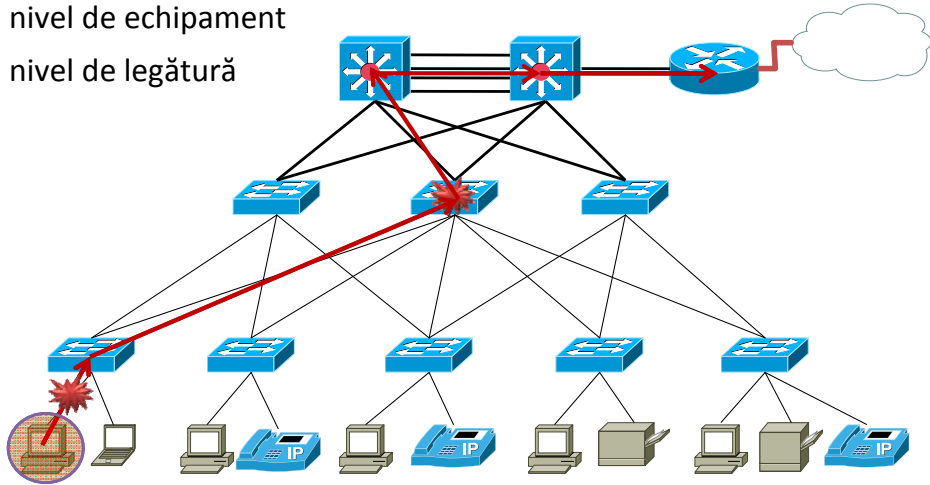


## Capitolul 5: STP

## Redundanță

- La nivel de echipament
- La nivel de legătură



Modelul de design ierarhic combate problemele ce apar în topologiile plate. Una dintre ele este necesitatea de a asigura redundanță. Redundanța la nivelul 2 îmbunătățește funcționarea rețelei prin implementarea unor căi alternative prin rețea. Acest lucru este realizat prin adăugarea de noi echipamente (cu aceleași funcționalități) și de noi cabluri ce le interconectează. A avea mai multe drumuri către aceeași destinație într-o rețea, permite ca în cazul în care una dintre legături pică rețeaua să funcționeze în continuare la parametrii optimi pe una dintre căile alternative.

Odată ce un business devine din ce în ce mai dependent de o rețea, rezistența rețelei la defecte devine din ce în ce mai importantă. Redundanța este soluția pentru a asigura toleranța la defecte a unei rețele.

## Bucle la nivelul 2

- Mecanism de protecție în cazul apariției unei bucle la nivelul 2
  - nu există
  - probleme ce pot apărea:
    - broadcast storms
    - copii multiple ale unui cadru
    - inconsistența tabelului CAM
- Mecanism de prevenire a buclelor la nivelul 2
  - Spanning Tree Protocol (STP)

Funcționarea permanentă a serviciilor de rețea a devenit unul din cele mai importante obiective pentru rețelele enterprise ce se bazează în principal pe o rețea multi-layer switched pentru a îndeplini cerințele companiei. O metodă de a asigura HA (High Availability) este de a realiza redundanță la nivelul 2 al stivei OSI, al echipamentelor de rețea, modulelor și link-urilor de-a lungul întregii rețele.

Redundanță la nivelul Legătură de date introduce bucle de nivelul 2 al stivei OSI, unde pachetele sunt transmise la nesfârșit între echipamente. Acestea pot avea un efect devastator asupra întregii rețele.

Menționăm câteva probleme ce pot apărea în cazul buclelor de nivelul 2: Broadcast storms, copii multiple ale unui cadru și inconsistența tabelului CAM.

Un mecanism de identificare și prevenire a acestor bucle este Spanning Tree Protocol.

# Spanning Tree Protocol



- Standard IEEE 802.1d (1990)
- Rulează **Spanning Tree Algorithm** pe fiecare switch
- Generează o topologie logică fără bucle
  - se închid porturi astfel încât legăturile să se mapeze pe arborele general

În 1990, IEEE a publicat primul standard pentru acest protocol și anume 802.1D bazat pe algoritmul realizat de Perlman. Variantele următoare au fost lansate în 1998 și 2004 incorporând diferite extensii.

STP rulează pe fiecare switch din topologie un algoritm Spanning Tree Algorithm generând astfel o topologie logică fără bucle de nivel 2. STP permite realizarea de topologii cu căi redundante, fără apariția efectelor nedorite a buclelor active în rețea.

