INTERNET TEHNOLOGIJE

PACKET TRACER KOMANDE ZA LAB. VJEŽBE

2023/24

SADRŽAJ

1. Lab 1- STP, VLAN-ovi

1.1. STP – Spanning Tree Protocol

- **S(config)#spanning-tree vlan 1 root primary** - postavljanje svica na ROOT BRIDGE

ILI

S(config)#spanning-tree vlan 1 priority <0-61440> u inkrementima od 4096

- S#show spanning tree - uvidjaj u info o STP-u (interfejsi te statusi o njima i svojstva)

- S0(config)#int fa0/1

- **s(config-if)# spanning-tree portfast** - ubrzava STP proces na ACCESS PORTU (gdje je PC povezan, nikako ne treba koristi na trank vezama). Ovo ide u KOMBINACIJI sa komandom…

s(config-if)# spanning-tree bpduguard enable

- **s0(config-if)#spanning-tree vlan 1 port-priority <0-240>** - Daje prioritet interfejsima u slucaju da sa svica ide vise veza ka drugim i ne zelimo da neki bude blokiran u STP procesu (manji broj = visi prioritet). Inkrementi od 16 (16, 32, 48, …)

1.2. VLAN – Virtual Local Area Network

**** SWITCH-EVI ****

- s#show vlan brief -> pokazuje koji interfejsi pripadaju pojedinim vlan-ovima

- **s(config)#vlan 10** -> kreiranje vlana 10

- s(config-vlan)#**name** Studenti -> davanje imena 'Studenti' vlanu 10

- **s(config-if)#switchport mode access** -> konfigurisanje interfejsa svica na ACCESS port (PC)

- **s(config-if)#switchport access vlan 10** -> pridruzivanje nekog ACCESS interfejsa (obično se radi na access linkovima) nekom vlanu (10)

- s(config)#interface range fa0/1-n -> prelazak u opsežni rezim pridruzivanja interfejsa 1-n (n broj do nekih interfesa, npr. fa0/1-4 = konfigurisanje interfejsa 1 do 4)

- s(config-if-range)#switchport mode trunk -> konfigurisanje interfejsa svica na TRUNK port (SWITCHEVI-HUB-RUTER veze)

- **s(config-if)#switchport trunk native vlan n** -> stavljanje porta na native vlan broja 'n'

**** RUTERI *****

- R0(config-if)#no shutdown

- R0(config)#int g0/0.n -> ulaz u podinterfejs rutera sa brojem 'n'

- **R0(config-if)#encapsulation dot1q n** <native> -> postavljanje podinterfejsa da radi sa IEEE802.1Q u VLAN-u 'n'. Tj. omogućavanje inter-vlan routing-a! Opcionalno se može staviti **native** na kraju da se pridruži nativan vlan nekom podinterfejsu

2. Lab 2- EtherChannel, WLAN setup, PPP konfigurisanje

2.1. MLS – Multi Layer Switch

L3 switch se može koristiti i za L2 svičing i za L3 rutiranje. Neke komande za konfiguraciju L3 switch-a su sljedeće:

……

Konfigurisanje L3 svičinga

……

- MLS(config)# int g0/0
- **MLS(config-if)# no switchport** -> konfigurisanje porta L3 sviča za rutiranje (routed port)
- MLS(config-if)# ip address <adresa n.n.n.n> <maska n.n.n.n>

……

Konfigurisanje Inter-VLAN routing-a

……

Kao i na L2 sviču, moramo dodati VLAN-ove na L3 svič i aktivirati tzv. **SVI (Switch Virtual Interface)** interfejse.

- MLS(config)# vlan <n> -> kreiranje VLAN-a na sviču (može se eventualno još dati i ime)
- **MLS(config)# int vlan <n>** -> Moramo ući u interfejs da ga konfigurišemo
- **MLS(config-if)# ip add <adresa n.n.n.n> <maska n.n.n.n>**

Konfigurisanje trunkova na L3 sviču je malo drugačije, mora se koristi dot1q enkapsulacija kao na ruteru:

- **MLS(config-if)# switchport mode trunk** -> postavljanje statičkog trunk interfejsa
- MLS(config-if)# switchport trunk native vlan <n> -> označavanje nativnog VLAN-a
- **MLS(config-if)# switchport trunk encapsulation dot1q** -> označavanje interfejsa za rutiranje sa do1q protokolom
- **MLS(config)# ip routing** -> omogućavanje RUTIRANJA

2.2. EtherChannel

PREDUSLOV: interfejsi **moraju biti istog tipa** (svi 100Mbp ili svi 1Gb)

- **s(config-if)#channel-group <1-6> mode <on/active/auto/desirable/passive>** -> omogućavanje EtherChannel opcije na sviču za neki interfejs sviča (ili opseg -> int xx0/1-n)

MODOVI:

---

active LACP is enabled unconditionally

passive LACP is enabled only if another LACP-capable device is connected.

desirable PAgP is enabled unconditionally

auto PAgP is enabled only if another PAgP-capable device is connected.

on EtherChannel is enabled, but without either LACP or PAgP.

