



Türkçe CCNA EĞİTİM Notları

INTERNETWORKING TEMELLERİ

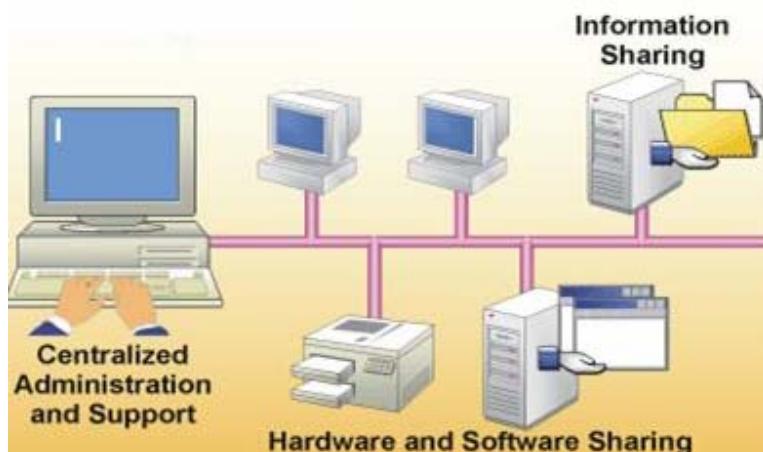
Network Nedir, Ne İşe Yarar?

Birden fazla bilgisayarın çeşitli sebeplerden dolayı birbirlerine bağlandığı yapıya network (ağ) denir.

Bir çok bilgisayarın aynı yapı içerisinde bulup birbirleriyle haberleşebiliyor olması çok ciddi yararlar sağlar. Bilgi paylaşımı, yazılım ve donanım paylaşımı, merkezi yönetim ve destek kolaylığı gibi konular göz önüne alındığında birden fazla bilgisayarın bulunduğu ortamlarda artık bir network kurulması zorunlu hale gelmiştir diyebiliriz.

Networklerin kurulmasıyla birlikte disketle data taşıma devri bitmiş, tek tuşla istenilen bilgiye ulaşma kolaylığı meydana gelmiştir. Bir veya birkaç yazıcı ile bir işletmenin bütün print ihtiyaçları da yine network sayesinde karşılanabilmektedir.

Yönetim ve destek hizmetleri kolaylaşmış, network yöneticisi tek bir bilgisayardan çok daha hızlı bir şekilde bütün networkü izleyebilir ve sorunları çözebilir hale gelmiştir.



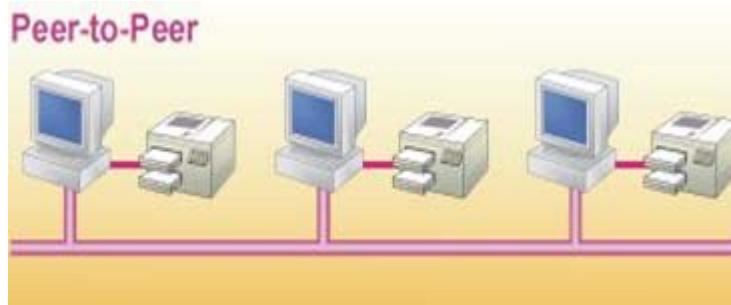
Bilgisayarlar networklerde çeşitli görevler üstlenebilirler. Genel olarak bir bilgisayar bir networkte client (istemci) ya da server (sunucu) rollerinden birini üstlenir.

Network ortamında paylaşılan yazılım ve donanımlara sahip bir bilgisayara server ya da Ana Bilgisayar denir. Burada Server sahip olduğu kaynakları istemci bilgisayarların kullanımına açarken bazen de tüm verinin toplandığı ana merkez konumundadır.

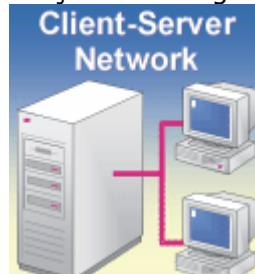
Network ortamında kaynak ya da veri isteyen bilgisayarlara ise Client adı verilir. Client sadece kendisinden donanımsal olarak büyük olan Server lardan değil gerektiğinde diğer client' lardan da kaynak ya da veri talebinde bulunabilir.

Network Tipleri

Networkler Peer To Peer ve Client/Server mimarisi başlıkları altında incelenebilirler. Peer To Peer networklerde ana bir bilgisayar yoktur. Bütün bilgisayarlar eşit haklara sahiptir ve yeri geldiğinde iletişime geçtikleri bilgisayarlara bir Client – Server yapısında hareket ederler. Her bilgisayar kaynaklarını ya da sahip olduğu datayı istediği kadariyla kullanımına açabilir ya da açmaz.