Protocolul STP forțează anumite porturi să intre într-o stare de standby astfel încât aceste porturi să nu asculte, trimită sau „flood”-eze pachete. Efectul acestor măsuri este faptul că va exista o singură cale activă de-a lungul întregii topologii.

Dacă vor exista probleme cu calea activă generată, STP va restabili conectivitatea automat reactivând unul din link-urile dezactivate anterior, dacă o asemenea cale există.

## Spanning Tree Algorithm

- Rețeaua de switch-uri este un graf conex cu cicluri
- Fiecare switch din rețea este un nod în graf
  - fiecare switch are un ID unic (**bridge ID**)
- Fiecare legătură este un arc nedirecționat cu cost
  - costul este invers proporțional cu viteza legăturii
- Un switch este considerat rădăcină (**root bridge**)
- Arborele generat va fi arborele minim de acoperire
- Fiecare switch va avea drumul minim către rădăcină

STP folosește Spanning Tree Algorithm (STA) pentru a determina ce porturi din rețea trebuie să fie configurate să nu accepte trafic pentru a preveni apariția buclelor. STA alege un singur switch din cadrul rețelei ca Root Bridge și îl folosește ca punct de referință în calcularea căilor optime și fără bucle.

Fiecare switch ce participă în procesul de STP este un nod în graful calculat de STA și îi va fi asociat un Bridge ID unic. Protocolul va face schimb de pachete BPDU (Bridge Protocol Data Unit) pentru a determina cel mai mic Bridge ID în vederea alegerii Root Bridge-ului.

## Bridge ID

- Fiecare switch are un ID unic (BID)
- Valoare pe **64 biți**
  - 16 biți **prioritatea**
  - 48 biți **adresa MAC**
- Prioritatea este implicit 32768
- Switch-ul cu BID-ul cel mai mic este root bridge-ul

2 Bytes	6 Bytes
Prioritate	MAC

Protocolul STP necesită ca fiecare switch din rețea să aibă asignat un **BID (Bridge ID)** unic. În versiunea inițială standardizată, în 802.1D, BID-ul era compus din 2 câmpuri: **Prioritate** și **MAC**, iar toate VLAN-urile erau reprezentate de un arbore STP comun.

Switch-ul cu **cel mai mic** BID devine Root Bridge.

**Prioritatea (Bridge ID)** este un câmp de o dimensiune de 16 biți ce stochează prioritatea switch-ului. Valoarea implicită a priorității este 32768, valoare de mijloc ce poate fi specificată. Datorită adăugării câmpului de VLAN ID în cadrul priorității, pentru PVST+ și PVRST+ dimensiunea câmpului ce specifică prioritatea se reduce la 4 biți. Din această cauză, în noul format, valoarea pentru prioritate se incrementează cu 4096.

**Adresa MAC** este un câmp de 48 de biți ce memorează adresa de Layer 2 a switch-ului respectiv.

## Link cost

Viteză legătură	802.1D (1990)	802.1D (1998)	802.1t (2001)
4 MB/s	-	250	5,000,000
10 MB/s	100	100	2,000,000
16 MB/s	-	62	1,250,000
100 MB/s	10	19	200,000
1 GB/s	1	4	20,000
2 GB/s	-	3	10,000
10 GB/s	1	2	2,000

În momentul în care a fost ales Root Bridge-ul pentru o instanță de STP, algoritmul protocolului (STA) începe procesul de determinare a căii optime dinspre fiecare destinație din domeniul de broadcast către acesta.

Costul implicit al fiecărei căi este determinat de viteza la care operează porturile folosite în calea respectivă. Dintre cele mai importante valori amintim: porturile cu o viteză de 10GB/s ce au costul 2 , porturile cu viteză de 1 GB/s ce au costul 4, porturile cu viteză de 100 MB/s FastEthernet ce au costul 19 și porturile de 10 MB/s Ethernet ce au costul 100.

## BPDU

- Mesaje trimise între switch-uri
  - Bridge Protocol Data Unit (BPDU)
  - trimise la fiecare 2 secunde
  - multicast spre **01:80:C2:00:00:00**

2 Bytes	1 Byte	1 Byte	1 Byte	8 Bytes	4 Bytes	8 Bytes	2 Bytes	2 Bytes	2 Bytes	2 Bytes	2 Bytes
Protocol Identifier	Version	Message Type	Flags	Root ID	Cost to bridge	Bridge ID	Port ID	Message Age	Max Age	Hello Time	Forward Delay

Alegerea Root Bridge-ului pentru instanța de STP se realizează folosind pachete BPDU (Bridge Protocol Data Unit).