*** Komunikacija je jedino moguca izmedju sviceva koji imaju active-active/active-passive ILI desirable-desirable/desirable-auto, sto se tice za te komande

- **s(config)# interface port-channel <1-6>** -> ulazak u konfiguraciju porta na kanalima 1-6 (npr. postavljanje na trunk/access sa switchport mode trunk/access)

- **s#show etherchannel summary**-> daje listu svih etherchannel-a koji napravljeni na nekom svicu

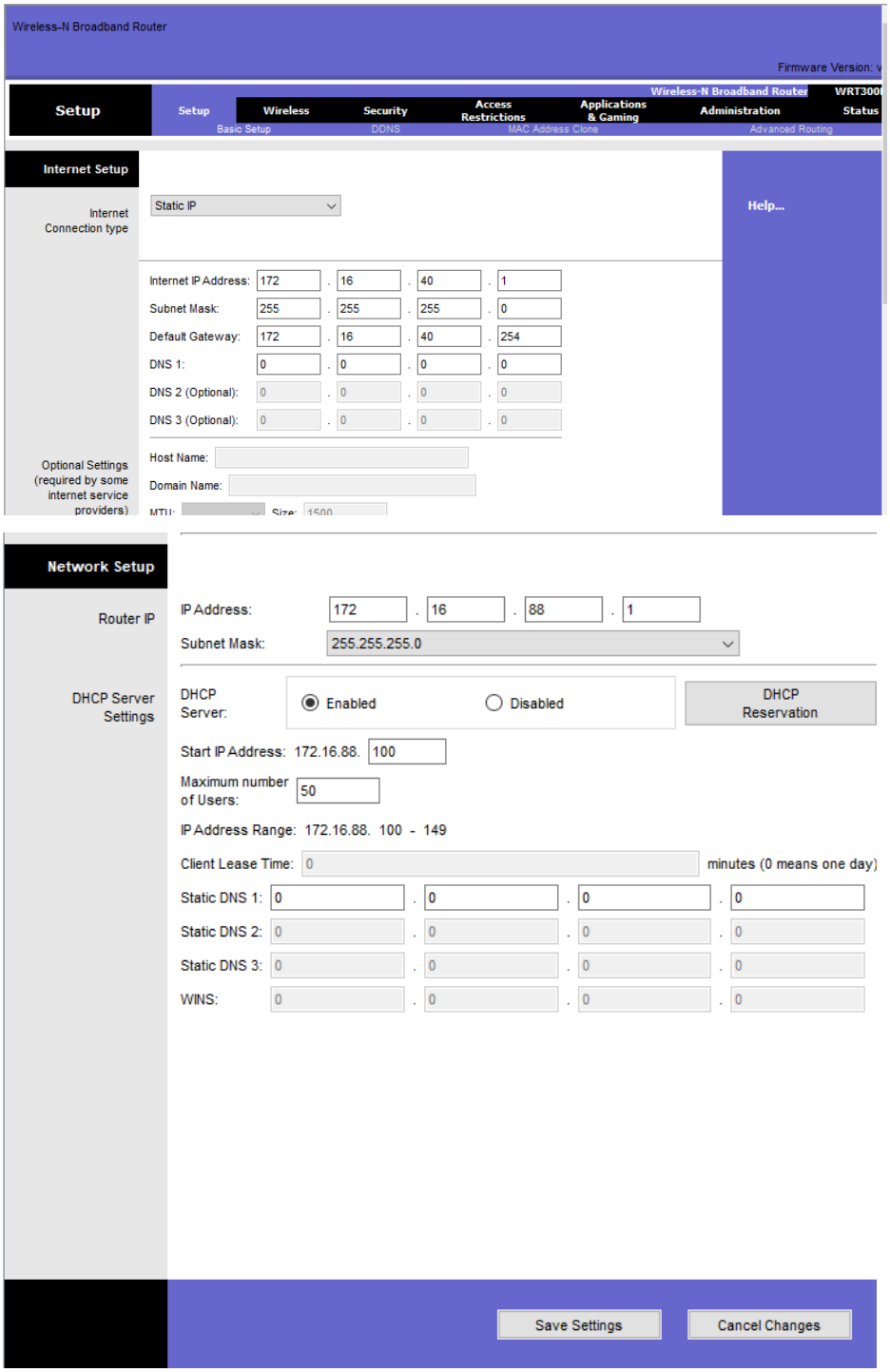
2.3. WLAN konfigurisanje

Svaki WLAN ruter (Wireless Router0 – Home Router) ima (uglavnom) 5 portova, od kojih jedan je **internet** (WAN) interfejs koji omogućava povezivanje na internet (vanjsku mrežu). Ostala 4 porta su za LAN mrežu na koju povezujemo uređaje. Wireless ruter dakle, ima 2 mreže.

