shut
Nis 2911(config-if)#exit
Nis 2911(config)#interface tunnel 23
Nis 2911(config-if)#ip ospf 1 area 0
Nis 2911(config-if)#description Tunel interface ka Novom Sadu
Nis 2911(config-if)#ip address 10.23.0.3 255.255.255.0
Nis 2911(config-if)#tunnel mode gre ip

```

```
Nis 2911(config-if)#tunnel source serial 0/0
Nis 2911(config-if)#tunnel destination x.x.x.x / adresa interfejsa ka MPLS/VPN
provajderu u Novom Sadu
Nis 2911(config-if)crypto map NOVI-NIS
Nis 2911(config-if)#no shut
Nis 2911(config-if)#exit
```

Konfiguracija switcheva:

Beograd Switch 1

```
Switch3560>enable
Switch3560#configure terminal
Switch3560(config)#hostname BeogradSW1
BeogradSW1(config)#vlan 11
BeogradSW1(config-vlan)#name DataVlan
BeogradSW1(config-vlan)#exit
BeogradSW1(config)#vlan 12
BeogradSW1(config-vlan)#name VoiceVlan
BeogradSW1(config-vlan)#exit
BeogradSW1(config)#vlan 13
BeogradSW1(config-vlan)#name MenagmentVlan
BeogradSW1(config-vlan)#exit
BeogradSW1(config)#vlan 900
BeogradSW1(config-vlan)#rspan vlan
BeogradSW1(config-vlan)#name RSPANVlan
BeogradSW1(config-vlan)#exit
BeogradSW1(config)#spanning-tree mode rstp
BeogradSW1(config)#spanning-tree vlan 11 priority 0
BeogradSW1(config)#spanning-tree vlan 12 priority 0
BeogradSW1(config)#spanning-tree vlan 13 priority 0
BeogradSW1(config)#spanning-tree vlan 900 priority 0
BeogradSW1(config)#vtp mode server
```

```

BeogradSW1(config)#vtp version 2
BeogradSW1(config)#vtp domain Dunjainc
BeogradSW1(config)#vtp password dunjaincpass
BeogradSW1(config)#interface port-channel 1
BeogradSW1(config-if)#no shut
BeogradSW1(config-if)#switchport mode trunk
BeogradSW1(config-if)#switchport trunk encapsulation dot1q
BeogradSW1(config-if)#channel-group 1 mode on
BeogradSW1(config)#interface range fa0/1 - fa0/8
BeogradSW1(config-if)#no shut
BeogradSW1(config-if)#exit
BeogradSW1(config-if)#channel-group 1 mode on
BeogradSW1(config-if)#exit
BeogradSW1(config)#interface fa0/9
BeogradSW1(config-if)#no switchport
BeogradSW1(config-if)#ip address 10.1.12.2 255.255.255.0
BeogradSW1(config-if)#ip ospf hello-interval 1
BeogradSW1(config-if)#ip ospf dead-interval 3
BeogradSW1(config-if)#no shut
BeogradSW1(config-if)#exit
BeogradSW1(config)#interface range fa0/10 - fa0/35
BeogradSW1(config-if)#description Ka hostovima
BeogradSW1(config-if)#switchport mode access
BeogradSW1(config-if)#switchport access vlan 11
BeogradSW1(config-if)#spanning-tree port-fast
BeogradSW1(config-if)#switchport port-security
BeogradSW1(config-if)#switchport port-security maximum 3
BeogradSW1(config-if)#switchport port-security violation restrict
BeogradSW1(config-if)#switchport port-security mac-address sticky
BeogradSW1(config-if)#storm-control unicast level {threshold values } pps|bps
BeogradSW1(config-if)#storm-control action trap
BeogradSW1(config-if)#exit
BeogradSW1(config)#ip dhcp snooping vlan 11 - 13
BeogradSW1(config)#ip dhcp snooping information-option
BeogradSW1(config)#Interface vlan 11

```

```

BeogradSW1(config-if)#no shut
BeogradSW1(config-if)#ip address 10.1.0.1 255.255.255.0
BeogradSW1(config-if)#exit
BeogradSW1(config)#interface vlan 12
BeogradSW1(config-if)#no shut
BeogradSW1(config-if)#ip add 10.1.1.1 255.255.255.0
BeogradSW1(config-if)#exit
BeogradSW1(config)#interface vlan 13
BeogradSW1(config-if)#no shut
BeogradSW1(config-if)#ip address 10.1.2.1 255.255.255.0
BeogradSW1(config-if)#exit
BeogradSW1(config)#interface range fa0/35- fa0/40
BeogradSW1(config-if)#switchport mode access
BeogradSW1(config-if)#description Ka Lokalnim Serverima
BeogradSW1(config-if)#switchport access vlan 13
BeogradSW1(config-if)#spanning-tree port-fast
BeogradSW1(config-if)#switchport port-security
BeogradSW1(config-if)#switchport port-security maximum 3
BeogradSW1(config-if)#switchport port-security violation restrict
BeogradSW1(config-if)#switchport port-security mac-address sticky
BeogradSW1(config-if)#storm-control unicast level {threshold values } pps|bps
BeogradSW1(config-if)#storm-control action trap
BeogradSW1(config-if)#exit
BeogradSW1(config)#snmp-server location Beograd
BeogradSW1(config)#snmp-server contact Dunja Adzic
BeogradSW1(config)#snmp-server community SNMP_STRING rw
BeogradSW1(config)#snmp-server enable traps
BeogradSW1(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
BeogradSW1(config)#snmp-server system-shutdown
BeogradSW1(config)#logging on
BeogradSW1(config)#logging host 10.1.2.10
BeogradSW1(config)#logging trap 7
BeogradSW1(config)#exception protocol ftp
BeogradSW1(config)#exception dump 10.1.2.15
BeogradSW1(config)#exception corefile dunjacoredump

```

