VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS
ELEKTRONIKOS FAKULTETAS
KOMPIUTERIJOS IR RYŠIŲ TECHNOLOGIJŲ KATEDRA

Dengiančio medžio protokolas
Dalyko: Kompiuteriniai tinklai ir jų sauga
1 Namų darbas

Spanning tree protocol
Subject: Computer networks and their security
1 Home work

Atliko: KIKF-17 gr. Vsevolod Kapustin
Tikrino: Doc. Dr. Eimantas Garšva

Vilnius
2020

TURINYS

1. Darbo tikslas 3
2. Temos aktualumas 3
3. Dengiančio medžio protokolas 3
3.1. STP paskirtis 3
3.2. Dengiančio medžio protokolo (STP) veikimo principas 9
3.3. Dengiančio medžio protokolo (STP) versijos 11
4. Išvados 12
5. Literatūra 13

1. Darbo tikslas

Šiame darbe bus aiškinama kaip veikia dengiančio medžio protokolas, esantis perteklinių duomenų perdavimo takų užtikrinantys protokolas antrajame OSI (Angl. Open systems interconnect) sluoksnyje, kitaip kanaliniame sluoksnyje. Kuom skiriasi trečiame OSI lygmenyje ir antrajame OSI lygmenyje. Išsiaiškinama, kam naudojamas dengiančio medžio protokolas perteklinio tinklo sudarymui. Kaip dengiančio medžio protokolas užtikrina potencialių duomenų kanalo ciklų prevenciją. Kaip yra konfigūruojamas dengiančio medžio protokolas CISCO komutatoriuose. Kaip dengiančio medžio protokolas apima skirtingus virtualiu potinklius, ir įrodyta, kad dengiančio medžio protokolas naudojamas kiekvienam virtualiam potinkliui nepriklausomai vienas nuo kito.

2. Temos aktualumas

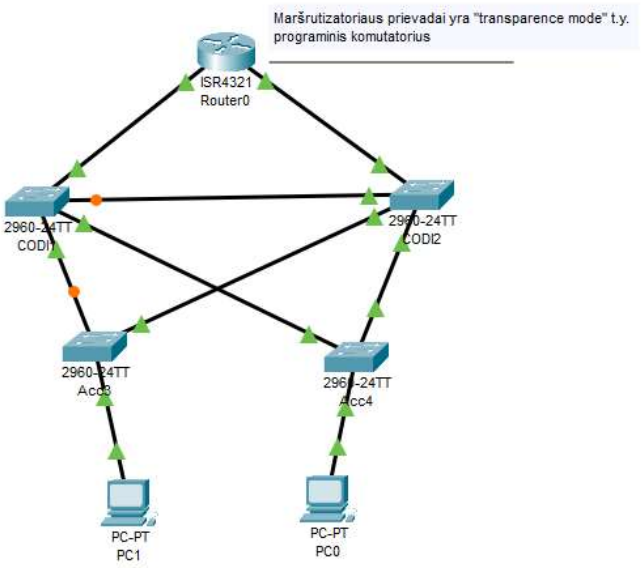
Dideliuose organizacijose, kur tinkle esantys įrenginiai turi pasiekti vienas kitą net gedimo atveju, reikia sudaryti sąlygas, kad tinklo funkcionavimas būtų nepertraukiamas. Tam naudojamas perteklinis komutatorių kiekis pagrindiniams ir pertekliniams duomenų kanalams sudaryti. Jeigu organizacijoje yra galinio vartojimo įrenginys, ryšio su kurio praradimas lemia didelius nuostolius nepriklausomai nuo laiko, perteklinė įranga ir konfigūruotas STP yra ne privalumas bet būtinybė. Taip pat, interneto tiekėjai privalo tūrėti konfigūruotą STP, nes esant paskirstymo (Angl. „Distribution“) bei pasiekimo (Angl. „Access“) sluoksnių komutatoriams be perteklinių kelių, esant profilaktikos, gedimų šalinimo darbams, bei nelaimės arba įrangos gedimo atvejais varduotojų duomenų srautai yra nutraukti, ir tai gali atsižvelgti ant interneto tiekėjo reputacijos bei padėties konkurentų rinkoje.

3. Dengiančio medžio protokolas

Šiame skyriuje bus aprašomas dengiančio medžio protokolas, kam jis yra naudojamas, kaip yra konfigūruojamas ir kokiems tinklams yra naudingas.

