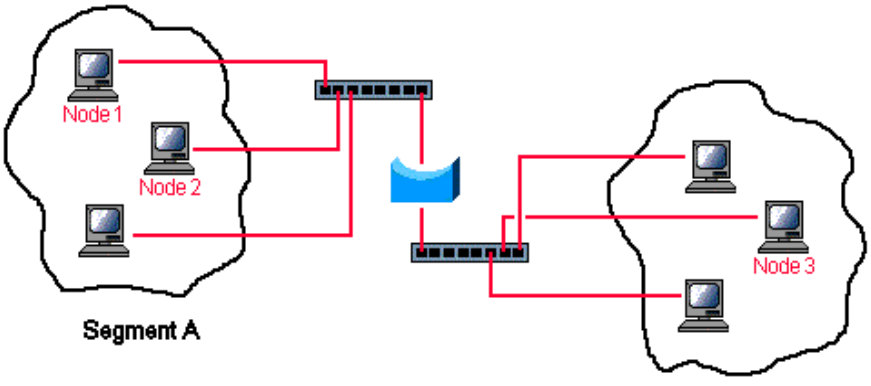
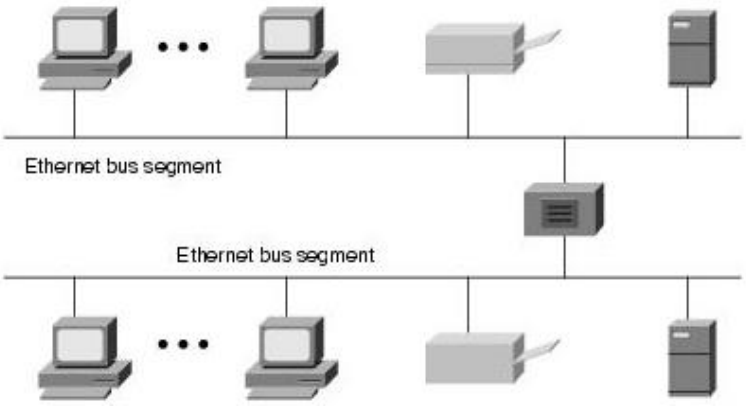
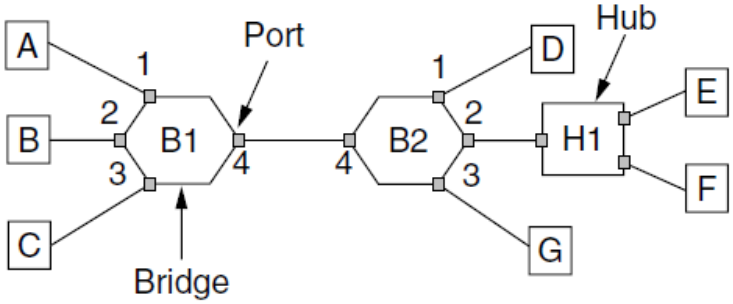
17. fejezet – Kapcsolódás az adatkapcsolati rétegben

Kapcsolódás az adatkapcsolati rétegben

A helyi hálózatok egyre nagyobb térhódítása szülte azt az igényt, hogy az önálló LAN hálózatokat valamilyen módon össze kéne kapcsolni egy nagyobb LAN hálózattá. A későbbiekben látni fogunk olyan gyakorlati példákat, melyek ezt a kérdést egy réteggel feljebb (a hálózati rétegben az útvonalválasztást IP címek segítségével) oldják meg, de most kövessük a LAN hálózatok evolúcióját, és vizsgáljuk meg a hálózatok összekapcsolásának lehetőségét az adatkapcsolati rétegben.
Az adatkapcsolati réteg két legjellemzőbb aktív hálózati eleme a Bridge, és a Switch. A két eszköz közti lényegi különbség a portok számában van, a Bridge jellemzően 2-3 portos, a Switch jellemzően legalább 4 portos eszköz. Funkciójukat és felhasználásukat tekintve a Bridge gyakorlatilag csak a szegmensek összekapcsolására használható, míg a Switch e mellett a saját szegmensének (vagy szegmenseinek) hosztjaihoz is kapcsolódik. Segítségükkel különböző típusú hálózatok is összekapcsolhatók.