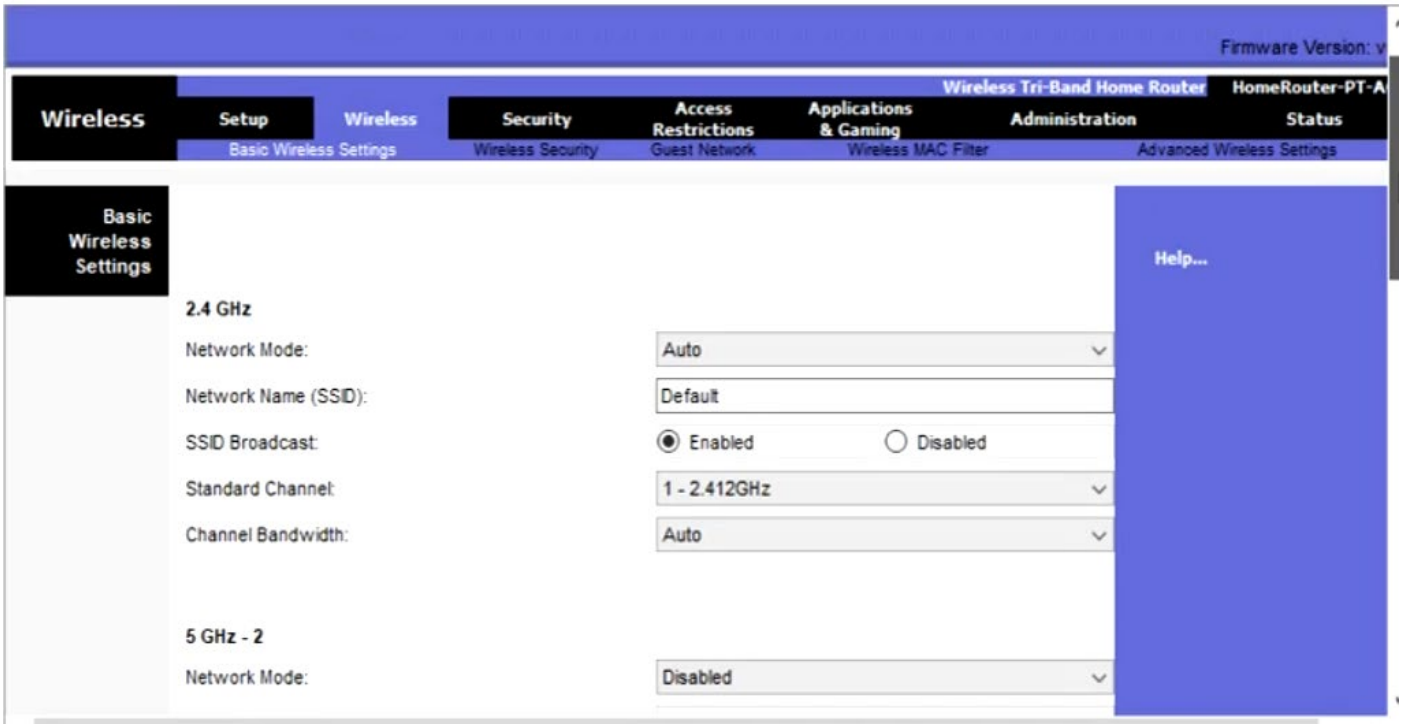
Podešavanje Wireless rutera se vrši iz njegovog GUI-a, koji ima 2 sekcije – **Internet Setup** i **Network Setup**. OBAVEZNO poslije svake izmjene treba pritisnuti dugme na kraju prozora **Save Settings** kako bi se promjene sačuvale.

U **Internet Setup-u**, po default-u se adresa internet interfejsa dobija preko DHCP-a od ISP-a, za ručni unos adrese na tom interfejsu potrebno je izabrati padajući meni i izabrati Static IP opciju, te unijeti potrebne adrese (IP adresa interfejsa, maska, default gateway).

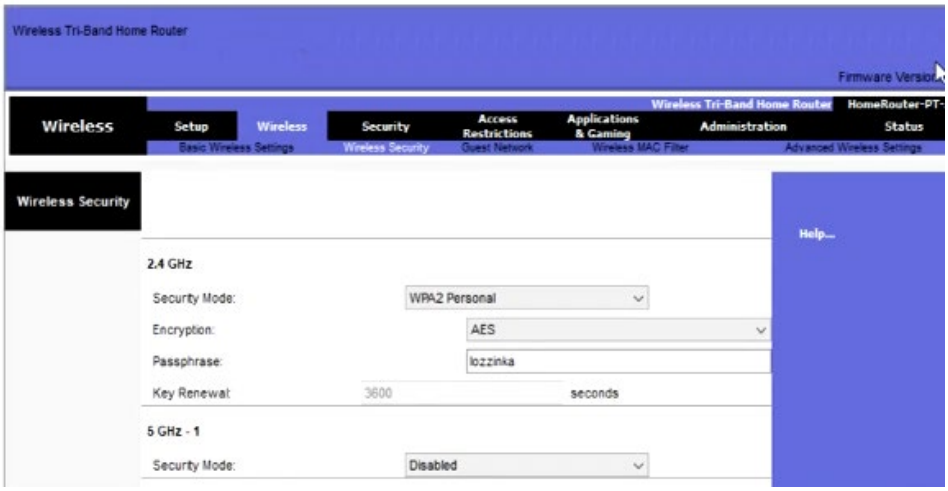
U **Network Setup-u** podešavamo interfejs rutera koji je u LAN-u gdje su i uređaji, pa voditi računa da se poklapaju IP adrese za taj dio mreže. Po default-u IP adresa može biti 192.168.0.1 i maska 255.255.255.0, ali može se i to naravno promijeniti (sekcija Router IP). U ovom dijelu je moguće podešavati i DHCP opcije (Wireless Ruter je i DHCP server i switch/AP) u sekciji DHCP Server Settings.