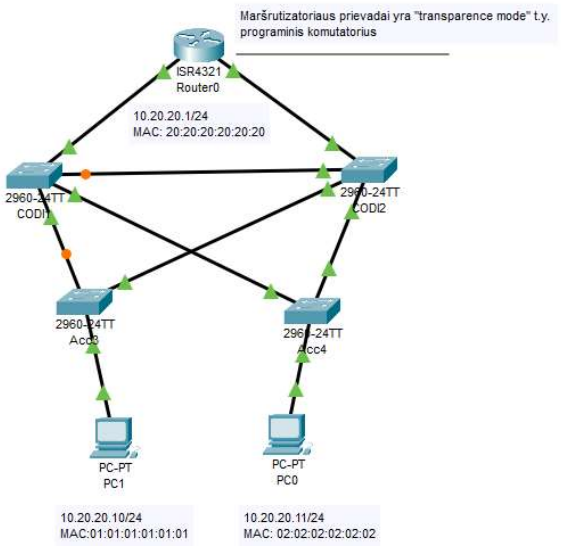
3.1. STP paskirtis

Tinkluose patikimumo didinimui dažnai naudojama perteklinė įranga, atsižvelgiant į faktą , kad dengiančio medžio protokolas veikia kanaliniame lygmenyje, tai ir STP naudojamas kanalinio lygmens įrangoje, t.y. komutatoriuose. Taigi jeigu kalba yra apie perteklinės įrangos naudojimą galime tokią sistemą atvaizduoti kaip parodyta pav. 3.1.:



Pav. 3.1. aiškinamoji vidinio tinklo schema

Galima pastebėti jog vieną iš komutatorių branduolio lygmens „CODI1“ arba „CODI2“ galima buvo nenaudoti, o prijungti pasiekimo komutatorius „ACC3“ ir „ACC4“ į vieną komutatorių, tačiau dėl perteklinių kelių sudarymo principo pridedamas dar vienas įrenginys. Atsižvelgiant į tai, kad ETHERNET protokolo kelio parinkimas yra vykdomas pagal komutatorių MAC adresų duomenų bazės (Užrašytus ARP (Angl. „Address Resolution Protocol“) lentelėje), kurie yra priskirti prie tam tikrų fizinių prievadų po pirmosios praeinančios per įrenginius ARP užklausos, stiebiame bandymo paketo kelią per tinklą. Nagrinėjimo patogumui, įvedamas tinklo adresavimas (pav. 3.2.):



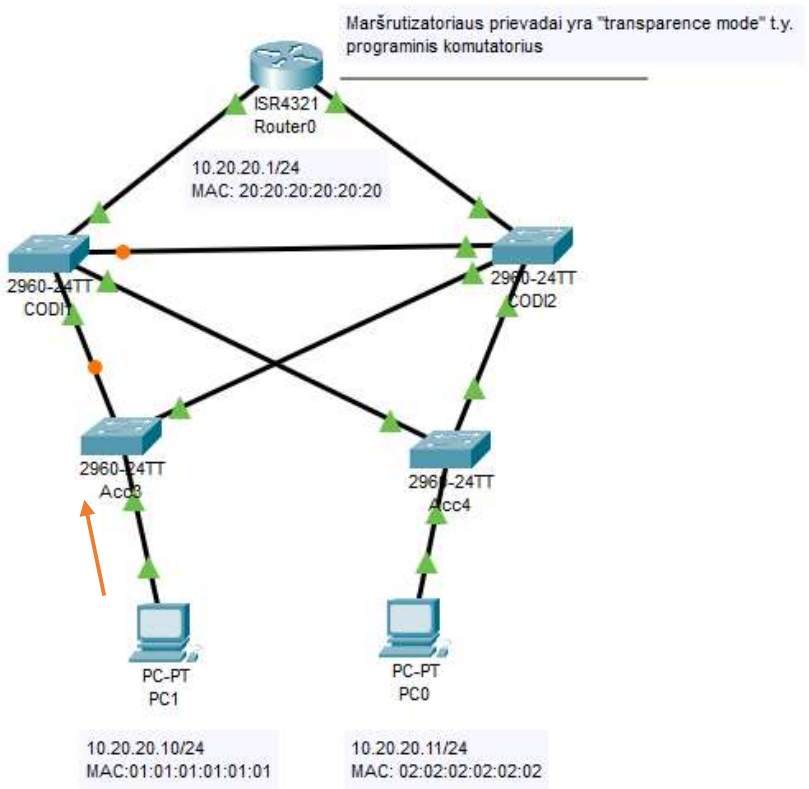
Pav. 3.2. aiškinamoji vidinio tinklo schema su loginiais ir fiziniais adresais