Mindkettő eszköz a MAC címek alapján irányítja a kereteket az adó hoszttól a vevő hosztig. Az adatkapcsolati réteg aktív eszközei emiatt nem is tudnak különbséget tenni a LAN hálózat egyes alhálózatai között, azaz ilyen módon alhálózatok nem hozhatók létre, csak az önálló szegmensek kapcsolhatóak össze.

A szegmensek összekapcsolásának jellemzően szervezési, morfológiai okai vannak. Több épület esetében az egyes épületek akkor is összeköthetőek megfelelő optikai kábellel, ha a vezetékes hálózat 100m-es távolsághatárán túl helyezkednek el. Ez esetben is csak a szegmensek összekapcsolása segít, hiszen a hálózat szabványban rögzített maximális hosszát túllépni az eddig megtanultak szerint nem bölcs dolog.

A szegmensek összekapcsolása ráadásul nem jár a hálózat teljesítményének csökkenésével, hiszen a keretek ez után is mindig csak a megfelelő portokra lesznek irányítva. Ráadásul egy szegmens meghibásodása (általában) nem rántja magával a többi Bridge segítségével összekapcsolt szegmenst, mivel a Bridge-k képletesen szólva úgy viselkednek, mint a tűzbiztos ajtók egy épületen belül – amíg nincs tűz teljesen átjárhatóak, tűz esetén viszont lezárnak.

Ahhoz, hogy ezeket az előnyöket egy Bridge valóban teljesíteni is tudja, a Bridge a következő módon kell hogy működjön. Mikor a Bridge egyik portjára adatkeret érkezik, kiolvassa az adatkeretben szereplő MAC címet. Ha ez a cím már szerepel a MAC táblájában (Forwarding Table), mely a fizikai címeket portokhoz rendeli – azaz megmutatja, melyik eszköz mely portjára csatlakozik – akkor csak arra a portra továbbítja az adatkeretet. Ha a cím még nem szerepel a MAC táblában, akkor minden portjára elküldi a keretet, majd figyeli, melyik porton válaszol a címzett hoszt, és ennek alapján készít egy új bejegyzést a MAC táblában. (Az első generációs Bridge-k esetében a táblát még manuálisan kellett kitölteni…)

Ez a tanulási módszer az úgynevezett átlátszó híd (Transparent Bridging) technika. Ennek segítségével egy Bridge azonnal, minden szoftveres vagy hardveres beállítás nélkül azonnal használatba vehető. Maguk az összekapcsolt szegmensekben lévő hosztok nem is tudnak különbséget tenni abban, hogy egy szegmensen belül, vagy különböző szegmensekben helyezkednek el.

Az átlátszóság megvalósításához két algoritmus használatos, a hátrafelé tanulás (Backward Learning) és a feszítőfa (SPT – Spanning Tree). Előbbi abban segít, hogy az adatforgalom nem kerülhessen olyan helyre, ahol arra semmi szükség sincs, az utóbbi pedig a kábelezés problémáit, a hurkok okozta problémákat hivatott megoldani. Az algoritmusok ok-okozati viszonyainak megértéséhez nem szabad arról megfeledkeznünk, hogy az ipari gyakorlatban a hálózatok szinte sohasem statikusak, a topológia, az éppen kapcsolódó hosztok helye és száma dinamikusan változhat.