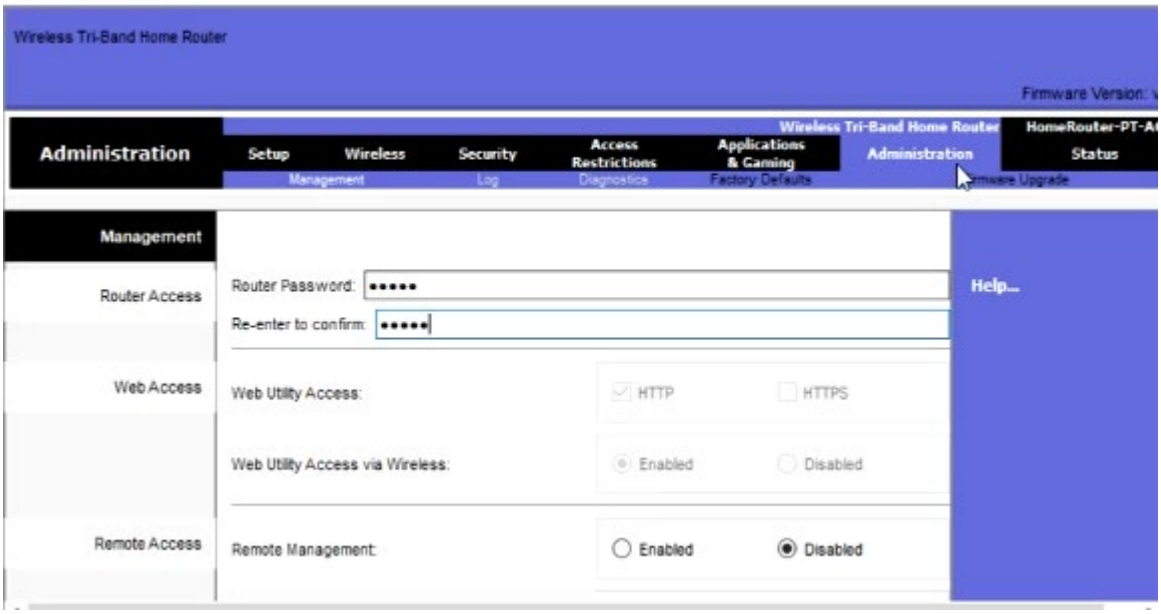
2. korak pri podešavanju rutera jeste da podesimo sama podešavanja za bežičnu komunikaciju. To radimo prelaskom u tab **Wireless**. Ovdje podešavamo ime mreže (SSID) i na kojim frekvencijama će raditi (2.4GHz ili 5Ghz ili oboje).



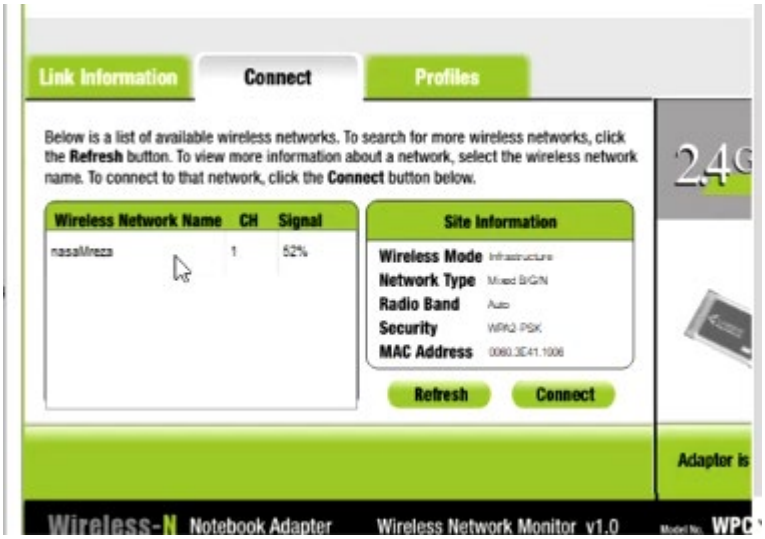
Dalje moramo podesiti <u>LOZINKU</u> za wireless mrežu, što radimo u pod-tabu **Wireless Security** (ne zaboraviti Save Settings!). Potrebno je za naš 2.4GHz frekvenciju iz primjera izabrati iz padajućeg menija za Security Mode <u>WPA2 Personal</u> (za kućne mreže, za poslovno okruženje može <u>Enterprise</u>). U sljedećoj sekciji koja se pojavi, izabrati AES kao algoritam enkripcije i postaviti lozinku.



Još jedna sekcija je pod tabom **Administration** gdje se podešava lozinka za pristup samom ruteru sa nekog računara.



Da bi neki računar/laptop pristupio ruteru, mora da ima wireless mrežnu karticu. Ako je nema, potrebno je otići u tab Physical -> WPC300N karticu ubaciti (prvo isključiti računar/laptop). Poslije toga otići u tab Desktop od računara i naći opciju **PC Wireless**, pa u tom prozoru izabrati tab **Connect**, po potrebi uraditi **Refresh** i kliknuti na postojeću wireless mrežu i povezati se sa **Connect** dugmetom. Mreža će tražiti lozinku za pristup, unijeti onu koju smo konfigurisali u ruteru.



2.4. PPP, ruteri

- **R1(config-if)#encapsulation ppp** -> postavlja interfejs rutera da radi na PPP protokolu (MORA SE URADITI I SA JEDNE I DRUGE STRANE, tj. i na 1. i na 2. ruteru!)