Tarkime kompiuteris „PC1“ nori išsiųsti paketą, kuris turi pasiekti maršrutizatoriaus prievadą esantį loginiu adresu 10.20.20.1. Panagrinėkime, kas bus tuo atveju, jeigu schemoje nėra įgyvendintas joks ciklų prevencijos protokolas.

Tarkime, kad tyrimas atliekamas momentu, kai per įrangą dar nebuvo nukeliavęs nei vienas paketas. Taigi pirmiausiai „PC1“ siųs ARP užklausą, tam, kad sužinoti paketo gavėjo MAC adresą, ARP užklausos turinys yra:

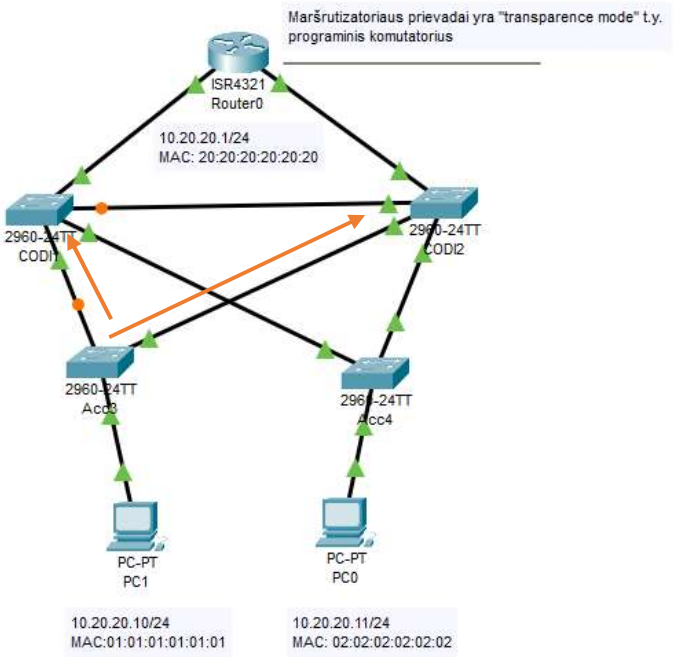
- ARP užklausą į maršrutizatorių (Gavėjas: 10.20.20.1)
- Siuntėjo MAC adresas: 01:01:01:01:01:01
- Gavėjo MAC adresas: FF:FF:FF:FF:FF:FF (pranešimas visam tinklui, nes kompiuterio atmintyje dar nėra įrašo, kuris atitiktų maršrutizatoriaus MAC adreso)

Grafiškai tai parodyta paveikslėlyje 3.3.:



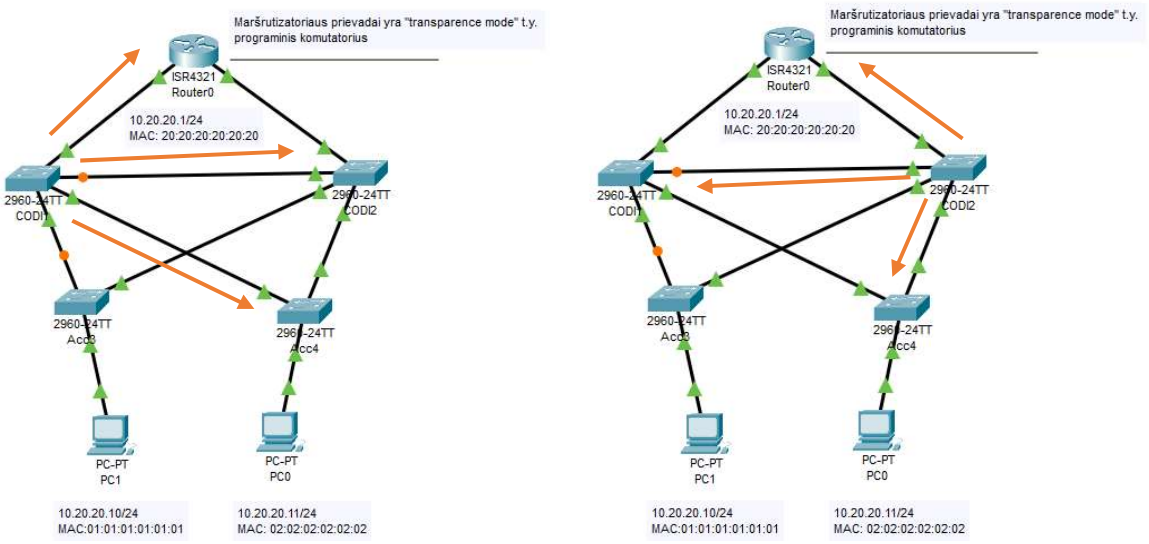
Pav. 3.3. ARP užklausa nuo PC1 kompiuterio