Frame-ul BPDU conține douăsprezece câmpuri distincte ce sunt folosite în cadrul procesului de STP pentru a determina Root Bridge-ul și căile optime spre acesta.

Primele 4 câmpuri din cadrul BPDU-ului identifică protocolul, versiunea, tipul mesajului și flag-uri de status.

Următoarele 4 câmpuri sunt folosite pentru a specifica Root Bridge-ul și costul căii spre acesta.

Ultimele 4 câmpuri sunt valori de timere ce determină cât de des trebuie trimise pachetele.



## Porturi

### ▪ Tipuri de porturi:

#### – root ports

- porturile care duc spre root pe calea cu costul cel mai mic
- unul pe fiecare switch non-root

#### – designated ports

- porturile care permit traficul

#### – non-designated ports

- porturile care blochează traficul

În continuare vom defini rolurile port-urilor unui switch ce este clasificat drept non – designated:

•Portul ce se găsește pe calea cea mai bună spre un Root Bridge se numește **Root Port**. Aceste porturi transmit datele mai departe către **Root Bridge**, iar adresa MAC a pachetele primite pe acest port vor fi introduse în tabela CAM. Pe un switch Non – designated poate fi definit un singur Root Port.

•Porturile **Designated** există atât pe switch-uri Root Bridge cât și pe cele care nu sunt Root Bridge. Pentru switch-urile Root Bridge toate porturile acestuia sunt porturi Designated. Pentru switch-urile ce nu sunt Root Bridge, porturile Designated permit trecerea traficului. În cadrul unui segment de rețea un singur port poate fi Designated.

•Porturile **Non – designated** sunt porturile ce nu permit transmiterea și primirea datelor și nici nu permite învățarea de noi adrese MAC.

## Etapele procesului de STP

- Fiecare switch se consideră root bridge
- Se schimbă BPDU-uri și se alege un singur root bridge
- Se ia în considerare doar calea cea mai scurtă către root
- Se închid porturile care nu duc spre root pe calea cea mai scurtă

Procesul de STP este format din următorii pași:

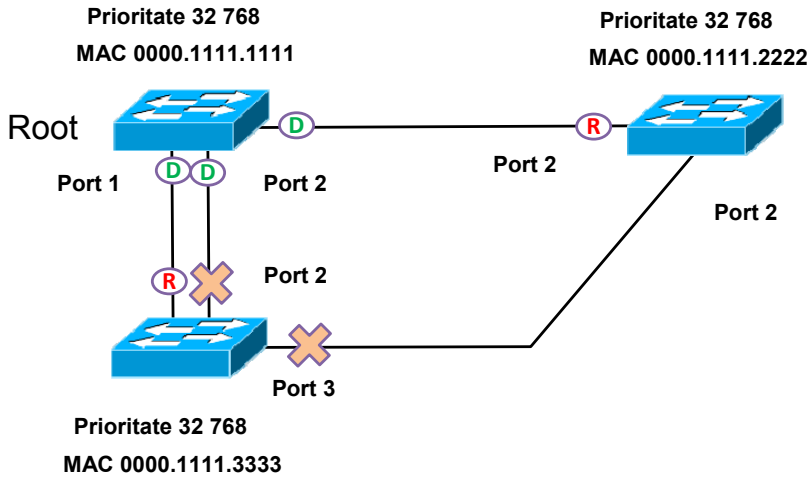
1) Desemnarea unui Root Bridge: la început fiecare switch din rețea se consideră Root Bridge. Protocolul STP rulează un proces de determinare a unui singur switch Root Bridge în cadrul unui domeniu de broadcast. Toate porturile acestuia sunt porturi Designated.

2) Selectarea Root Port-urilor: protocolul va stabili un singur Root Port pe fiecare switch ce nu a fost ales Root Bridge. Root Port-ul va fi calea de cost minim de la orice switch până la Root Bridge. Există posibilitatea să existe mai multe căi de cost egal, caz în care se va alege drept Root Port, portul cu ID-ul cel mai mic. Valoarea Port ID-ului este formată dintr-o prioritate ce poate fi configurată și numărul portului.

3) Se aleg porturile Designated pe fiecare segment.

