

A decorative graphic on the left side of the slide, consisting of a network of yellow lines and circles. The lines are vertical and horizontal, with some branching out, resembling a circuit board or a tree structure. The circles are small and are placed at various points along the lines, some at the ends and some in the middle. The overall effect is a stylized, abstract representation of a network or circuit.

STP

SPANNING TREE PROTOCOL

SPANNING TREE PROTOCOL

- Ezt a prototkolt a kapcsolók futtatják. Célja az, hogy az ethernet hálózaton megszüntesse a hurkokat.
- Ezt úgy teszi, hogy amikor bekapcsol egy interfész, akkor addig nem enged rajta forgalmat küldeni, amíg meg nem bizonyosodik arról, hogy azon az interfészen keresztül nem jön létre hurok az ethernet hálózaton.
- Ez a folyamat alapértelmezetten 50 másodpercig tart és ezért „narancssárga” a kapcsoló interfésze, ha csatlakoztatunk hozzá valamit.

HURKOK AZ ETHERNET HÁLÓZATON

Miért probléma a hurok?

- Ehhez a kapcsolók működéséből kell kiindulni: a kapcsoló ha nem ismeri, hogy a címzett merre van, akkor minden interfészen továbbítja az üzenetet kivéve azon, ahol beérkezett.
- Mi történik akkor, ha egyik kapcsoló sem ismeri a címzettet és a címzett nem válaszol?
 1. Ha nincs hurok, nincs baj, mert mindenki megkapja az üzenetet és nem válaszol senki.
 2. Ha hurok van, akkor az üzenet körbeér és a kapcsoló újra továbbítja, mert az üzenetre nincs ráírva, hogy ez már volt. Így az üzenet addig kering a hálózaton, míg a címzett eszközt be nem kapcsolják és nem kerül bele valamelyik kapcsoló MAC-cím táblájába.

HURKOK AZ ETHERNET HÁLÓZATON

- Még nagyobb a baj, ha az üzenet szórási, vagy multicast címre megy. Ezeket az üzeneteket a kapcsolók csak elárasztással tudják küldeni és hiába válaszol a címzett, az üzenet örökké keringeni fog!
- A másik probléma a keringéssel, hogy a kapcsolók a beérkező üzenetek forráscíméből tudják, hogy mi merre található. A folyamatosan keringő üzenetek felülírják a kapcsolókon lévő helyes bejegyzéseket és rossz MAC-cím tábla bejegyzéseket eredményeznek. Még a normál üzenetek is eltévednek.

SZÓRÁSI VIHAR

A kapcsolók alapvetően arra vannak tervezve, hogy minél gyorsabban továbbítsák az üzeneteket. Ez hurkok esetén azt jelenti, hogy a keringő üzeneteket teljes sebességgel folyamatosan küldi tovább az összes kapcsoló. Ha elegendő keringő üzenet gyűlik össze, akkor ezek az üzenetek telítik a teljes sávszélességét a kapcsolónak és onnan kezdve semmilyen „normális” üzenetet sem tudnak továbbítani. A hálózat leáll. Ezt hívják szórási viharnak. Nagyobb hálózatok esetén néhány másodperc is elég ehhez. Megszüntetni csak a hurok megszakításával lehet!

MIÉRT LEHET HUROK AZ ETHERNET HÁLÓZATON?

- Véletlenül (direkt): a hurokmentes hálózathoz valaki úgy csatlakoztat egy kábelt, hogy hurkot okoz (ez DoS támadásnak minősül).
- Redundancia miatt: ha nincs hurok a hálózaton, akkor ha a hálózat bármely része meghibásodik a hibán túli rész elérhetetlenné válik. Ha hurkokat készítünk, akkor meghibásodás esetén is működőképes lesz a hálózat. Ezzel az elgondolással az a baj, hogy ekkor meghibásodás nélkül lesz működésképtelen a hálózat. Ezért készítették az STP protokolt.

STP MŰKÖDÉSE

- Ha bekapcsolunk egy interfészt azonnal elindul rajta az STP protokoll és elkezd ellenőrizni, hogy most létrejött-e hurok.
- Ennek érdekében üzeneteket (BPDU) kezd küldeni.
- A protokoll az OSI modell második rétegében (adatkapcsolati réteg, TCP/IP: hálózatalérési réteg) fut.
- Működéséhez nem kell IP cím sem.