Antru žingsniu komutatorius „ACC3“ užrašys į MAC adresų lentelę MAC adresą 01:01:01:01:01:01 ir priskirs atitinkamą prievadą (pvz. fastethernet 0/10 – 10 komutatoriaus išvadas), kaip rezultatas, visi paketai kurie bus siunčiami adresui 01:01:01:01:01:01, patenkantys į komutatorių „ACC3“ bus nukreipti tik į 10 komutatoriaus prievadą. Atsižvelgiant į tai, kad užklausa yra siunčiama visiems „broadcast“, komutatorius „ACC3“ nukreips paketą į visus aktyvius prievadus išskyrus tą iš kurio jis buvo priimtas, tai parodyta pav. 3.4.:



Pav. 3.4. ACC3 peradresuota ARP užklausa

Komutatoriau „CODI1“ ir „CODI2“ įrašys į MAC adresų lentelę, kad kompiuterio „PC1“ MAC adresas yra pasiekimas atitinkamuose išvadose. T.y. mūsų tinklo komutatoriai žino kur yra kompiuteris „PC1“. Toliau komutatoriai branduolinio lygio persiųs kompiuterio ARP užklausą į visus aktyvius prievadus apart į tuos iš kurių buvo gautas pranešimas. Tai parodyta pav. 3.5.:



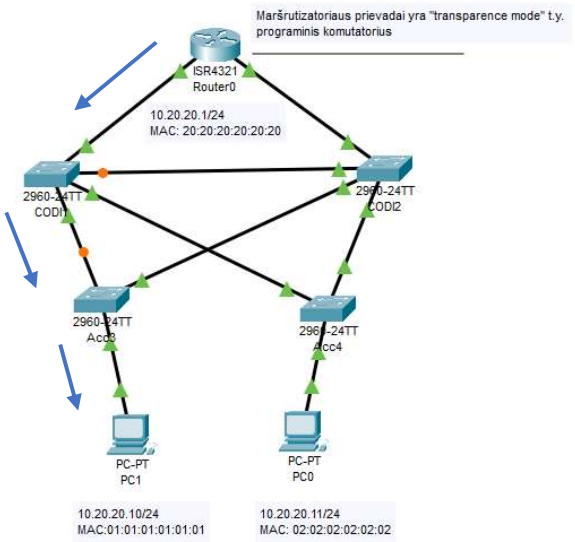
Pav. 3.5. CODI1 ir CODI2 komutatorių peradresuota ARP užklausa

Toliau matome, kad ARP užklausa sėkmingai pasieks maršrutizatoriaus prievadą. Vidinis maršrutizatoriaus loginis komutatorius užrašys kompiuterio MAC adresą į adresų lentelę, bei suformuos atsakymą, kurio turinys bus toks:

- ARP atsakymas (Maršrutizatorius siunčia savo MAC adresą, gavėjas 10.20.20.10)

- Siuntėjo MAC: 20:20:20:20:20:20
- Gavėjo MAC: 01:01:01:01:01:01

Toliau ARP atsakymas pasieks komutatorių CODI1 ir šis komutatorius užrašys į savo MAC adresų lentelę maršrutizatoriaus prievado MAC adresą. Atsižvelgiant į tai, kad komutatoriai CODI1 ir ACC3 jau turi MAC adresu lentelėse kompiuterio MAC adresą, paketas keliaus jau kryptingai, kaip parodyta, pav. 3.6.:



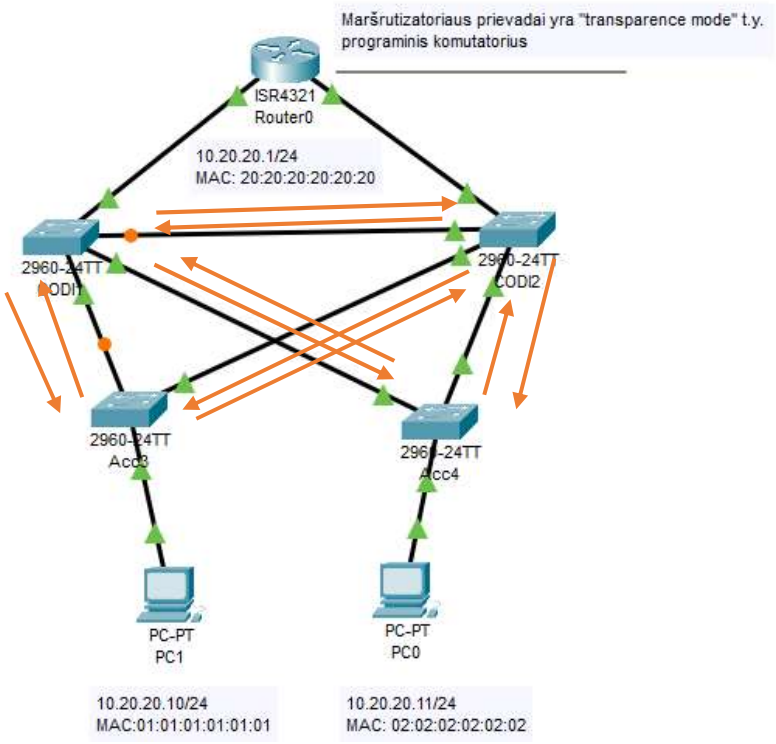
Pav. 3.6. ARP Atsakymo kelias tinkle

Tačiau galime pastebėti, kad pav. 3.5. pavaizduoti komutatoriai irgi persiaučia paketą į visus prievadus, todėl vienas ARP atsakymas bus gautas bet, tai sukels didelius nemalonumus tinklui. Žinant, kad komutatoriai yra kanalinio lygmens įrenginiai galima pažiūrėti ETHERNET antraštę kuri pavaizduota lentelėje 3.1.:

Lentelė. 3.1. ETHERNET kanalinio lygmens antraštė

Preambulė	Gavėjo MAC	Siuntėjo MAC	Ilgis/ETHERTYPE	Duomenys	FCS
8 baitai	6 baitai	6 baitai	2 baitai	46 – 1500 B	4 baitai

Galima pastebėti, kad kanalinio lygmens antrašte neturi TTL (Angl. „Time to Live“) lauko, kuris apriboja paketo pasiektų prievadų, skaičių, todėl ARP užklausa gali būti perstumiama komutatoriais begalybę kartų. Žinant tai galima pastebėti, jog ARP užklausa kuri yra skirta visiems tinklo įrenginiams (Tai matomas pagal užklausos gavėjo adresą FF:FF:FF:FF:FF:FF, kuris yra „broadcast“) CODI1 komutatorių, bus perduota į CODI2 komutatorių, iš CODI2 į ACC3 ir taip iki begalybės, taip tinkle išsidarys begaliniai ciklai, kaip pavaizduota pav. 3.7.:



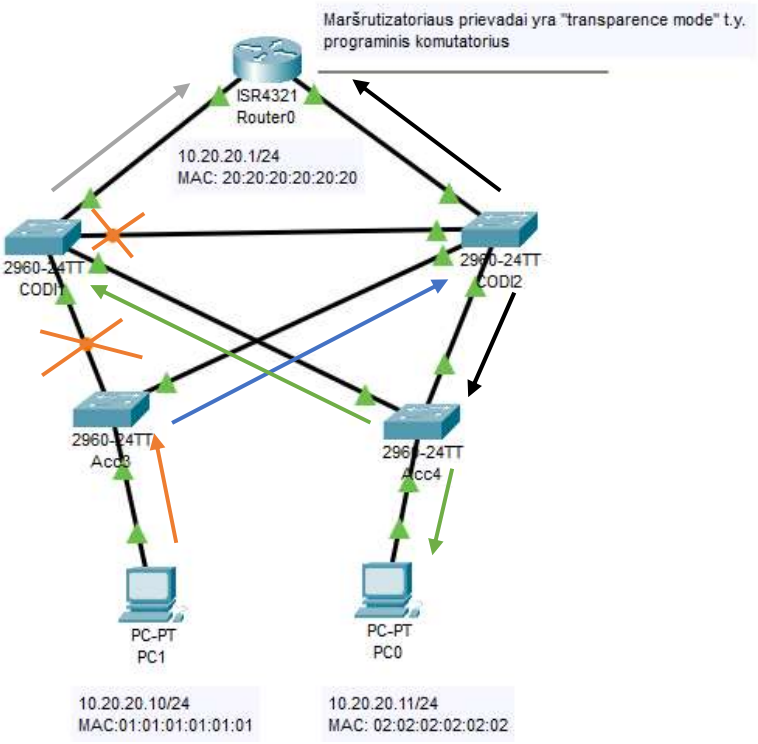
Pav. 3.7. Begaliniai ARP užklausos ciklai kanaliniame lygmenyje