4) Porturile rămase se închid (au rolul Blocked Port)

# Exemplu Porturi



## Stări Porturi în STP

- Un port face tranziția între mai multe stări

Stare port	Ațiune la nivel de Switch	Ațiune la nivel de Port
Disabled	Nu se acceptă nici un fel de trafic	Nu se transmit cadre Nu se transmit BPDU-uri
Blocking	Se primesc doar BPDU-uri	Nu se transmit cadre Se primesc BPDU-uri
Listening	Se construiește topologia STP	Nu se transmit cadre Se transmit BPDU-uri
Learning	Se construiește tabela de adrese MAC	Nu se transmit cadre Se învață adrese MAC Se transmit BPDU-uri
Forwarding	Se transmite traficul normal	Se transmit cadre Se învață adrese MAC Se transmit BPDU-uri

În starea **Disabled** portul nu va participa în procesul de STP și nu va transmite pachete și BPDU-uri.

Porturile aflate în starea **Blocking** sunt porturi non – designated și nu participă în trimiterea de pachete. Aceste porturi primesc BPDU-uri pentru a determina locația și ID-ul Root Bridge-ului și pentru a stabili rolurile fiecărui port al său (root, designated, nondesignated) în topologia finală.

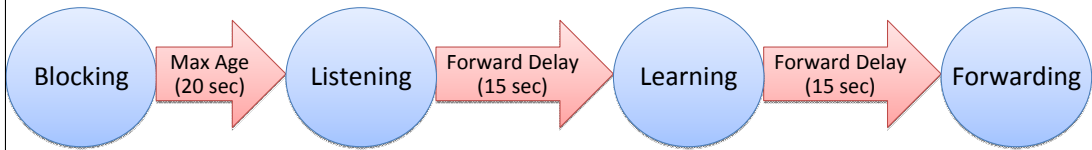
Începând din momentul în care porturile ajung în starea **Listening**, STP-ul a determinat că aceste port-uri pot participa la schimbul de BPDU-uri și se începe construirea topologiei STP.

În starea **Learning**, portul se pregătește să participe în transmiterea de frame-uri și începe să își populeze tabela CAM.

Un port în starea **Forwarding** este considerat ca membru al topologiei active și va trimite frame-uri și BPDU-uri.

## Timpi de tranziție

- Timere de tranziție
  - stabilitate de root bridge
  - **Hello time:** 2 sec
  - **Forwarding delay:** 15 sec
  - **Max Age:** 20 sec



– timp total de convergență: 50 sec

Fiecare port din cadrul unui proces de STP va trebui să treacă prin stările menționate anterior. În următoarele rânduri vom specifica timpii de tranziție între stările amintite.

Portul va sta în starea Blocking timp de 20 de secunde, timpul implicit **Max Age**. În fiecare din stările următoare și anume Listening și Learning, porturile vor sta câte 15 secunde (Forwarding Delay).

Astfel timpul total de convergență este de 50 de secunde.

## CST



- Common Spanning Tree
- O singură instanță de 802.1D
- Implementare pentru un singur VLAN

Protocolul CST (Common Spanning Tree) își asumă o singură instanță de STP pentru întreaga rețea indiferent de numărul de VLAN-uri existente. Datorită faptului că există o singură instanță, cerințele de memorie și de CPU sunt drastic micșorate în comparație cu variantele de STP apărute recent. Un dezavantaj al acestui protocol este faptul că fiind o singură instanță de STP va rezulta un singur arbore și un singur Root Bridge. Astfel tot traficul, indiferent din ce VLAN provine, va parcurge aceeași cale și poate duce la o transmitere suboptimală a datelor.

Deoarece procesul de convergență este moștenit de la standardul 802.1D, timpul de convergență este mare.

## RSTP



- Rapid Spanning Tree Protocol
- IEEE 802.1w (1998)
- Timp mai bun de convergență: 3-5 secunde

Rapid Spanning Tree Protocol (RSTP) sau 802.1w este un protocol ce s-a lansat ca evoluție a protocolului STP, ce oferă timpi de convergență mult mai buni. Această versiune adresează multe din problemele de convergență, însă, datorită faptului că folosește tot o singură instanță de STP, nu a putut rezolva problemele legate de separarea arborilor STP în funcție de VLAN-uri. Pentru a micșora timpii de convergență, RSTP folosește mai multe resurse de memorie decât CST, însă diferențele sunt evidente: De la 50 de secunde CST la 3-5 secunde RSTP.