```

BeogradSW1(config)#ntp server 10.12.0.1
BeogradSW1(config)#ntp logging
BeogradSW1(config)#aaa new-model
BeogradSW1(config)#radius-server host 10.1.2.18
BeogradSW1(config)#aaa authentication login default radius local
BeogradSW1(config)#aaa authentication attempts login 3
BeogradSW1(config)#aaa authorization exec default radius
BeogradSW1(config)#aaa authorization commands default radius
BeogradSW1(config)#aaa accounting exec default start-stop
BeogradSW1(config)#aaa accounting commands default start-stop
BeogradSW1(config)#ip domain-name dunjainc.com
BeogradSW1(config)#crypto key generate rsa modulus 1024
BeogradSW1(config)#ip ssh version 2
BeogradSW1(config)#line vty 0 4
BeogradSW1(config-line)#transport input ssh
BeogradSW1(config-line)#transport output ssh
BeogradSW1(config-line)#exit
BeogradSW1(config)#monitor session 1 source vlan 13 both
BeogradSW1(config)#monitor session 1 destination remote vlan 900
BeogradSW1(config)#router ospf 1
BeogradSW1(config-r)#router-id 1.1.1.11
BeogradSW1(config-r)#network 0.0.0.0 255.255.255.255 area 1
BeogradSW1(config-r)#exit

```

Beograd Switch 2

```

Switch3560>enable
Switch3560#configure terminal
Switch3560(config)#hostname BeogradSW2
BeogradSW2(config)#vlan 11
BeogradSW2(config-vlan)#name DataVlan
BeogradSW2(config-vlan)#exit
BeogradSW2(config)#vlan 12

```

```
BeogradSW2(config-vlan)#name VoiceVlan
BeogradSW2(config-vlan)#exit
BeogradSW2(config)#vlan 13
BeogradSW2(config-vlan)#name MenagmentVlan
BeogradSW2(config-vlan)#exit
BeogradSW2(config)#vlan 900
BeogradSW2(config-vlan)#rspan vlan
BeogradSW2(config-vlan)#name RSPANVlan
BeogradSW2(config-vlan)#exit
BeogradSW2(config)#spanning-tree mode rstp
BeogradSW2(config)#vtp mode client
BeogradSW2(config)#vtp version 2
BeogradSW2(config)#vtp domain Dunjainc
BeogradSW2(config)#vtp password dunjaincpass
BeogradSW2(config)#interface port-channel 1
BeogradSW2(config-if)#no shut
BeogradSW2(config-if)#switchport mode trunk
BeogradSW2(config-if)#switchport trunk encapsulation dot1q
BeogradSW2(config-if)#channel-group 1 mode on
BeogradSW2(config)#interface range fa0/1 - fa0/8
BeogradSW2(config-if)#no shut
BeogradSW2(config-if)#exit
BeogradSW2(config-if)#channel-group 1 mode on
BeogradSW2(config-if)#exit
BeogradSW2(config)#interface fa0/9
BeogradSW2(config-if)#no switcport
BeogradSW2(config-if)#ip address 10.1.13.2 255.255.255.0
BeogradSW2(config-if)#ip ospf hello-interval 1
BeogradSW2(config-if)#ip ospf dead-interval 3
BeogradSW2(config-if)#no shut
BeogradSW2(config-if)#exit
BeogradSW2(config)#interface range fa0/10 - fa0/35
BeogradSW2(config-if)#description Ka hostovima
BeogradSW2(config-if)#switchport mode access
BeogradSW2(config-if)#switchport access vlan 11
```

```

BeogradSW2(config-if)#spanning-tree port-fast
BeogradSW2(config-if)#switchport port-security
BeogradSW2(config-if)#switchport port-security maximum 3
BeogradSW2(config-if)#switchport port-security violation restrict
BeogradSW2(config-if)#switchport port-security mac-address sticky
BeogradSW2(config-if)#storm-control unicast level {threshold values } pps|bps
BeogradSW2(config-if)#storm-control action trap
BeogradSW2(config-if)#exit
BeogradSW2(config)#ip dhcp snooping vlan 11 - 13
BeogradSW2(config)#ip dhcp snooping information-option
BeogradSW2(config)#Interface vlan 11
BeogradSW2(config-if)#no shut
BeogradSW2(config-if)#ip address 10.1.0.255.255.255.0
BeogradSW2(config-if)#exit
BeogradSW2(config)#interface vlan 12
BeogradSW2(config-if)#no shut
BeogradSW2(config-if)#ip add 10.1.1.2 255.255.255.0
BeogradSW2(config-if)#exit
BeogradSW2(config)#interface vlan 13
BeogradSW2(config-if)#no shut
BeogradSW2(config-if)#ip address 10.1.2.2 255.255.255.0
BeogradSW2(config-if)#exit
BeogradSW2(config)#interface range fa0/35- fa0/40
BeogradSW2(config-if)#switchport mode access
BeogradSW2(config-if)#description Ka Lokalnim Serverima
BeogradSW2(config-if)#switchport access vlan 13
BeogradSW2(config-if)#spanning-tree port-fast
BeogradSW2(config-if)#switchport port-security
BeogradSW2(config-if)#switchport port-security maximum 3
BeogradSW2(config-if)#switchport port-security violation restrict
BeogradSW2(config-if)#switchport port-security mac-address sticky
BeogradSW2(config-if)#storm-control unicast level {threshold values } pps|bps
BeogradSW2(config-if)#storm-control action trap
BeogradSW2(config-if)#exit
BeogradSW2(config)#snmp-server location Beograd

```