- **R1(config-if)#no encapsulation ppp** -> uklanja PPP mod, vraća na HDLC protokol

- **R1(config)#username <hostname_drugog_povezanog_rutera> password <lozinka>** -> postavlja autentifikaciju PPP-a na rutere. Navesti hostname drugog rutera na koji je povezan R1 (npr. R2) i lozinka mora biti usaglašena na R2 takodje. Na R2 ruteru uraditi isto samo staviti za hostname R1!

- **R1(config-if)#ppp authentication chap** -> postavljanje autentifikacije na interfejs rutera (MORA I SA JEDNE I DRUGE STRANE, NA OBA RUTERA)

3. Lab 3- RIP, SSH, Port security

3.1. RIP – Routing Information Protocol

RIPv1

.....

- **r(config)#router <rip/bgp/eigrp/ospf>** -> ako izaberemo <rip>, ulazimo u konfiguracioni mod samog rip protokola na ruteru.

- r(config-router)#network ... -> glavna komanda u ovom podmodu (gdje se unose direktno povezane mreže i tako oglašavaju drugim ruterima)

- **r(config-router)#network <n.n.n.n>** -> unos IP adrese (direktno povezane mreže)

- **r# show ip route** -> prikaz routing tabele na ruteru

RIPv2

.....

- **r(config-router)#version 2** -> prelazak u RIPv2 verziju protokola. Ali ovo nije dovoljno iz legacy razloga, jer se vrsi automatska sumarizacija - isto sto radi i RIPv1. Zato kucamo komandu ispod...

- **r(config-router)#no auto-summary** -> sada oglasavamo mrezu kakvu jeste

- **r(config-router)#passive-interface <naziv_interf>** -> oznacavanje interfejsa rutera da NE ŠALJE RIP pakete na taj interfejs rutera zbog optimizacije (jer svicevi/PC-evi ne razumiju RIP pakete i ne treba da ih dobijaju). Može se koristiti i za OSPF protokol rutiranja, vidjeti sekciju 4.

3.2. SSH – Secure Shell

.....
SSH - basic
.....

- **r(config)#no ip-domain lookup** -> sprjecavanje IOS-a da razrjesava ime pogresne komande (ako se ukuca "komanda" npr. 'dsadsadas' u CLI, IOS ce to pokusati razrjesiti sto naravno nece uspjeti i gubimo vrijeme na tome)

- r(config)#security password min-length <n> -> postavljanje minimalne duzine lozinke IOS-a na 'n' karaktera

- **r(config)#exec-timeout <n>** -> postavljanje tajmera na otvorenu sesiju IOS-a na nekom uredjaju (broj 'n' dat u MINUTAMA)

- r(config)#enable secret <lozinka> -> postavljanje enkriptovane lozinke u neki mod (ovdje config mod tj. privileged EXEC)

- **r(config)#service password encryption** -> enkriptovanje svih lozinki

.....

SSH - koraci/RUTER (vrijedi i za switch!)

.....

- **r(config)#username <ime_korisnika> secret <lozinka_korisnika>** -> podesavanje korisnika za SSH sesiju

- **r(config)#ip domain-name <ime_domena>** -> postavljanje domena kome pripada korisnik

- **r(config)#crypto key generate rsa** -> generisanje RSA kljuceva pomocu kojih ce se kriptovati SSH komunikacija (uzeti 1024 kao prompt sljedeci)

- r(config)#line vty 0 4

- **r(config-line)#transport input <ssh/telnet/all/none>** -> postavljanje VTY linija da koriste neki ili oba od SSH/Telnet protokola (kucati 'ssh' za omogucavanje samo SSH konekcije)

- **r(config-line)#login local** -> pozivanje na lokalnu bazu korisnika kreiranu od maločas (komanda username <...> secret <...>')

- **r(config)#login block-for <a> attempts <b> within <c>**-> blokiranje brute-force napada na lozinku. Tj. onemogućuje se unos lozinke na 'a' sekundi ako se uradi 'b' pokusaja u prozoru od 'c' sekundi

.....

SSH - koraci/SWITCH

.....

- **s(config)#interface range <prvi, drugi, treci, ...>** -> ulazak u interfejse da bi ih ugasili (ako je neki napadac kojim slucajem vec bio povezan na njih)

- s(config-if-range)#shutdown -> gašenje interfejsa

- **s(config)#int vlan 1**

- **s(config-if)#ip address <n.n.n.n - adresa> <n.n.n.n - maska>** -> da switch moze primati konekcije iz mreže

- s(config-if)#no shutdown

- **s(config)#ip default-gateway <n.n.n.n>**

3.3. Port security

- **s(config-if-range)#switchport port-security** -> enable-ovanje port security-a na nekom portu switch-a (BEZ PARAMETARA, prva komanda koja se kuca)

- **s(config-if-range)#switchport port-security maximum <n>** -> postavljanje maksimalnog broja uredjaja koji mogu pristupiti nekom interfejsu

- **s(config-if-range)#switchport port-security mac-address sticky** -> MAC adrese uredjaja su dinamicki naucene

- **s(config-if-range)#switchport port-security violation restrict** -> portovi se NEĆE ISKLJUČITI kada se desi konflikt (javlja se notifikacija)

protect Security violation protect mode

restrict Security violation restrict mode

shutdown Security violation shutdown mode

- s**# show port-security address** -> prikaz svih sigurnih mac adresa tabelarno

4. Lab 4- OSPF – Open Shortest Path First

- **r(config)# router ospf <broj_ospf_procesa, 1-65535>** -> staviti **"1" tj. "sve"** zbog dobre prakse

- **r(config-router)# network** <ip_adresa_MREŽE> <WILDCARD_maska> **area** <a, 0-INT_MAX> -> dodavanje mreze (adresa n.n.n.n + maska n.n.n.n) u routing tabelu. Takodje dodavanje oblasti kojoj pripada ruter, ako se ne definise neka nova oblast (area) po default-u se uzima BACKBONE area a=0. 2 rutera ce biti susjedi samo ako im se POKLAPAJU area-e.