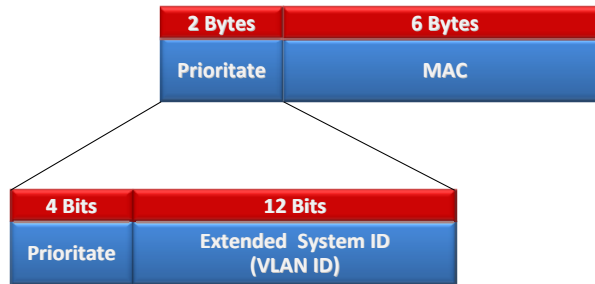
## PVST/PVST+/RPVST+

- Proprietare Cisco
- Câte o instanță de STP pentru fiecare VLAN
- PVST
  - funcționează doar peste trunk-uri Cisco ISL
- PVST+
  - funcționează peste trunk-uri 802.1q
- RPVST+
  - Rapid PVST+
  - timpi de convergență similari cu RSTP

Per VLAN Spanning Tree Plus (PVST+) este o îmbunătățire a STP-ului ce oferă o instanță de STP pentru fiecare VLAN configurat în rețea. Fiecare instanță suportă PortFast, BPDU guard, BPDU filter, Root guard și Loop guard, noțiuni ce vor fi studiate în amănunt în cursurile de CCNP. Realizându-se o instanță pentru fiecare VLAN, numărul de resurse necesare crește (memorie și CPU), însă permite alegerea unui Root Bridge per VLAN. Timpul de convergență este asemănător cu 802.1D dar această convergență este per VLAN.



## Identificarea VLAN-ului



Deoarece PVST+ și PVRST+, ambele fiind variațiuni ale protocolului STP, au nevoie de un arbore STP separat pentru fiecare VLAN, câmpul **Bridge ID**-ului trebuie să conțină și informații despre VLAN ID. Acest lucru este posibil prin re folosirea unei porțiuni din câmpul de prioritate pentru stocarea VLAN ID-ului.

# MSTP



- IEEE 802.1s
- Extensie la RSTP pentru a putea folosi VLAN-uri

MST (Multiple Spanning Tree) este un standard IEEE inspirat din versiunile anterioare ale protocolului proprietar Cisco Multi-Instance Spanning Tree Protocol (MISTP). Implementarea celor de la Cisco asigură până la 16 instanțe de RSTP (802.1w) și combină mai multe VLAN-uri cu topologii fizice și logice într-o instanță comună de RSTP. Fiecare instanță suportă PortFast , BPDU guard, BPDU filter, Root guard și Loop guard.

## Configurări globale

- Mod de funcționare

```
spanning-tree mode pvst | rapid-vsp | mst
```

- Setare prioritate manual

```
spanning-tree vlan VLAN_NO priority PRIORITY
```

- Setare prioritate automat

```
spanning-tree vlan VLAN_NO root primary  
spanning-tree vlan VLAN_NO root secondary
```

Una din comenzile foarte importante în implementarea STP-ului în infrastructura noastră este **spanning-tree mode pvst / rapid-pvst / mst** cu ajutorul căreia putem nominaliza tipul de STP.

În cazul în care dorim să influențăm alegerea Root Bridge-ului, pe anumite VLAN-uri putem folosi comenzile **spanning-tree vlan VLAN\_NO root primary** și **spanning-tree vlan VLAN\_NO secondary**.

Dacă avem nevoie de mai mult control pentru setarea Root Bridge-ului avem la dispoziție comanda **spanning-tree vlan VLAN\_NO priority** ce ne permite să setăm prioritatea în multipli de 4096.

## Configurări pe interfață

- Cost legătură

```
spanning-tree cost COST
```

- Activare portfast

```
spanning-tree portfast
```

Datorită faptului că tehnologia se dezvoltă foarte rapid, trebuie să ne așteptăm ca valorile asociate costului porturilor să se schimbe pentru a acomoda viteze din ce în ce mai mari.

Cu toate că porturile au definite valori implicite, în momentul inițializării switch-ului, aceste valori pot fi modificate, permițând administratorului să aibă control asupra căii ce este aleasă de procesul de STP. Comanda cu ajutorul căreia se realizează modificarea, este **spanning-tree cost COST**.

Există cazuri în care interconectăm echipamente terminale la switch, precum servere, imprimante, ce duc la limitarea posibilității de realizare a unei bucle de nivel 2. Aceste porturi pot fi setate să nu ruleze algoritmul de STP, micșorând astfel timpul până când intră în starea de forwarding.

Comanda folosită este **spanning-tree portfast** executată din prompt-ul interfeței.

## Rezumat

- Spanning Tree Protocol
- BPDU
- RSTP
- PVST/PVST+/RPVST+



1. Dați exemplu de o implementare de STP proprietara Cisco.
2. Enumerați stările porturilor în STP.
3. Ce tipuri de porturi permit traficul în protocolul STP?
4. Pe câți biți este reprezentat Bridge ID-ul?
5. Care este costul unei legături de 100Mb/s?