```
BeogradSW2(config)#snmp-server contact Dunja Adzic
BeogradSW2(config)#snmp-server community SNMP_STRING rw
BeogradSW2(config)#snmp-server enable traps
BeogradSW2(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
BeogradSW2(config)#snmp-server system-shutdown
BeogradSW2(config)#logging on
BeogradSW2(config)#logging host 10.1.2.10
BeogradSW2(config)#logging trap 7
BeogradSW2(config)#exception protocol ftp
BeogradSW2(config)#exception dump 10.1.2.15
BeogradSW2(config)#exception corefile dunjacoredump
BeogradSW2(config)#ntp server 10.12.0.1
BeogradSW2(config)#ntp logging
BeogradSW2(config)#aaa new-model
BeogradSW2(config)#radius-server host 10.1.2.18
BeogradSW2(config)#aaa authentication login default radius local
BeogradSW2(config)#aaa authentication attempts login 3
BeogradSW2(config)#aaa authorization exec default radius
BeogradSW2(config)#aaa authorization commands default radius
BeogradSW2(config)#aaa accounting exec default start-stop
BeogradSW2(config)#aaa accounting commands default start-stop
BeogradSW2(config)#ip domain-name dunjainc.com
BeogradSW2(config)#crypto key generate rsa modulus 1024
BeogradSW2(config)#ip ssh version 2
BeogradSW2(config)#line vty 0 4
BeogradSW2(config-line)#transport input ssh
BeogradSW2(config-line)#transport output ssh
BeogradSW2(config-line)#exit
BeogradSW2(config)#monitor session 1 source vlan 13 both
BeogradSW2(config)#monitor session 1 destination interface fa0/45
BeogradSW2(config)#router ospf 1
BeogradSW2(config-r)#router-id 1.1.1.12
BeogradSW2(config-r)#network 0.0.0.0 255.255.255.255 area 1
BeogradSW2(config-r)#exit
```

Novi Sad Switch 1

```
Switch3560>enable
Switch3560#configure terminal
Switch3560(config)#hostname NovisadW1
NovisadSW1(config)#vlan 21
NovisadSW1(config-vlan)#name DataVlan
NovisadSW1(config-vlan)#exit
NovisadSW1(config)#vlan 22
NovisadSW1(config-vlan)#name VoiceVlan
NovisadSW1(config-vlan)#exit
NovisadSW1(config)#vlan 23
NovisadSW1(config-vlan)#name MenagmentVlan
NovisadSW1(config-vlan)#exit
NovisadSW1(config)#vlan 900
NovisadSW1(config-vlan)#rspan vlan
NovisadSW1(config-vlan)#name RSPANVlan
NovisadSW1(config-vlan)#exit
NovisadSW1(config)#spanning-tree mode rstp
NovisadSW1(config)#spanning-tree vlan 21 priority 0
NovisadSW1(config)#spanning-tree vlan 22 priority 0
NovisadSW1(config)#spanning-tree vlan 23 priority 0
NovisadSW1(config)#spanning-tree vlan 900 priority 0
NovisadSW1(config)#vtp mode server
NovisadSW1(config)#vtp version 2
NovisadSW1(config)#vtp domain Dunjainc
NovisadSW1(config)#vtp password dunjaincpass
NovisadSW1(config)#interface port-channel 1
NovisadSW1(config-if)#no shut
NovisadSW1(config-if)#switchport mode trunk
```

```

NovisadSW1(config-if)#switchport trunk encapsulation dot1q
NovisadSW1(config-if)#channel-group 1 mode on
NovisadSW1(config)#interface range fa0/1 - fa0/8
NovisadSW1(config-if)#no shut
NovisadSW1(config-if)#exit
NovisadSW1(config-if)#channel-group 1 mode on
NovisadSW1(config-if)#exit
NovisadSW1(config)#interface fa0/9
NovisadSW1(config-if)#no switchport
NovisadSW1(config-if)#ip address 10.2.12.2 255.255.255.0
NoviasdSW1(config-if)#ip ospf hello-interval 1
NovisadSW1(config-if)#ip ospf dead-interval 3
NovisadSW1(config-if)#no shut
NovisadSW1(config-if)#exit
NovisadSW1(config)#interface range fa0/10 - fa0/35
NovisadSW1(config-if)#description Ka hostovima
NovisadSW1(config-if)#switchport mode access
NovisadSW1(config-if)#switchport access vlan 11
NovisadSW1(config-if)#spanning-tree port-fast
NovisadSW1(config-if)#switchport port-security
NovisadSW1(config-if)#switchport port-security maximum 3
NovisadSW1(config-if)#switchport port-security violation restrict
NovisadSW1(config-if)#switchport port-security mac-address sticky
NovisadSW1(config-if)#storm-control unicast level {threshold values } pps|bps
NovisadSW1(config-if)#storm-control action trap
NovisadSW1(config-if)#exit
NovisadSW1(config)#ip dhcp snooping vlan 11 - 13
NovisadSW1(config)#ip dhcp snooping information-option
NovisadSW1(config)#Interface vlan 21
NovisadSW1(config-if)#no shut
NovisadSW1(config-if)#ip address 10.2.0.1 255.255.255.0
NovisadSW1(config-if)#exit
NovisadSW1(config)#interface vlan 22
NovisadSW1(config-if)#no shut
NovisadSW1(config-if)#ip add 10.2.1.1 255.255.255.0

```