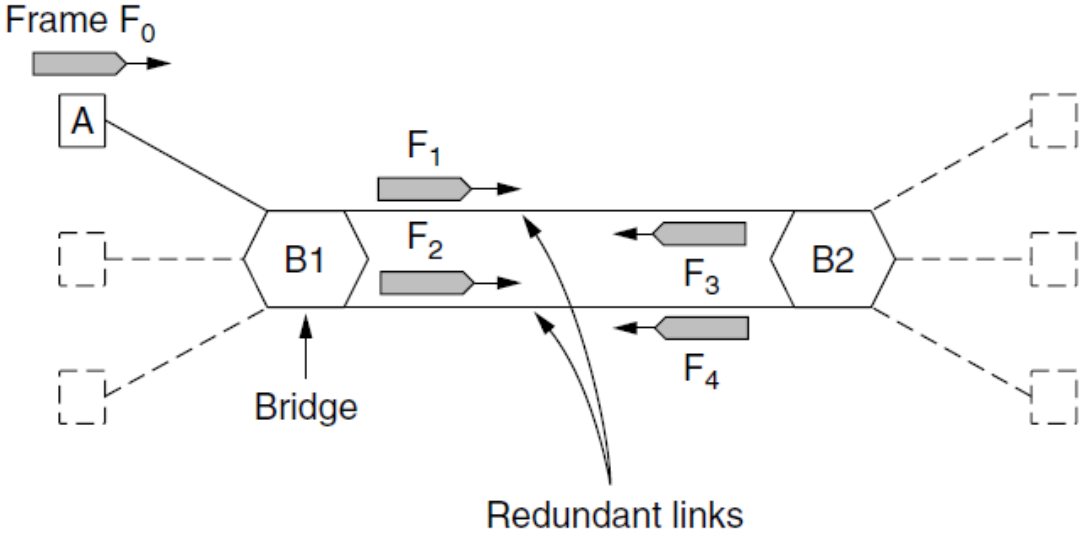
Hátrafelé tanulás (Backward Learning)

A hátrafelé tanulás a problémát a következő képpen oldja meg. A hálózat előbb említett dinamikus változása egyben azt is jelenti, hogy a Bridge MAC táblája is dinamikusan kell hogy megváltozzon, frissüljön, lehetőleg automatikusan, emberi beavatkozás nélkül. Ehhez arra van szükség, hogy a Bridge belső szoftvere folyamatosan ellenőrizze a táblát, és abból a régi (a gyakorlatban ez néhány percet jelent) bejegyzéseket törölje. Igaz, hogy ennek következtében a Bridge a hálózatot szinte folyamatosan Broadcast keretekkel terheli, viszont akkor is rövid időn belül normálisan akkor is elérhető egy hoszt, ha másik portra vagy másik (pl. WLAN) szegmensbe került.
Normál működés mellett tehát a Bridge szempontjából három működési elv lehetséges.

- Ha a cél hoszthoz tartozó port és a forrás port azonos, akkor a keretet el kell dobni. (Ez az első ránézésre valószerűtlen helyzet elő tud fordulni például az első oldal legalsó ábráján, amikor az „E” és az „F” hosztok között zajlik forgalom, hiszen az a HUB miatt a „B2” jelű Bridge-be is eljut.)
- Ha a cél hoszthoz tartozó port és a forrás port különböző, akkor a keretet a megfelelő portra továbbítani kell.
- Ha a cél hoszthoz tartozó port ismeretlen, akkor az összes portra – kivéve a forrás portját – adatszórással kell a keretet elküldeni.