STP ÁLLAPOTOK

- Az STP működése során 4 állapoton megy keresztül az interfész:
- Blokkoló (blocking): ekkor csak BPDU-t küld és fogad, ha hurkot okoz az interfész ebben az állapotban marad (minimum 20 mp).
- Hallgató (listening): ekkor is csak BPDU-kal foglalkozik, de ekkor már tudja, hogy be lesz kapcsolva (15 mp)
- Tanuló (learning): ekkor már fogadja a felhasználók üzeneteit és elkezdti feltölteni a MAC-cím táblát. Még nem továbbítja az üzeneteket (15 mp).
- Továbbító (forwarding): ekkor normálisan működik az interfész, továbbít minden üzenetet

HURKOK FELDERÍTÉSE

- Az STP protokoll úgy találja meg a hurkokat a hálózatban, hogy felépít egy feszítőfát (spanning tree). A fa struktúra ugyanúgy épül fel, mint a „rendes” fák. Van neki egy gyökere (root) – a valóságban ez inkább a törzse lenne, de nem biológusok találták ki az algoritmust - és a gyökérből mindenhová csak egy út vezet. Nincs benne hurok.
- Tehát az STP protokoll első dolga, hogy megtalálja a gyökeret. Ha egy kapcsolót bekapcsolnak, amíg nem kap olyan információt, hogy valaki más alkalmasabb gyökérnek, addig magát tekinti annak és ezt hirdeti is.

GYÖKÉRVÁLASZTÁS

- Hogyan döntik el a kapcsolók, hogy ki lesz a gyökér?
- Ezt egy BID-nek nevezett azonosítóval döntik el. Akinek a legalacsonyabb a BID-e, az lesz a gyökér. Ez úgy alakul ki, hogy miután kezdetben mindenki magát tekinti gyökérnek. Kap egy üzenetet a szomszéd kapcsolótól és amelyiknek alacsonyabb a BID-e, a továbbiakban azt tekinti gyökérnek és azt hirdeti. Előbb-utóbb minden kapcsoló megtudja ki a legalacsonyabb BID-rel rendelkező kapcsoló. Az összes kapcsolón ugyanaz lesz a gyökér.

BID – BRIDGE IDENTIFIER

- A BID a kapcsoló azonosítója. 8 bájt hosszú és két részből áll:
- 2 bájt: prioritás (4 bit) + VLAN sorszáma (12 bit, minden VLAN-ra külön számolnak feszítőfát a kapcsolók). Alapértelmezett prioritás: 1000
- 6 bájt: a kapcsoló MAC címe
- Alapvetően a prioritás dönt, de ha nincs beállítva, akkor a legalacsonyabb MAC-című kapcsoló lesz a gyökér.

GYÖKÉRHEZ VEZETŐ ÚT

- Ha megvan a gyökér, akkor a gyökér az összes interfészét kijelöli továbbítónak.
- A többi kapcsoló meghatározza a gyökérhez vezető legrövidebb útvonalat és az ehhez kiinduló interfészét. Ez lesz a gyökérport. Ezeket továbbító állapotba kapcsolják. A legrövidebb útvonalat a sáv szélesség és a távolság alapján számolják ki.

ALTERNATÍV PORTOK

A kapcsolók többi interfésznél két eset lehetséges:

- Nincs STP-t futtató eszköz a kapcsolaton: ekkor az interfészen állomás található, így az interfész kijelölhető továbbításra.
- A túlsó végén is STP-t futtató eszköz van (kapcsoló): ITT VAN EGY HUOK

Ekkor a két kapcsoló megegyezik, hogy az egyikük lezárja a kapcsolatot, a másik pedig továbbít rá. Ilyen módon megszűnik a hurok. Az lesz továbbító amelyiknek alacsonyabb a BID-je. A másik egy speciális blokkoló állapotba kerül: alternatív port lesz.