```

NovisadSW1(config-if)#exit
NovisadSW1(config)#interface vlan 13
NovisadSW1(config-if)#no shut
NovisadSW1(config-if)#ip address 10.2.2.1 255.255.255.0
NovisadSW1(config-if)#exit
NovisadSW1(config)#interface range fa0/35- fa0/40
NovisadSW1(config-if)#switchport mode access
NovisadSW1(config-if)#description Ka Lokalnim Serverima
NovisadSW1(config-if)#switchport access vlan 13
NovisadSW1(config-if)#spanning-tree port-fast
NovisadSW1(config-if)#switchport port-security
NovisadSW1(config-if)#switchport port-security maximum 3
NovisadSW1(config-if)#switchport port-security violation restrict
NovisadSW1(config-if)#switchport port-security mac-address sticky
NovisadSW1(config-if)#storm-control unicast level {threshold values } pps|bps
NovisadSW1(config-if)#storm-control action trap
NovisadSW1(config-if)#exit
NovisadSW1(config)#snmp-server location Beograd
NovisadSW1(config)#snmp-server contact Dunja Adzic
NovisadSW1(config)#snmp-server community SNMP_STRING rw
NovisadSW1(config)#snmp-server enable traps
NovisadSW1(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
NovisadSW1(config)#snmp-server system-shutdown
NovisadSW1(config)#logging on
NovisadSW1(config)#logging host 10.1.2.10
NovisadSW1(config)#logging trap 7
NovisadSW1(config)#exception protocol ftp
NovisadSW1(config)#exception dump 10.1.2.15
NovisadSW1(config)#exception corefile dunjacoredump
NovisadSW1(config)#ntp server 10.12.0.1
NovisadSW1(config)#ntp logging
NovisadSW1(config)#aaa new-model
NovisadSW1(config)#radius-server host 10.2.2.18
NovisadSW1(config)#aaa authentication login default radius local
NovisadSW1(config)#aaa authentication attempts login 3

```

```

NovisadSW1(config)#aaa authorization exec default radius
NovisadSW1(config)#aaa authorization commands default radius
NovisadSW1(config)#aaa accounting exec default start-stop
NovisadSW1(config)#aaa accounting commands default start-stop
NovisadSW1(config)#ip domain-name dunjainc.com
NovisadSW1(config)#crypto key generate rsa modulus 1024
NovisadSW1(config)#ip ssh version 2
NovisadSW1(config)#line vty 0 4
NovisadSW1(config-line)#transport input ssh
NovisadSW1(config-line)#transport output ssh
NovisadSW1(config-line)#exit
NovisadSW1(config)#monitor session 1 source vlan 23 both
NovisadSW1(config)#monitor session 1 destination remote vlan 900
NovisadSW1(config)#router ospf 1
NovisadSW1(config-r)#router-id 2.2.2.21
NovisadSW1(config-r)#network 0.0.0.0 255.255.255.255 area 2
NovisadSW1(config-r)#exit

```

Novi Sad Switch 2

```

Switch3560>enable
Switch3560#configure terminal
Switch3560(config)#hostname NovisadSW2
NovisadSW2(config)#vlan 21
NovisadSW2(config-vlan)#name DataVlan
NovisadSW2(config-vlan)#exit
NovisadSW2(config)#vlan 22
NovisadSW2(config-vlan)#name VoiceVlan
NovisadSW2(config-vlan)#exit
NovisadSW2(config)#vlan 23
NovisadSW2(config-vlan)#name MenagmentVlan
NovisadSW2(config-vlan)#exit
NovisadSW2(config)#vlan 900
NovisadSW2(config-vlan)#rspan vlan

```

```

NovisadSW2(config-vlan)#name RSPANVlan
NovisadSW2(config-vlan)#exit
NovisadSW2(config)#spanning-tree mode rstp
NovisadSW2(config)#vtp mode client
NovisadSW2(config)#vtp version 2
NovisadSW2(config)#vtp domain Dunjainc
NovisadSW2(config)#vtp password dunjaincpass
NovisadSW2(config)#interface port-channel 1
NovisadSW2(config-if)#no shut
NovisadSW2(config-if)#switchport mode trunk
NovisadSW2(config-if)#switchport trunk encapsulation dot1q
NovisadSW2(config-if)#channel-group 1 mode on
NovisadSW2(config)#interface range fa0/1 - fa0/8
NovisadSW2(config-if)#no shut
NovisadSW2(config-if)#exit
NovisadSW2(config-if)#channel-group 1 mode on
NovisadSW2(config-if)#exit
NovisadSW2(config)#interface fa0/9
NovisadSW2(config-if)#no switcport
NovisadSW2(config-if)#ip address 10.2.13.2 255.255.255.0
NovisadSW2(config-if)#ip ospf hello-interval 1
NovisadSW2(config-if)#ip ospf dead-interval 3
NovisadSW2(config-if)#no shut
NovisadSW2(config-if)#exit
NovisadSW2(config)#interface range fa0/10 - fa0/35
NovisadSW2(config-if)#description Ka hostovima
NovisadSW2(config-if)#switchport mode access
NovisadSW2(config-if)#switchport access vlan 21
NovisadSW2(config-if)#spanning-tree port-fast
NovisadSW2(config-if)#switchport port-security
NovisadSW2(config-if)#switchport port-security maximum 3
NovisadSW2(config-if)#switchport port-security violation restrict
NovisadSW2(config-if)#switchport port-security mac-address sticky
NovisadSW2(config-if)#storm-control unicast level {threshold values } pps|bps
NovisadSW2(config-if)#storm-control action trap

```