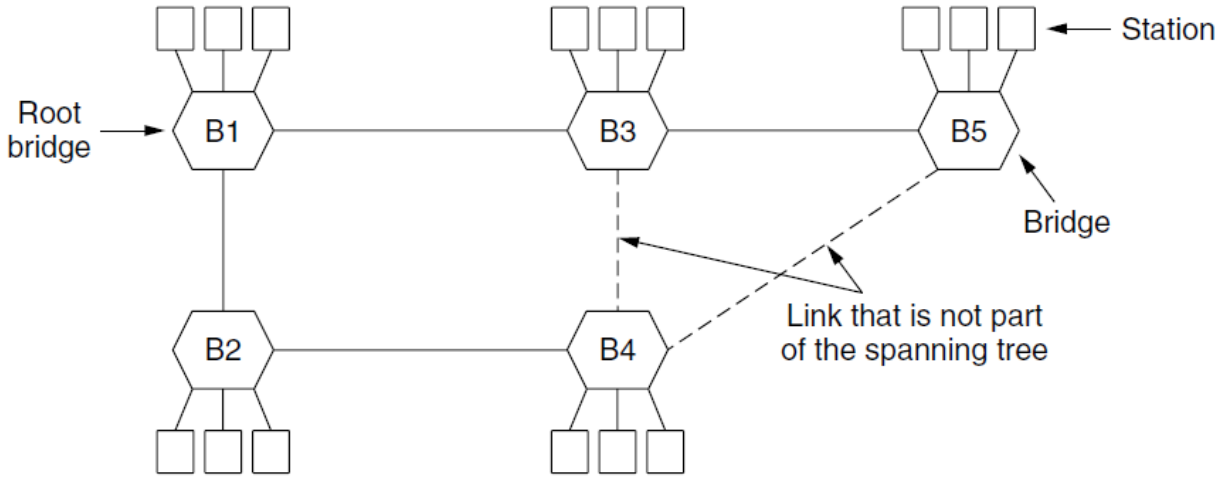
Feszítőfa (Spanning Tree)

A Spanning Tree (továbbiakban már csak az angol rövidítést használom) algoritmusát Radia Perlman, a DEC (Digital Equipement Corporation) mérnöke dolgozta ki 1985-ben, az algoritmus szabvány szerinti elnevezése IEEE802.1D. (Két érdekesség: A feladatra egyébként egy hetet kapott, de Ő egy nap alatt megoldotta. A DEC céget 1998-ban felvásárolta a COMPAQ, a COMPAQ-ot viszont 2002-ben felvásárolta a HP…)
A SPT kidolgozásának közvetlen oka az volt, hogy miközben a hálózatok összekapcsolásakor a minél nagyobb megbízhatóságra törekedtek, váratlan problémák merültek fel. A nagyobb megbízhatóságot redundáns kapcsolatok kiépítésével szándékozták elérni. Ez valóban közvetlen biztonságot jelent például kábelszakadás esetére, de egyben hálózati hurok is képződik a topológiában.
Amennyiben két LAN szegmenst olyan Bridge-k segítségével kötünk össze, melyek az összeköttetéshez két portjukat is használják, szükségképpen hurkot alakítunk ki.



Tekintsük át a felmerülő problémát. A fenti ábra szerint az „F_0” keret címzettje egy korábban ismeretlen, azaz a MAC táblában nem szereplő hoszt, aminek a port számát a „B1” Bridge nem ismeri, ezért a korábban lefektetett elvek szerint a „B1” Bridge a forrás (azaz az „A” hoszt) portjának kivételével minden portjára elküldi a keretet. A „B2” Bridge irányába így két keret, az „F_1” és az „F_2” is el fog indulni. A „B2” Bridge veszi a két beérkező keretet, de azzal nincs tisztába, hogy ugyanannak a keretnek két példányával találkozik. Tekintettel arra, hogy a keretek címzettje a „B2” Bridge MAC táblájában sem szerepel, szintén a korábban lefektetett elvek szerint jár el, azaz a forrás kivételével minden portjára elküldi az egyes kereteket. Így tehát az „F_1” keretből készít egy „F_4” keretet, az „F_2” keretből pedig egy „F_3” keretet, és ezeket megfelelő portjaira el is küldi. A folyamat ezután egy végtelen ciklusba kerül, hiszen „B1” Bridge hasonló módon visszaküldi a kereteket. A végtelen ciklus hálózati torlódást okoz, és gyakorlatilag megbénítja a hálózat működését.