Tai, susidarys du begaliniai ciklai tarp komutatorių:

1. CODI1 – CODI2 – ACC3;
2. CODI1 – CODI2 – ACC4.

Galima pastebėti, kad kiekvieną ciklo iteraciją paketų skaičius didėja dvigubai, reiškia mes turime reiškinį, kuris vadinasi transliavimo štormu („Broadcast storm“), kenksmingų paketų skaičius didės, kol įrangos procesorius nebus pajėgus praleisti tokį srautą duomenų ir įrenginiai nepraras funkcionalumo. Išeitis yra tik viena – perkrauti komutatorių, tačiau sekančiu bandymu kompiuterio PC1 siusti kanalinio lygmens protokolų transliavimo užklausas pakartos sistemos funkcionalumo sustabdymą.

Taigi, tam, kad sudaryti transliavimo štormo prevenciją, yra naudojamas dengiančio medžio protokolas – STP, kurio pagalba bus blokuojami prievadai, kurie gali dalyvauti begaliniuose cikluose. Koks prievadas bus blokuojamas sprendžiama pagal komutatorių prioritetus, tačiau tai bus aptarta sekančiame poskyryje. Panagrinėkime to pačio tinklo pavyzdį, kai STP yra sukonfigūruotas (pav. 3.8.):



Pav. 3.8. Aktyvuojamas STP

Matome, kad begaliniu ciklų šio atveju nesusidarys, nes prievadai, kurie galėjo dalyvauti begaliniame cikle yra užblokuoti, dabar tinklas gali pinai funkcionuoti. Taip, pat jei vienas iš komutatorių CODI1 arba CODI2 bus profilaktiškai aptarnaujamas arba turės gedimą, visi įrenginiai prijungti prie pasiekimo komutatorių ACC3 ir ACC4 be sutrikimų dalyvaus tinkle, o užblokuoti prievadai taps aktyvus, tai užtrunka apytiksliai 50 sekundžių tarp įrangos apklausų. Taigi, pagrindinė dengiančio medžio protokolas (STP) yra – esant perteklinei įrangai dubliuojantiems naudojamos duomenų srautų keliams sudaryti begalinių kanalinių ciklų prevencijos priemonė.

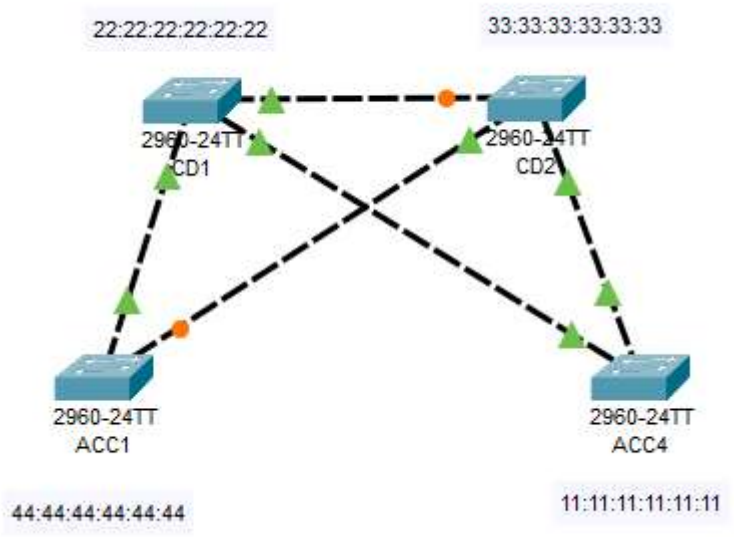
3.2. Dengiančio medžio protokolo (STP) veikimo principas