```

NovisadSW2(config-if)#exit
NovisadSW2(config)#ip dhcp snooping vlan 21 - 23
NovisadSW2(config)#ip dhcp snooping information-option
NovisadSW2(config)#Interface vlan 21
NovisadSW2(config-if)#no shut
NovisadSW2(config-if)#ip address 10.2.0.2 0.255.255.255.0
NovisadSW2(config-if)#exit
NovisadSW2(config)#interface vlan 22
NovisadSW2(config-if)#no shut
NovisadSW2(config-if)#ip add 10.2.1.2 255.255.255.0
NovisadSW2(config-if)#exit
NovisadSW2(config)#interface vlan 23
NovisadSW2(config-if)#no shut
NovisadSW2(config-if)#ip address 10.2.2.2 255.255.255.0
NovisadSW2(config-if)#exit
NovisadSW2(config)#interface range fa0/35- fa0/40
NovisadSW2(config-if)#switchport mode access
NovisadSW2(config-if)#description Ka Lokalnim Serverima
NovisadSW2(config-if)#switchport access vlan 23
NovisadSW2(config-if)#spanning-tree port-fast
NovisadSW2(config-if)#switchport port-security
NovisadSW2(config-if)#switchport port-security maximum 3
NovisadSW2(config-if)#switchport port-security violation restrict
NovisadSW2(config-if)#switchport port-security mac-address sticky
NovisadSW2(config-if)#storm-control unicast level {threshold values } pps|bps
NovisadSW2(config-if)#storm-control action trap
NovisadSW2(config-if)#exit
NovisadSW2(config)#snmp-server location Beograd
NovisadSW2(config)#snmp-server contact Dunja Adzic
NovisadSW2(config)#snmp-server community SNMP_STRING rw
NovisadSW2(config)#snmp-server enable traps
NovisadSW2(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
NovisadSW2(config)#snmp-server system-shutdown
NovisadSW2(config)#logging on
NovisadSW2(config)#logging host 10.1.2.10

```

```
NovisadSW2(config)#logging trap 7
NovisadSW2(config)#exception protocol ftp
NovisadSW2(config)#exception dump 10.1.2.15
NovisadSW2(config)#exception corefile dunjacoredump
NovisadSW2(config)#ntp server 10.12.0.1
NovisadSW2(config)#ntp logging
NovisadSW2(config)#aaa new-model
NovisadSW2(config)#radius-server host 10.1.2.18
NovisadSW2(config)#aaa authentication login default radius local
NovisadSW2(config)#aaa authentication attempts login 3
NovisadSW2(config)#aaa authorization exec default radius
NovisadSW2(config)#aaa authorization commands default radius
NovisadSW2(config)#aaa accounting exec default start-stop
NovisadSW2(config)#aaa accounting commands default start-stop
NovisadSW2(config)#ip domain-name dunjainc.com
NovisadSW2(config)#crypto key generate rsa modulus 1024
NovisadSW2(config)#ip ssh version 2
NovisadSW2(config)#line vty 0 4
NovisadSW2(config-line)#transport input ssh
NovisadSW2(config-line)#transport output ssh
NovisadSW2(config-line)#exit
NovisadSW2(config)#monitor session 1 source vlan 13 both
NovisadSW2(config)#monitor session 1 destination interface fa0/45
NovisadSW2(config)#router ospf 1
NovisadSW2(config-r)#router-id 2.2.2.22
NovisadSW2(config-r)#network 0.0.0.0 255.255.255.255 area 2
NovisadSW2(config-r)#exit
```

Niš Switch 1

```
Switch3560>enable
Switch3560#configure terminal
Switch3560(config)#hostname NišSW1
Nis SW1(config)#vlan 31
Nis SW1(config-vlan)#name DataVlan
Nis SW1(config-vlan)#exit
Nis SW1(config)#vlan 32
Nis SW1(config-vlan)#name VoiceVlan
Nis SW1(config-vlan)#exit
Nis SW1(config)#vlan 33
Nis SW1(config-vlan)#name MenagmentVlan
Nis SW1(config-vlan)#exit
Nis SW1(config)#vlan 900
Nis SW1(config-vlan)#rspan vlan
Nis SW1(config-vlan)#name RSPANVlan
Nis SW1(config-vlan)#exit
Nis SW1(config)#spanning-tree mode rstp
Nis SW1(config)#spanning-tree vlan 31 priority 0
Nis SW1(config)#spanning-tree vlan 32 priority 0
Nis SW1(config)#spanning-tree vlan 33 priority 0
Nis SW1(config)#spanning-tree vlan 900 priority 0
Nis SW1(config)#vtp mode server
Nis SW1(config)#vtp version 2
Nis SW1(config)#vtp domain Dunjainc
Nis SW1(config)#vtp password dunjaincpass
Nis SW1(config)#interface port-channel 1
Nis SW1(config-if)#no shut
Nis SW1(config-if)#switchport mode trunk
```

```

Nis SW1(config-if)#switchport trunk encapsulation dot1q
Nis SW1(config-if)#channel-group 1 mode on
Nis SW1(config)#interface range fa0/1 - fa0/8
Nis SW1(config-if)#no shut
Nis SW1(config-if)#exit
Nis SW1(config-if)#channel-group 1 mode on
Nis SW1(config-if)#exit
Nis SW1(config)#interface fa0/9
Nis SW1(config-if)#no switchport
Nis SW1(config-if)#ip address 10.3.12.2 255.255.255.0
Nis SW1(config-if)#ip ospf hello-interval 1
Nis SW1(config-if)#ip ospf dead-interval 3
Nis SW1(config-if)#no shut
Nis SW1(config-if)#exit
Nis SW1(config)#interface range fa0/10 - fa0/35
Nis SW1(config-if)#description Ka hostovima
Nis SW1(config-if)#switchport mode access
Nis SW1(config-if)#switchport access vlan 31
Nis SW1(config-if)#spanning-tree port-fast
Nis SW1(config-if)#switchport port-security
Nis SW1(config-if)#switchport port-security maximum 3
Nis SW1(config-if)#switchport port-security violation restrict
Nis SW1(config-if)#switchport port-security mac-address sticky
Nis SW1(config-if)#storm-control unicast level {threshold values } pps|bps
Nis SW1(config-if)#storm-control action trap
Nis SW1(config-if)#exit
Nis SW1(config)#ip dhcp snooping vlan 11 - 13
Nis SW1(config)#ip dhcp snooping information-option
Nis SW1(config)#Interface vlan 31
Nis SW1(config-if)#no shut
Nis SW1(config-if)#ip address 10.3.0.1 255.255.255.0
Nis SW1(config-if)#exit
Nis SW1(config)#interface vlan 32
Nis SW1(config-if)#no shut
Nis SW1(config-if)#ip add 10.3.1.1 255.255.255.0

```