*** **WILDCARD MASKA** **** - to je maska koja ima **inverzan oblik** u odnosu na korištenu masku u mreži, tako da ukupan zbir okteta bude = 255.

Primjer 1: 255.255.255.0 (/24) bi imala wildcard masku 0.0.0.255

Primjer 2: 255.255.248.0 (/21) bi imala wildcard masku 0.0.7.255

*** **LOOPBACK INTERFEJS** *** -> logički/softverski interfejs, koji NE MOŽE BITI UGAŠEN (down). Može simulirati neku mrežu!

- **r(config)# interface loopback <n>** -> kreiranje loopback interfejsa sa brojem n = 0-INT_MAX/2

- **r(config-if)# ip add <adresa> <maska>** -> davanje adrese/mreze loopback interfejsu

- **r(config)# ip route 0.0.0.0 0.0.0.0 loopback <n>** -> stavljanje default-ne rute na ruteru da gađa loopback interfejs

- **r(config-router)# default-information originate** -> spustanje/slanje defaultne rute svim ostalim ruterima u IP domenu (umjesto rucnog kucanja defaultne rute na svakom ruteru)

*** LSA 1,2 - IntraArea (unutar oblasti) ***

*** LSA 3,4 - InterArea (izmedju oblasti) ***

*** LSA 5 - ExternalArea (van oblasti) ***

Ako postoji veliki broj mreža u nekoj oblasti, može se izvršiti sumarizacija ruta. Umjesto ručnog dodavanja pojedinačnih mreza, možemo dodati jednu sumarnu adresu kao rutu:

- **r(config-router)# area <n> range <ip_adresa> <PRAVA_maska>** -> dodavanje sumarne rute sa sumarnom ip adresom i maskom (NIJE WILDCARD maska)

*** **MULTIACCESS MREŽA** - imamo više rutera koji dijele jednu mrežu (preko jednog switch-a npr.) ***

Ovdje se ruteri dogovaraju ko će biti glavni ruter (**Designated router** - **DR**) u OSPF-u, svi će uspostaviti susjedstvo sa njim i sve LSA poruke ce ići preko njega. Da se DR ne bi preopteretio (bio usko grlo), bira se **Backup DR** (**BDR**) ako glavni ispadne iz priče. Parametri pri izboru DR-a (od najvišeg ka najmanjem):

1. **router-id** - moze biti bilo kakav 32b podatak, ali uglavnom je u dotted-decimal notaciji (1.1.1.1, 2.2.2.2, ...)

2. **loopback interfejs** - ruter uzima NAJVECI loopback interfejs (po bilo kojoj IP ADRESI). Npr. 192.168.x.x > 172.16.x.x

3. **fizicki interfejs** - ruter uzima NAJVECI fizicki interfejs (po bilo kojoj IP ADRESI). Npr. 192.168.x.x > 172.16.x.x

- r# show ip ospf neighbor -> pregled multiaccess ospf rutera

- **r(config-if)# ip ospf priority <0-255>** -> postavljanje prioriteta nekog interfejsa u OSPF procesu (da bude npr. DR). Veci broj = veci prioritet. Međutim ne tretira se odmah, mora se opet pokrenuti mreža za nove izbore OSPF DR-ova, tj. ugasiti i upaliti interfejse na switch-u.

5. Lab 5 – DHCP, NAT

5.1. DHCP – Dynamic Host Configuration Protocol

*** Prvi korak pri kreiranju DHCP servisa na nekom ruteru (R2) jeste kreiranje DHCP pool-ova, onoliko koliko ima mreža kojima treba IP adresa. Takođe, moramo i neke adrese isključiti iz pool-a koje se ne trebaju dodijeliti, npr. default-gateway. ***

- r2(config)#ip dhcp excluded-address <1_adresa n.n.n.n> <2_adresa n.n.n.n> -> isključuje opseg adresa

- **r2(config)#ip dhcp pool <ime>** -> kreiranje DHCP pool-a po nekom imenu.