Dengiančio medžio protokolas – tai standartinis pramoninis protokolas, kuris yra pajungtas pagal nutylėjimą visuose parduodamuose konfigūruojamose komutatoriuose. Komutatoriai siunčia tilto (Angl. „Bridge Port Data Units - BPDU“) protokolo duomenų paketus į visus esamus prievadus, kai jie yra aktyvus. Tai naudojamą kitų komutatorių ir potencialių begalinių ciklų aptikimą. Komutatorius neleis duomenims keliauti iš prievado, kol nebus informacijos, kad jis nėra potencialus begaliniam ciklui.

Kai prievadas išeina į aktyvų režimą jis bus blokavimo stadijoje (Angl. „Blocking state“), kol nebus įsitikinta, kad nėra potencialių ciklų , šis procesas gali užtrukti apie 50 sekundžių. BPDU

paketuose yra įrašytas tilto identifikatorius (Angl. „Bridge ID“), kuris unikaliai aprašo kiekvieną komutatorių esantį tinkle. Tilto identifikatorius sudaromas iš komutatoriaus MAC adreso ir administratorių suteiktu tilto prioritetu (Angl. „Bridge Priority“), kuris gali svyruoti nuo 0 iki 65535. Pagal nutylėjimą tilto identifikatorius yra 32768.

Apsikečiant BPDU paketai, tinkle yra nustatomas pagrindinis komutatorius (Angl. „Root Bridge“), įrenginys, kurio prioritetas yra žemesnis, tampa pagrindiniu tiltu. Kiti komutatoriai užtikrina medžio topologija veikiančius komutatorių sekas iki pagrindinio tilto. Panagrinėkime prioretizavimo grandinę (pav. 3.9):

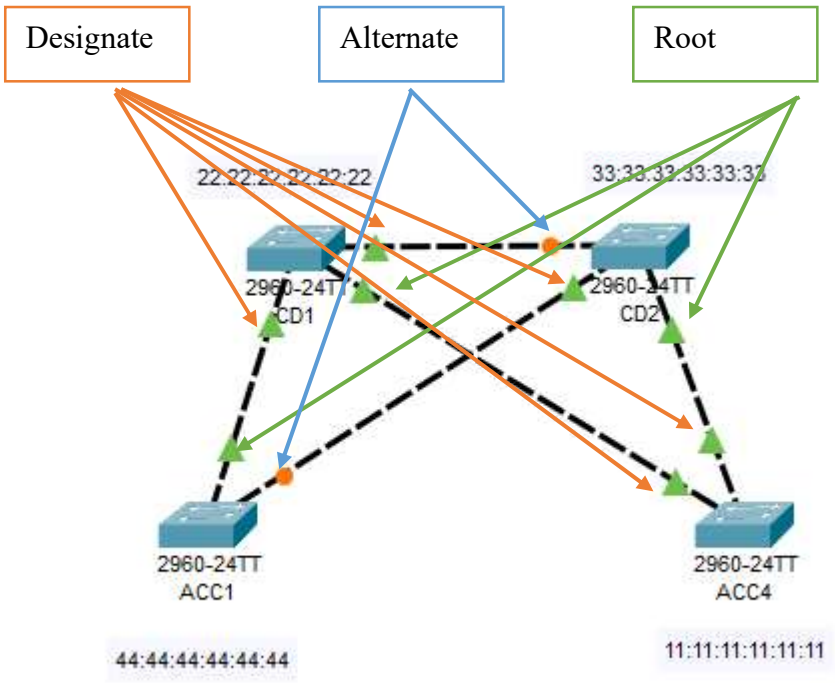


Pav. 3.9. STP prioretizavimo principas

Jeigu administratorius nenurodo prioritetą rankiniu būdu, tai pagrindiniu tiltu taps komutatorius, kurio fizinis adresas yra mažiausias, matome, kad tai yra komutatorius ACC3. Kiekvienas komutatoriaus prievadas prie kurio yra prijungtas kitas komutatorius tokioje schemoje įgyja savo pavadinimą, kurių yra 3:

1. Root – prievadas nukreiptas į „Root“ komutatoriaus pusę;
2. Designate – prievadas priešingai „Root“ komutatoriaus pusei;
3. Alternate – prievadas, kuris yra blokuojamas dėl potencialaus begalinio ciklo pavojaus.