```

Nis SW1(config-if)#exit
Nis SW1(config)#interface vlan 33
Nis SW1(config-if)#no shut
Nis SW1(config-if)#ip address 10.3.2.1 255.255.255.0
Nis SW1(config-if)#exit
Nis SW1(config)#interface range fa0/35- fa0/40
Nis SW1(config-if)#switchport mode access
Nis SW1(config-if)#description Ka Lokalnim Serverima
Nis SW1(config-if)#switchport access vlan 33
Nis SW1(config-if)#spanning-tree port-fast
Nis SW1(config-if)#switchport port-security
Nis SW1(config-if)#switchport port-security maximum 3
Nis SW1(config-if)#switchport port-security violation restrict
Nis SW1(config-if)#switchport port-security mac-address sticky
Nis SW1(config-if)#storm-control unicast level {threshold values } pps|bps
Nis SW1(config-if)#storm-control action trap
Nis SW1(config-if)#exit
Nis SW1(config)#snmp-server location Beograd
Nis SW1(config)#snmp-server contact Dunja Adzic
Nis SW1(config)#snmp-server community SNMP_STRING rw
Nis SW1(config)#snmp-server enable traps
Nis SW1(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
Nis SW1(config)#snmp-server system-shutdown
Nis SW1(config)#logging on
Nis SW1(config)#logging host 10.1.2.10
Nis SW1(config)#logging trap 7
Nis SW1(config)#exception protocol ftp
Nis SW1(config)#exception dump 10.1.2.15
Nis SW1(config)#exception corefile dunjacoredump
Nis SW1(config)#ntp server 10.12.0.1
Nis SW1(config)#ntp logging
Nis SW1(config)#aaa new-model
Nis SW1(config)#radius-server host 10.2.2.18
Nis SW1(config)#aaa authentication login default radius local
Nis SW1(config)#aaa authentication attempts login 3

```

```

Nis SW1(config)#aaa authorization exec default radius
Nis SW1(config)#aaa authorization commands default radius
Nis SW1(config)#aaa accounting exec default start-stop
Nis SW1(config)#aaa accounting commands default start-stop
Nis SW1(config)#ip domain-name dunjainc.com
Nis SW1(config)#crypto key generate rsa modulus 1024
Nis SW1(config)#ip ssh version 2
Nis SW1(config)#line vty 0 4
Nis SW1(config-line)#transport input ssh
Nis SW1(config-line)#transport output ssh
Nis SW1(config-line)#exit
Nis SW1(config)#monitor session 1 source vlan 23 both
Nis SW1(config)#monitor session 1 destination remote vlan 900
Nis SW1(config)#router ospf 1
Nis SW1(config-r)#router-id 3.3.3.31
Nis SW1(config-r)#network 0.0.0.0 255.255.255.255 area 3
Nis SW1(config-r)#exit

```

Niš Switch 2

```

Switch3560>enable
Switch3560#configure terminal
Switch3560(config)#hostname NišSW2
Nis SW2(config)#vlan 31
Nis SW2(config-vlan)#name DataVlan
Nis SW2(config-vlan)#exit
Nis SW2(config)#vlan 32
Nis SW2(config-vlan)#name VoiceVlan
Nis SW2(config-vlan)#exit
Nis SW2(config)#vlan 33
Nis SW2(config-vlan)#name MenagmentVlan
Nis SW2(config-vlan)#exit
Nis SW2(config)#vlan 900
Nis SW2(config-vlan)#rspan vlan

```

```

Nis SW2(config-vlan)#name RSPANVlan
Nis SW2(config-vlan)#exit
Nis SW2(config)#spanning-tree mode rstp
Nis SW2(config)#vtp mode client
Nis SW2(config)#vtp version 2
Nis SW2(config)#vtp domain Dunjainc
Nis SW2(config)#vtp password dunjaincpass
Nis SW2(config)#interface port-channel 1
Nis SW2(config-if)#no shut
Nis SW2(config-if)#switchport mode trunk
Nis SW2(config-if)#switchport trunk encapsulation dot1q
Nis SW2(config-if)#channel-group 1 mode on
Nis SW2(config)#interface range fa0/1 - fa0/8
Nis SW2(config-if)#no shut
Nis SW2(config-if)#exit
Nis SW2(config-if)#channel-group 1 mode on
Nis SW2(config-if)#exit
Nis SW2(config)#interface fa0/9
Nis SW2(config-if)#no switcport
Nis SW2(config-if)#ip address 10.3.13.2 255.255.255.0
Nis SW2(config-if)#ip ospf hello-interval 1
Nis SW2(config-if)#ip ospf dead-interval 3
Nis SW2(config-if)#no shut
Nis SW2(config-if)#exit
Nis SW2(config)#interface range fa0/10 - fa0/35
Nis SW2(config-if)#description Ka hoštovima
Nis SW2(config-if)#switchport mode access
Nis SW2(config-if)#switchport access vlan 31
Nis SW2(config-if)#spanning-tree port-fast
Nis SW2(config-if)#switchport port-security
Nis SW2(config-if)#switchport port-security maximum 3
Nis SW2(config-if)#switchport port-security violation restrict
Nis SW2(config-if)#switchport port-security mac-address sticky
Nis SW2(config-if)#storm-control unicast level {threshold values } pps|bps
Nis SW2(config-if)#storm-control action trap

```