STP PORT SZEREPKÖRÖK

- Gyökér (root): továbbító port. Az az interfész amelyen keresztül a kapcsoló a gyökérhez kapcsolódik, vagy a gyökérhez vezető legrövidebb útvonalhoz tartozó interfész (csak egy lehet minden kapcsolón)
- Kijelölt (designated): továbbító port. A kapcsoló összes olyan interfésze, amelyik továbbít és nem gyökérport.
- Alternatív (alternate): blokkoló port. A hurkokhoz tartozó lezárt interfész.

STP BEÁLLÍTÁSA

- Az STP alapértelmezetten fut a kapcsolókon. Nem igényel beállítást.
- Az egyik dolog, amit állítunk a működésén az az, hogy melyik kapcsoló legyen a gyökér. Ezt a prioritás átállításával tehetjük meg (VLAN 1):

```
S1(config)# spanning-tree vlan 1 root primary
```

- Kijelölhetünk másodlagos gyökeret is:

```
S1(config)# spanning-tree vlan 1 root secondary
```

- Erre akkor van szükség, ha a gyökér kiesése esetén is biztosítani akarjuk, hogy melyik legyen az új gyökér.

STP BEÁLLÍTÁSA

- Ha nem a VLAN 1 prioritását akarjuk átállítani természetesen lehet bármelyik VLAN-t megadni. Egyszerre több VLAN-t is be lehet állítani: vesszővel felsorolva (vlan 10,20,30), vagy kötőjellel egy tartományt (vlan 1-100).

- Meg lehet adni a prioritást direkt is:

```
S1(config)# spanning-tree vlan 1,10-15,20 priority 4096
```

- Ekkor a prioritás értéke 4096 többszöröse lehet (alapértelmezett: 32768)
- A gyökér megváltozására a kapcsolók újra kiszámolják a gyökér útvonalakat és áthelyeződnek a lekapcsolt interfészek.

STP ELLENŐRZÉSE

- Meg lehet nézni a kapcsolón, hogy melyik VLAN-ban melyik interfész és a kapcsoló milyen szerepet tölt be:

```
S1#show spanning-tree
```

- Ennél a parancsnál a prioritás helyett a BID első 2 bájtyjának értéket írja ki.
- VLAN 1 esetében , ha a prioritás: 32768, akkor a kiírt érték $32768+1=32769$ lesz!

RSTP – RAPID SPANNING TREE PROTOCOL

- Ez a gyors feszítőfa protokoll. Annyiban különbözik a normál feszítőfa protokolltól, hogy ez csak full-duplex interfészekkel rendelkező hálózatokon futhat (ma már minden hálózat ilyen) és 2-3mp alatt képes kialakítani és bekapcsolni a feszítőfát.
- Ha ezt akarjuk használni, az összes kapcsolón el kell indítani:

```
S1(config)#spanning-tree mode rapid-pvst
```


FESZÍTŐFA PROTOKOLL KIKAPCSOLÁSA

- Ezt interfészenként tehetjük meg és óvatosan bánjunk vele, mert ekkor az adott interfész azonnal bekapcsol és ha így hurok keletkezik, akkor szórási vihar alakulhat ki (PT esetében ez általában a program kifagyását jelenti). Általában csak olyan porton használják, amelyre állomásokat kötnek:

```
S1(config-if)# spanning-tree portfast
```

- Lehetséges az összes elérési interfészt alapértelmezetten ebbe az állapotba tenni:

```
S1(config)# spanning-tree portfast default
```

- Ezek a parancsok nem is kikapcsolják az STP-t, hanem csak a kapcsoló nem veszi figyelembe az eredményét az adott portokon!

FESZÍTŐFA VÉDELME

- Ha portfast-tal kikapcsoljuk egy porton a feszítőfa protokollt, attól még az adott porton fut az STP és a felhasználó megfelelő üzenetekkel átállíthatja a feszítőfát. Ez ellen véd a bpduguard funkció. Ez nem engedi portfast típusú interfészekről módosítani a feszítőfát. Lehet állítani interfészenként:

```
S1 (config-if) #spanning-tree bpduguard enable
```

- Itt is lehetséges az összes portfast interfész védelme alapértelmezetten:

```
S1 (config) #spanning-tree portfast bpduguard default
```