- **r2(dhcp-config)#network** <mrezna_adresa> <maska> -> dodavanje adrese u DHCP pool

- **r2(dhcp-config)#default-router** <default_gateway_mreze> -> postavlja defaultni gateway pool-a za odg. mrežu (interfejs rutera koji treba da bude DG za tu mrežu gdje je PC)

- **r2(dhcp-config)#dns-server** <adresa_dns_servera> -> postavlja adresu DNS servera za taj pool

*** Kada neki PC bude tražio adresu, on će slati broadcast poruku kao zahtjev, ali samo u svojoj mreži naravno. Tako da ruter na koji je povezan taj PC mora da pretvori tu broadcast poruku u unicast i šalje na interfejs rutera koji glumi DHCP server. Dakle, moramo da konfigurišemo te rutere (R1 ovdje) da budu **DHCP Relay Agenti** ***

- r1(config-if)#ip helper-address <adresa_DHCP_servera> -> omogućavanje DHCP relay-a na nekom ruteru da proslijedi svoj LAN saobraćaj ka interfejsu DHCP servera (može biti adresa interfejsa rutera koji glumi DHCP server ili adresa stvarnog servera)

*** Kucati ipconfig /renew ako dodje do greške prilikom slanja parametara nekom hostu od DHCP servera ***

- **r2(config-if)# ip address dhcp** -> DHCP dodjeljuje adresu nekom interfejsu rutera (ako je npr. povezan na CLOUD)

5.2. Statički NAT – Network Adress Translation

Ovaj tip prevođenja je 1:1 prevođenje, tako da samo jedan uređaj može da ima pravo da se prevede u SAMO JEDNU javnu adresu (ili neku drugu). Svaka konf. NAT-a se radi na RUTERU.

- **r1(config)#ip nat inside source static** <sta_se_prevodi n.n.n.n> <u_sta_se_prevodi n.n.n.n> -> kreiranje statičke NAT translacije za mapiranje adrese udaljene mreže na vanjsku mrežu. Ali nije još gotovo, potrebno je AKTIVIRATI interfejse!

- **r1(config-if)#ip nat inside** -> inside interfejs je uvijek gdje **ULAZI PRIVATNI** saobraćaj

r1(config-if)#ip nat outside -> outside interfejs je uvijek gdje **IZLAZI JAVNI** saobraćaj

Komanda **ip nat inside** može da se koristi više puta, ALI na **RAZLIČITIM INTERFEJSIMA**, tako da ako neki ruter ima više povezanih mreža, mora se otići na svaki interfejs gdje je odgovarajuća mreža i kucati tu komandu.

- r1#show ip nat translations -> prikaz tabele NAT-ovanja na ruteru

5.3. Dinamički NAT

*** Standardna/pristupna control lista *** - jednostavan zapis koji će da obuhvati sve adrese sumarno (**Access Control List** - **ACL**)

- **r2(config)#access-list <n> permit** <sumarna_adresa> <wildcard_maska> -> dodavanje privatnih adresa sumarno (opseg) u ACL listu

- **r2(config)#ip nat pool** <naziv> <javna_start_adresa> <javna_end_adresa> **netmask** <stvarna_maska_javnog_linka> -> postavljanje pool-a javnih adresa u koje će se prevesti privatne adrese

Npr. ip nat pool POOL 209.165.200.229 209.165.200.230 netmask 255.255.255.224

- **r2(config)#ip nat inside source list** <n broj_liste> **pool** <naziv_poola>-> povezuje kreiranu ACL listu privatnih adresa sa pool-om javnih adresa

- r2(config-if)#ip nat inside

r2(config-if)#ip nat outside

*** Problem kod ovog konfigurisanja jeste što moramo omogućiti dovoljan broj javnih adresa u koje će se prevesti sve privatne adrese, jer u suprotnom ako omogućimo manje ruter će rezervisati sve javne adrese u koje se mogu prevesti za one hostove koji su već to tražili. Tako da ako neki treći host bude tražio pristup internetu/javnoj adresi, neće to moći da uradi (kao u primjeru gore gdje smo naveli 2 javne adrese u pool-u 229-230, treći host neće imati pristup internetu). ***

- R(config)# **ip access-list standard** <ime> -> konfigurisanje STANDARDNE ACL liste koja će omogućiti više uključenih adresa

- R(config-std-nacl)# **permit** <adresa_mreže> <wildcard_maska> -> davanje pristupa ACL listi bilo kojem hostu u mreži navedenoj sa <adresa_mreže>

5.4. PAT – Port Adress Translation

- r2(config)#access-list <n> permit <sumarna_adresa> <wildcard_maska>

- r2(config)#ip nat pool <naziv> <javna_start_adresa> <javna_end_adresa> netmask <stvarna_maska_javnog_linka>

- **r2(config)#ip nat inside source list** <n broj_liste> **pool** <naziv_poola> **overload** -> omogućavanje PAT-a sa 'overload'

- r2(config-if)#ip nat inside -> 1. interfejs (1. LAN mreža)
  r2(config-if)#ip nat inside -> 2. interfejs (2. LAN mreža)
  ... ...
  r2(config-if)#ip nat outside -> vanjski interfejs

*** U ovom dijelu se radi konfiguracija PAT-a gdje se sve privatne adrese prevode u JEDNU JAVNU adresu ***

- r2(config)#access-list <n> permit <sumarna_adresa> <wildcard_maska>

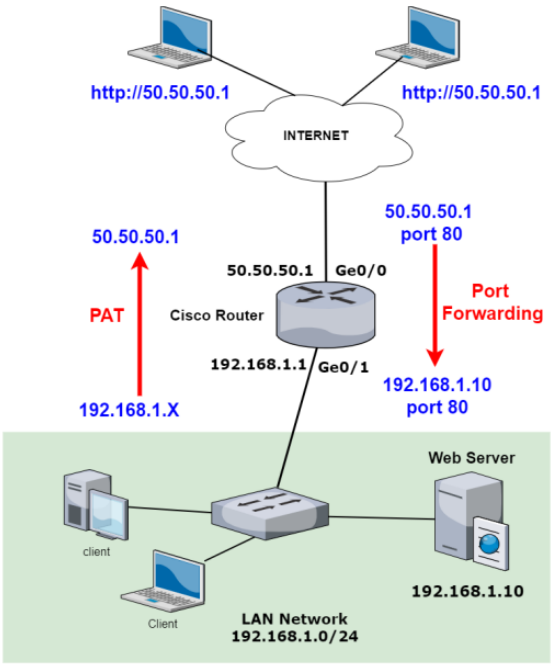
- **r2(config)#ip nat inside source list <n broj_liste> interface <naziv_interfejsa> overload** -> omogućavanje PAT interface konfiguracije