```
Nis SW2(config-if)#exit
Nis SW2(config)#ip dhcp snooping vlan 21 - 23
Nis SW2(config)#ip dhcp snooping information-option
Nis SW2(config)#Interface vlan 31
Nis SW2(config-if)#no shut
Nis SW2(config-if)#ip address 10.3.0.2 0.255.255.255.0
Nis SW2(config-if)#exit
Nis SW2(config)#interface vlan 32
Nis SW2(config-if)#no shut
Nis SW2(config-if)#ip add 10.3.1.2 255.255.255.0
Nis SW2(config-if)#exit
Nis SW2(config)#interface vlan 33
Nis SW2(config-if)#no shut
Nis SW2(config-if)#ip address 10.3.2.2 255.255.255.0
Nis SW2(config-if)#exit
Nis SW2(config)#interface range fa0/35- fa0/40
Nis SW2(config-if)#switchport mode access
Nis SW2(config-if)#description Ka Lokalnim Serverima
Nis SW2(config-if)#switchport access vlan 33
Nis SW2(config-if)#spanning-tree port-fast
Nis SW2(config-if)#switchport port-security
Nis SW2(config-if)#switchport port-security maximum 3
Nis SW2(config-if)#switchport port-security violation restrict
Nis SW2(config-if)#switchport port-security mac-address sticky
Nis SW2(config-if)#storm-control unicast level {threshold values } pps|bps
Nis SW2(config-if)#storm-control action trap
Nis SW2(config-if)#exit
Nis SW2(config)#snmp-server location Beograd
Nis SW2(config)#snmp-server contact Dunja Adzic
Nis SW2(config)#snmp-server community SNMP_STRING rw
Nis SW2(config)#snmp-server enable traps
Nis SW2(config)#snmp-server host 10.1.2.10 traps version 2c SNMP_STRING
Nis SW2(config)#snmp-server system-shutdown
Nis SW2(config)#logging on
Nis SW2(config)#logging host 10.1.2.10
```