A megoldás a SPT protokoll használata, mely működése során a fa struktúrához hasonló hurokmentes topológiát hoz létre, a redundáns élekkel létrehozott gráfból. Ehhez elengedhetetlen, hogy a Bridge-k kommunikáljanak egymással, és a hierarchikus fa struktúrát a gyökértől (Root) kezdve felépítsék. A kommunikáció egy speciális kerettel, a BPDU (Bridge Protocol Data Unit) kerettel történik. A hierarchia felépítésében és a prioritás kialakításában is a MAC címek játsszák a főszerepet, ugyanis kellő számú üzenet cseréje után a legkisebb értékű MAC azonosítóval rendelkező Bridge lesz a gyökér. A folyamat úgy játszódik le, hogy amennyiben „B1" Bridge kisebb értékű MAC címet hirdet, mint a „B2" Bridge, akkor a „B1" Bridge lesz a gyökér, és a „B2" Bridge nem is folytatja tovább saját MAC címének BPDU keretekben történő hirdetését, hanem elfogadja „B1" Bridge-t, mint gyökér Bridge-t. A következő lépés a gyökértől a többi Bridge felé a legalacsonyabb útköltségű (jellemzően a legrövidebb) útvonal kijelölése, azaz a fa meghatározása. Az útköltség számítása a vonal sebessége és a vonalak darabszáma segítségével történik. Újabb adatcsere után minden Bridge tárolja a gyökérhez vezető legrövidebb és egyben egyetlen utat, azokat a portjaikat pedig, amelyek nem szerepelnek a gyökérhez vezető legrövidebb úton kikapcsolják a normál adatforgalomból, azokon a továbbiakban csak BPDU kereteket fogadnak.



A lezárt portokon a BPDU keretek fogadására azért van szükség, mert ez biztosítja, hogy ha egy aktív összeköttetés vagy készülék meghibásodik, akkor új SPT-t lehessen létrehozni. A hálózati kapcsolóelemek ritkábban változtatják a helyüket, mint a hosztok, mégsem tekinthetjük a SPT egyszeri felépítését kőbe vésett modellnek. Az algoritmus ciklikusan tovább fut, ellenőrizve az esetleges hardver illetve útvonal megváltozásokat.

Az SPT támadhatósága

Mivel az SPT egy bizalomteljes, gyakorlatilag állapotmentes, autentikáció nélküli protokoll, ezért a működése könnyen befolyásolható speciális „támadó” BPDU keretek injektálásával. Természetesen ez esetben (mint például a vírusok esetében) is már csak a támadások után születtek meg a védelmi módszerek. A támadásra három módszer is ismeretes.

A „Yersinia” néven ismert eljárás során a támadó egy olyan BPDU keretet juttat a hálózatba, amely az aktuális gyökér Bridge MAC címénél kisebb értéket tartalmaz, így átveheti a gyökér szerepet, megváltoztatva hálózat logikai topológiáját. Ez természetesen a hálózati teljesítmény romlásához vezet, de segítségével például lehetőség nyílik a hálózati forgalom lehallgatására. Ez a támadás (ma már) két módszerrel is kivédhető. A Root Guard (Gyökér Védelem) portvédelmi funkció megakadályozza gyökér pozíció megváltoztatását. A BPDU Guard (BPDU Védelem) pedig lehetővé teszi, hogy a hálózatban az SPT határai beállíthatók legyenek, így a hálózati topológia megjósolhatóvá válik. Azok az eszközök, melyek egy BPDU Guard funkcióra beállított port mögött vannak, nem képesek átalakítani az SPT topológiáját.

A másik támadási mód a DoS (Denial of Service / Szolgáltatásmegtagadással Járó Támadás), a túlterheléses támadás. A támadó ez esetben másodpercenként akár több tízezer BPDU kerettel árasztja el a hálózatot, amit az eszközök még 100%-os CPU felhasználás mellet sem képesek feldolgozni, így a hálózati forgalom akadozni fog vagy teljesen meg is bénulhat. Szerencsére ez a támadás is védhető a BPDU Guard segítségével.

A harmadik támadási mód egy Switch szimulációs támadás, amikor a támadó olyan hosztot használ, mely két darab hálózati kártyával rendelkezik, és olyan BPDU kereteket juttat a hálózatba, ami a teljes forgalmat átirányítja a hoszton keresztül. Ez az úgynevezett MITM (Man In The Middle / Közbeékelődéses Támadás), mely így akár titkos adatok megszerzését is lehetővé teszi. Ez a támadás is védhető a fenti módszerekkel.