- r2(config-if)#ip nat inside -> 1. interfejs (1. LAN mreža)
  r2(config-if)#ip nat inside -> 2. interfejs (2. LAN mreža)
  ... ...
  r2(config-if)#ip nat outside

Još jedna korisna komanda je ili tehnika je **PORT FORWARDING**, što je proces preusmjeravanja dolaznih mrežnih paketa s jednog mrežnog porta na drugi ili na određenu IP adresu unutar lokalne mreže. Ovo se često koristi za omogućavanje pristupa servisima unutar privatne mreže (LAN) iz vanjske mreže (internet). Opšti oblik komande za to je:

ip nat inside source static { tcp | udp } <local-ip> <local-port> <global-ip> <global-port>

Na primjer, ako imamo sljedeću topologiju na slici ispod i želimo pristupiti veb serveru izvana na portu 80 (pod pretpostavkom da su sva ostala podešavanja urađena tj. rutiranje i PAT):



Kucali bismo komandu:

R1(config)#ip nat inside source **static tcp** 192.168.1.10 **80** 50.50.50.1 **80**

6. Lab 6 – IPv6 – Internet Protocol Version 6, OSPFv3

6.1. IPv6 adrese

*** Svaki interfejs koji radi sa IPv6 adresama, zapravo radi sa 2 IPv6 adrese, od kojih jednu MOŽE da ima, a drugu MORA da ima. Adresu koju **MORA** da ima jeste **LINK-LOCAL** IPv6 adresa (**FE80::**) i ako je ne zadamo, interfejs će je sam kreirati. Druga vrsta jesu **GLOBALNE UNICAST** adrese, uglavnom počinju sa hextetom **2001** (npr. 2001:db8:1:1::1/64). ***

- **R(config-if)# ipv6 address** <adresa>**/**<maska> -> pridruzivanje IPv6 adrese nekom interfejsu rutera. Maska se može ISKLJUČIVO dati u prefiksnom obliku (i to je maska mreže)

- **R(config-if)# ipv6 address** <adresa>**/**<maska> **link-local** -> pridruživanje link-local adrese interfejsu. Može se automatski dodijeliti, ali je u tom slučaju dosta dugačka. Može služiti kao DEFAULT GATEWAY

*** Da bi ruter radio sa IPv6 adresiranjem potrebno je to omogućiti ***

- **R(config)# ipv6 unicast-routing**

6.2. IPv6 podmrežavanje

*** Podmrežavanje sa IPv6 adresama se uvijek radi mijenjanjem 4. HEXTETA (ne okteta jer je sada jedno polje 16B). ***

Npr. ako je dat opseg 2001:db8:acad:ooc8::/64 i imamo 5 mreža (4LAN + 1 izmedju rutera)

VLAN1 -> 2001:db8:acad:**ooc8**::/64

VLAN2 -> 2001:db8:acad:**00c9**::1/64

VLAN3 -> 2001:db8:acad:**00ca**::1/64

VLAN4 -> 2001:db8:acad:**00cb**::1/64

R1-R2 -> 2001:db8:acad:**00cc**::1/64

*** Moguće je da neki host (PC) dobije automatski IPv6 adresu preko auto-configa (izabere se Automatic u IP configuration prozoru), pa će tako automatski dobiti globalnu unicast i link-local adresu. ***

6.3. OSPFv3 za IPv6

Da bi omogućili OSPFv3 rutiranje (tj. rutiranje za IPv6 adrese) na nekom ruteru, potrebno je kucati sljedeću komandu (ali najprije izvršiti komandu za samo IPv6 rutiranje **ipv6 unicast-routing**):

- R(config)# **ipv6 router ospf** <PROCESS_ID>

- R(config-rtr)#**router-id** <router_ID A.B.C.D> -> dodavanje ID-a ruteru

Nakon ovoga, moramo uključiti svaki interfejs u sam proces OSPF rutiranja i odgovarajući **area** sa komandama:

R(config)#interface <naziv interfejsa>

R(config-if)#**ipv6 ospf** <PROCESS_ID> area <AREA_ID>

PRIMJER (pridruživanje IPv6 adrese portu fa0/0 na nekom ruteru R1 i omogućavanje OSPFv3):

R1(config)#int fa0/0

R1(config-if)#ipv6 add 2000::1/64

R1(config-if)#no shut

R1(config)#ipv6 unicast-routing

R1(config)#ipv6 router ospf 1

R1(config-rtr)#router-id 1.1.1.1

R1(config)#int fa0/0

Router(config-if)# ipv6 ospf 1 area 0