Taigi pav. 3.10. yra paskirstyti skirtingi prievadų tipai:



Pav. 3.10. Prievadų paskirstymo schema

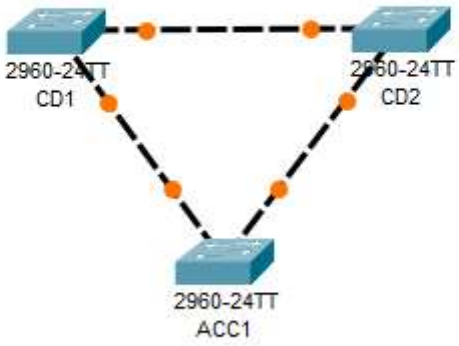
Galima atkreipti dėmesį, kad visi „Root“ komutatoriaus prievadai yra „designate“ tipo, ir šis komutatorius yra vienintelis, kuris tūri visus aktyvius prievadus duomenų perdavimui, nuo „Root“ komutatoriaus medžio topologija yra sujungti kiti komutatoriai.

3.3. Dengiančio medžio protokolo (STP) versijos

Dengiančio medžio protokolas (STP) yra standartinis protokolas, kuris yra pagal nutylėjimą paleistas visuose konfigūruojamose komutatoriuose. Jis atitinka tokiem atviriems standartams:

- IEEE802.1D Spanning tree protocol – originalus dengiančio medžio protokolas, naudojamas visiems virtualiems tinklams (Angl. „VLAN“) esantiems tinkle;
- IEEE802.1W Rapid Spannig tree protocol – yra patobulintas atsako laikas, naudojamas visiems virtualiems potinkliams tinkle;
- IEEE802.1S Multiple spanning tree protocol – Yra leidžiamas skirtingų virtualių potinklių grupavimas, tam kad sudaryti apkrovos balansavimą (Angl. „Load balansing“).

Apkrovos balansavimo pavyzdį galima panagrinėti ant pav. 3.11. esančio schemos:



Pav. 3.11. Apkrovos balansavimo nagrinėjimo schema

Tarkime, kad prie ACC3 komutatoriaus yra prijungti kompiuteriai, kurie yra paskirstyti į skirtingus potinklius, tam komutatoriuje yra sukonfigūruojami „Switchport mode Access“ arba „Untagged“ VLAN identifikatoriai. Tam, kad sudaryti apkrovos balansavimą, galime priskirti prie VLAN10 – 19 – CD1 kaip pagrindinį „Root“ komutatorių ir VLAN20 – 29 – CD2 kaip pagrindinį „Root“ maršrutizatorių. Taip pat duomenų srautas uždraustas tarp CD1 ir CD2 komutatorių. Taip yra sudarytas tinklas, kuriame STP procesai paleidžiami skirtingi, skirtingiems virtualių potinklių grupėms.

4. Išvados

Dengiančio medžio protokolas yra sukonfigūruotas visuose komutatoriuose esančiose gamyboje šiais laikais. Komutatorius neleis duomenų srautui judėti, jei nėra įsitikinimo, kad nėra potencialių kilpų tarp įrenginių. Faktiškai STP yra bauduojamas dideliuose tinkluose, sudarytuose pagal „Core – Distribution - access“ technologiją, kai yra montuojama perteklinę įranga fizinio duomenų kanalo praradimo prevencijai.

STP bendravimui tarp kanalinio lygmens įrenginių naudojami specialūs paketai, vadinami BPDU, kurie savyje neša informaciją apie kiekvieną komutatorių, jo identifikatorių ir prioritetą tinkle. Komutatorius su žemiausiu priritėtu tampa pagrindiniu, ir nuo jo yra sudaromas medžio topologija sujungtas duomenų kanalas. Jei tinklas yra segmentinis, t.y. tinklą sudaro keletas virtualių potinkliu, jiems atskirai arba grupėms gali būti naudojamos skirtingos STP instancijos, taip daroma dėl apkrovos balansavimo.

5. Literatūra

1. CCNP and CCIE Enterprise Core ENCOR 350-401 Official Cert. Ramiro Garza Rios, Brad Edgeworth, David Hucaby, Jason Gooley 2019-12-02.
2. CISCO CCNA. Neil Anderson.
3. CISCO PACKET TRACER guide. Prieiga per internetą: https://www.packettracernetwork.com/