Természetesen a legfontosabb az, hogy a rendszergazda (vagy a rendszer felügyeletével megbízott személy) folyamatosan ellenőrizze a hálózat működését, mert legkönnyebben mindig a „magára hagyott” illetve a nem megfelelően konfigurált hálózat támadható.

Virtuális LAN (VLAN) hálózatok

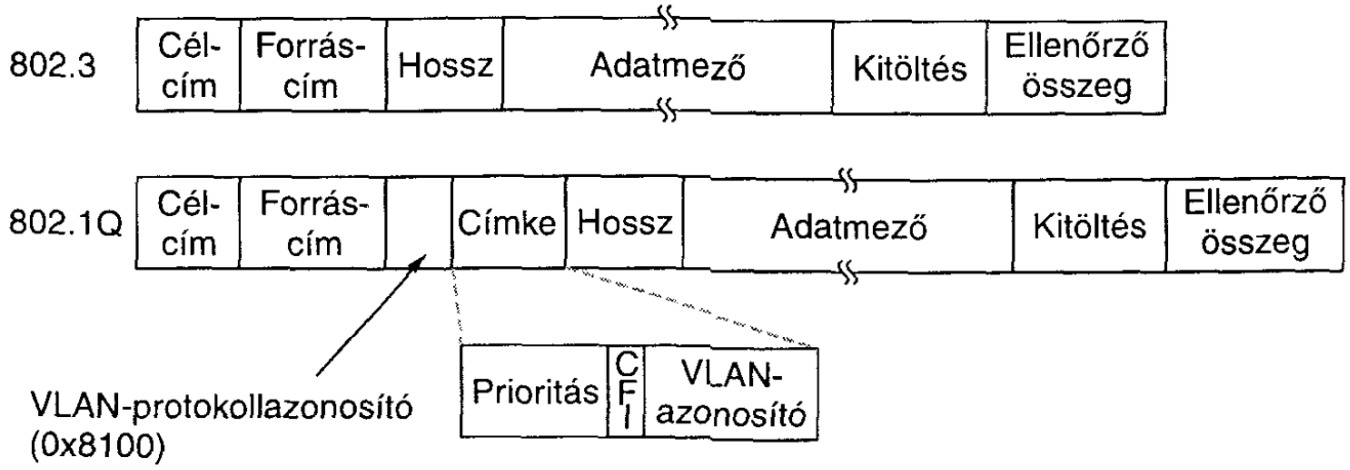
A LAN hálózatok esetében eddig leginkább a bővítés lehetőségeiről volt szó. Az eddig ismertetett módokon lehetőség nyílt több szegmens összekapcsolására, nagyobb hálózatok létrehozására. Így elérhető vált egy vállalat esetében, hogy az összes gépe egy nagy lokális hálózathoz tartozzon. Természetesen ebben az esetben az összes szegmens összes gépe „látta egymást", azaz képes volt adatforgalmat bonyolítani, kereteket cserélni.

Az egymás „látásából" fakadó előny azonban, hátránnyá is tud válni. Egy vállalat belső hierarchiája megköveteli az adatok biztonságát, illetve az adatok elérhetőségének különböző szintjeit. Egyrészt egyes szervezeti egységek adatai (például személyzeti osztály, pénzügyi osztály, kutatás-fejlesztés) nem tartoznak más osztályokra, leginkább nem tartoznak az érintetteken kívül senki másra. Másrészt az egyes szervezeti egységek kisebb, mások nagyobb adatforgalmat bonyolíthatnak, azaz különböző hálózati kapacitást igényelnek. Arról sem szabad megfeledkeznünk, hogy több szegmens esetében nyilván megnő a Bridge-k által adatszórásként küldött keretek száma is.