```
Nis SW2(config)#logging trap 7
Nis SW2(config)#exception protocol ftp
Nis SW2(config)#exception dump 10.1.2.15
Nis SW2(config)#exception corefile dunjacoredump
Nis SW2(config)#ntp server 10.12.0.1
Nis SW2(config)#ntp logging
Nis SW2(config)#aaa new-model
Nis SW2(config)#radius-server host 10.1.2.18
Nis SW2(config)#aaa authentication login default radius local
Nis SW2(config)#aaa authentication attempts login 3
Nis SW2(config)#aaa authorization exec default radius
Nis SW2(config)#aaa authorization commands default radius
Nis SW2(config)#aaa accounting exec default start-stop
Nis SW2(config)#aaa accounting commands default start-stop
Nis SW2(config)#ip domain-name dunjainc.com
Nis SW2(config)#crypto key generate rsa modulus 1024
Nis SW2(config)#ip ssh version 2
Nis SW2(config)#line vty 0 4
Nis SW2(config-line)#transport input ssh
Nis SW2(config-line)#transport output ssh
Nis SW2(config-line)#exit
Nis SW2(config)#monitor session 1 source vlan 13 both
Nis SW2(config)#monitor session 1 destination interface fa0/45
Nis SW2(config)#router ospf 1
Nis SW2(config-r)#router-id 3.3.3.32
Nis SW2(config-r)#network 0.0.0.0 255.255.255.255 area 3
Nis SW2(config-r)#exit
```

11. Zaključak

Projekat obrađen u ovom master radu omogućava sve uslove za kvalitetan, skalabilan i siguran rad mreže.

Svi uređaji korišćeni u datom mrežnom rešenju su visoko kompatibilni sa visokim zahtevima mreže i predstavljaju poslednju generaciju Cisco mrežnih uređaja. Praksa je pokazala da su korišćeni uređaji idealni za SMB (*small to medium business*) rešenja u šta spadaju i zahtevi ovog projekta. Uređaji korišćeni u mrežnom rešenju pružaju visok stepen pouzdanosti, sigurnosti skalabilnosti mreže i podržavaju sve tražene funkcionalnosti.

Protokoli za uspostavljanje komunikacije u mreži su standardizovani i opšte prihvaćeni. OSPF koji je implementiran kao protokol za razmenu rutin informacija u mreži je protokol otvorenog koda visoko skalabilan, pouzdan sa veoma kratkim vremenom konvergencije. U slučaju pada ili otkaza bilo kog mrežnog elementa OSPF preuzima logičku odgovornost na sebe i veoma brzo uspostavlja nesmetan rad mreže u novonastalim uslovima. Kao jako skalabilan protokol omogućava potencijalno brzo nadograđivanje kompletnog mrežnog rešenja informacionog sistema. Zbog visokih zahteva bezbednosti mreže implementirana je zaštita razmena rutin informacija odnosno OSPF LSA paketa. OSPF je implementiran na nivou više OSPF area zbog trenutne veličine mrežnog rešenja čime je olakšana administracija mrežnom administratoru. Zbog potrebe za skalabilnosti cele mreže na nivou rutiranja, takođe je implementirana je sumarizacija ruta.

Na drugom mrežnom nivou OSI modela implementiran je opšte prihvaćen 802.1Q standard. Interfejsi između switcheva su postavljeni u Etherchannel mod rada čime se obezbeđuje ravnomerna raspodela saobraćaja u mreži i čime se povećava propusni opseg između switcheva. Etherchannel interfejsi rade u trunk modu kako bi komunikacija između različitih VLAN-ova bila moguća. Protokol za prevenciju petlji u mreži je 802.1W (rapid spanning tree). U praksi se 801.W pokazao kao protokol sa najvećom brzinom konvergencije i velikom mogućnošću za proširenjem L2 domena. Ovime je omogućen nesmetan rad LAN mreže.

Obzirom da je u datom projektu tražen visok stepen bezbednosti i zaštite implementiran je veliki broj zaštitnih mehanizama u mreži. Izabrani modeli Cisco uređaja podržavaju sve zahtevane protokole za bezbednost i veoma ih efikasno procesiraju. Za zaštitu komunikacije između samih lokacija u mreži, implementirani su IPSec protokoli. Enkripcija saobraćaja je implementirana ESP protokolom koji koristi AES 256 enkripciju. Autentikacija samih rutera (lokacija) je izvedena uz pomoć PSK (Pre-Shared Key) arhitekture. PSK arhitektura se uspostavlja uz pomoć predefinisanih statičkih kriptoloških ključeva. Za zaštitu ulaznog eksternog saobraćaja u mrežu konfigurisan je CBAC firewall koji prati ispravnost UDP, ICMP i TCP konekcija. CBAC takođe vrši praćenje protokola na višem nivou OSI modela. Za kontrolu pristupa korišćene su Acces-liste koje pružaju statičku zaštitu mreži. Zbog administrativnog i menadžment pristupa uređajima implementiran je AAA model. AAA model predstavlja integrisanu zaštitu pristupa uređajima. Podržana je autentikacija, autorizacija i accounting, čime se prati i obezbeđuje svaki rad na uređaju. Zbog ove potrebe, AAA direktno komunicira sa centralizovanim RADIUS serverom.

Za potrebe bezbednosti na L2 nivou implementiran je veliki broj neophodnih mehanizama. Ovime se obezbeđuje siguran rad u LAN mreži kao i sigurnost LAN protokola.

Praćenje samog LAN saobraćaja biće vršeno RSPAN protokolom koji sve podatke šalje na centralizovane servere, koji su instalirani u okviru svakog posebnog LAN-a.

U datom mrežnom rešenju neophodno je praćenje, beleženje i menadžment svih relevantnih događaja. Protokoli koji će vršiti te funkcije su SNMP, NTP, Logging i Core Dump. SNMP je standardizovan protokol koji se koristi za menadžment i upravljanje mrežnim elementima, kojim se manipuliše preko centralizovanog servera na kome je instaliran SNMP menadžer softver. SNMP skuplja SNMP trap poruke koje se šalju na server. Obezbeđena je zaštita u komunikaciji između SNMP klijenata i SNMP menadžera. Korišćena verzija SNMP-a je v2c. Takođe implementirani server za skupljanje log poruka koje šalju Cisco uređaji NTP implementiran radi sinhronizacije vremena u mreži. Uređaji će se vremenski sinhronizovati sa eksternim NTP serverom na internetu. U slučaju otkaza neke od korišćenih hardverskih platformi, Core Dump šalje poslednji presek stanja uređaja na centralizovani server. Implementacijom svih spomenutih menadžment i administrativnih protokola uspostavlja se efikasno, brzo i olakšano rukovođenje kompletne mrežne infrastrukture. Takođe je obezbeđen kvalitetan rad mreže i zahtevani stepen kvaliteta saobraćaja u mreži. Ovo se postiže konfiguracijom polisa kvaliteta servisa. Ove polise mreži garantuju željen protok saobraćaja i garanciju određenim aplikacijama koje će se koristiti u mreži.

Projekat kompletno integriše razmenu data i voice saobraćaja. Implementirana unificirana Cisco arhitektura koja predstavlja realizaciju kompletnog voice rešenja. Korisnici mreže će glasovno komunicirati uz pomoć Cisco IP telefona poslednje generacije. Uređaji koji kontrolišu uspostavu poziva i praćenje kontrole rada protokola za voice komunikaciju su centralno integrisani. Sve funkcije i potrebe za jedno ovakvo kvalitetno rešenje su podržane od strane Cisco uređaja.

Predstavljeno mrežno rešenje i mrežni dizajn napisan je po najsavremenijim standardima u svetu komunikacionih tehnologija.

Literatura

- [1] Dragan Pleskonjić, Nemanja Maček, Borislav Đorđević, Marko Carić :''Sigurnost računarskih sistema i mreža'', Beograd 2007.
- [2] Markus Jakobsson, Moti Yung, Jianying Zhou: ''Applied Cryptography and Network Security'' USA 2004
- [3] Raymond Blair, Arvind Durai: ''Cisco Secure Firewall Services Module (FWSM)'', Indianapolis 2008.
- [4] William Stallings: ''Cryptography and Network Security'', USA 2005
- [5] Michael Gregg, Stephen Watkins, George Mays, Chris Ries, Ron Bandes, Brandon Franklin: ''Hack the Stack using snort and ethereal to master the 8 layers of an insecure network'' Rockland 2006.
- [6] Eric Cole: ''Hackers beware'', USA 2002.
- [7] John A. Vacca: ''Network and System Security'', Burlington 2009.
- [8] Dr. Eric Cole, Dr. Ronald Krutz, and James W. Conley: ''Network Security Bible''.
- [9] Andrew Lockhart: ''Network Security Hacks'', 2006
- [10] Jazib Frahim, Qiang Huang: ''SSL Remote Access VPNs'', Indianapolis 2008.
- [11] Nong Ye: ''Secure Computer and Network Systems Modeling, Analysis and Design'', Arizona State University, USA 2008.
- [12] Stuart McClure, Joel Scambray, George Kurtz: ''Hacking Exposed 6: Network Security Secrets & Solutions'', USA 2009.
- [13] Vijay Bollapragada, Mohamed Khalid, Scott Wainner: ''IPSec VPN Design'', Indianapolis 2005.
- [14] James Henry Carmouche: ''IPSec Virtual Private Network Fundamentals'', Indianapolis 2006.
- [15] Joe Harris: ''Cisco Network Security Little Black Book'', Scottsdale, Arizona 2002.
- [16] Indranil Bose: ''Breakthrough Perspectives in Network and Data Communications Security, Design, and Applications'', USA 2009.
- [17] Weijia Jia, Wanlei Zhou: ''Distributed Network Systems'', Boston, USA 2005.