Az adatkapcsolati rétegben a célt csak a hálózat logikai szétdarabolását érhetjük el. A virtuális hálózatokról (VLAN) az IEEE802.10, illetve az azt felváltó IEEE802.1Q szabvány rendelkezik. Egy VLAN megvalósításához legelőször is a szabvány előírásait teljesítő, azaz VLAN képes Switch-ekre van szükség. Az elv tulajdonképpen az, hogy a hosztok fizikai elhelyezkedésétől függetlenül, a központi Switch-ben konfigurálható módon lehet az egyes hosztokat egyik vagy másik VLAN hálózathoz rendelni. A hozzárendelés a Switch-ben port szintű, de egy port (azaz a hozzá kapcsolódó hoszt) akár több VLAN-ban is szerepelhet.

Layer 2 Switch használata esetén a működés lényege az, hogy a keretek csak az érintett VLAN-(ok)on belül mozognak, azon kívül nem. Egy adott VLAN címkével (VLAN ID) ellátott keret csak a vele megegyező VLAN címkéjű portokon keresztül érkezhet, és csak ugyanilyen porton keresztül távozhat a megcímzett hoszt felé. A Layer 3 Switch neve kicsit megtévesztő, hiszen alapvetően ez is a Layer 2-ben, keretekkel dolgozik, de azon belül gyakorlatilag egy Layer 3 funkcionalitást valósít meg, azaz képes az egyes VLAN-ok közötti forgalomirányításra.

A VLAN címkézés megvalósításához azonban szükség volt az Ethernet fejléc megváltoztatására az IEEE802.1Q szabvány szerint. Egy ilyen elterjedt és széles körben használt dolog, mint az Ethernet fejléc megváltoztatása nem zökkenőmentes feladat. A szabvány megjelenése előtti eszközeit nyilván senki sem akarta eldobni, viszont a szabvány megjelenése után készült eszközöknek – különböző gyártók esetében is – kompatibilisnek kell lenniük.



Az egyetlen változás az új keret szerkezetben két darab 2 bájtos mező beépülése.
Az első 2 bájtos mező a VLAN protokoll azonosító (VLAN Protocol ID vagy TPID – Tag Protocol Identifier), melynek értéke fixen mindig 0x8100. Jelentése csupán annyi, hogy a keret VLAN információt hordoz. Mivel az érték nagyobb, mint 0x0600, a korábban megismertek szerint nem hosszként, hanem típusként kerül értelmezésre. Sajnos a régebbi kártyák egy része emiatt nem is képes a VLAN támogatásra.
A második 2 bájtos mező, a TCI (Tag Controll Information / Cimkekontroll Információ), ami három almezőt tartalmaz:

- A 3 bites prioritás (Priority) infromáció, ami tulajdonképpen nem is kötődik a VLAN-hoz. Viszont, ha már egyszer hozzányúltak a keret formátumához, majd csak jó lesz a későbbiekben valamire alapon létre lett hozva. Azóta már használatba is lett véve a valós idejű átvitel megjelöléséhez, amire például a beszédátvitelnél van szükség.
- Az 1 bites CFI (Canonical Format Indicator / Kanonikus Formátumjelző), ami szintén nem a VLAN-okhoz kötődik. Ez azt jelzi, hogy az adatmező egy nem módosítható, IEEE802.5 szabvány (Token Ring) szerinti keretet tartalmaz, azaz a keret végső célállomása nem is ebben az Ethernet hálózatban keresendő, a keret csak „átutazóban” van itt.
- A VLAN azonosító (VID – VLAN Identifier) ami az utolsó 12 bitet foglalja el. Ez jelzi azt, hogy a keret melyik VLAN-ba tartozik.

Az IEEE802.1Q szabvány sajátos helyzetet teremtett a hálózatok világában. Emlékezve arra, hogy az IEEE802.3 szerint meghatározott maximális (előtag nélküli) Ethernet keretméret 1518 bájt, ha egy maximális méretű keret címkézésre kerül, akkor ez a keret már 1522 bájtos lesz, ami megsérti az IEEE802.3 szabványt. Ennek feloldására a IEEE802.3-as bizottság létrehozott egy alcsoportot IEEE802.3ac néven, a maximális keretméret 1522 bájra való kiterjesztésére.