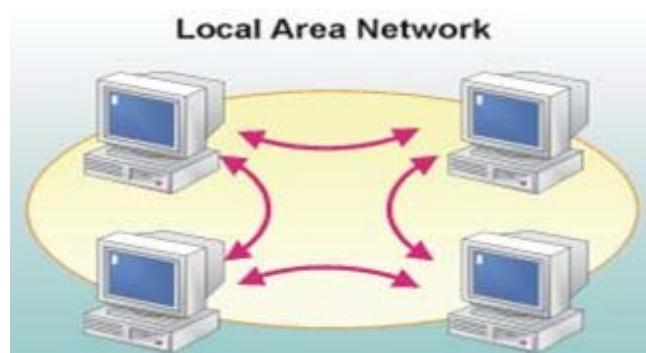
Client / Server mimarisinde is, adından da anlaşılacağı gibi hem donanımsal hem de yazılımsal olarak diğer bilgisayarlardan üstün, atanmış bir ana bilgisayar vardır.



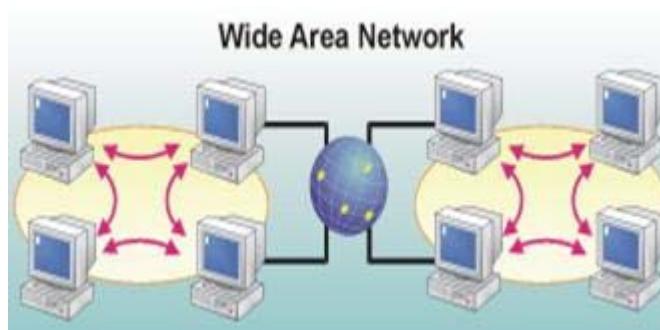
Burada Server olarak atanmış bilgisayarın yetersiz kaldığı durumlarda networke başka serverlarda dahil edilebilir. Örneğin gelişmiş bir networkte Mail Server'ın, DHCP ve DNS gibi serverların farklı bilgisayarlarda bulunması performansı olumlu yönde etkileyebileceğini önerilebilir.

LAN, WAN ve MAN

Ağlar büyüklüklerine göre LAN (Local Area Network), WAN (Wide Area Network) ve MAN (Metropolitan Area Network) olmak üzere üçer ayrırlar.

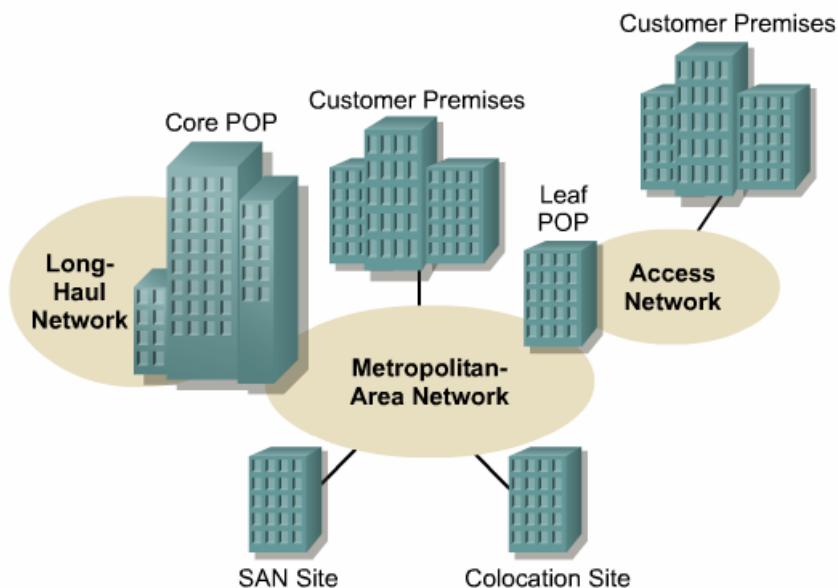


Birbirine yakın yerlerde konumlandırılmış ve kablolar ile fiziksel olarak birbirlerine bağlanmış yapıdaki networkler LAN olarak adlandırılırlar. Örneğin bir binada bulunan bütün bilgisayarların birbirlerine bağlanmasıyla oluşan yapı bir Local Area Network' tür.



İki yada daha fazla LAN'ın birbirlerine telefon hatları, kiralık hatlar yada benzer yollardan birbirlerine bağlanmasıyla oluşan yapı ise Wide Area Network olarak adlandırılır. Burada bilgisayarların fiziksel olarak birbirlerine yakın olmalarına gerek olmadığı gibi çok uzakta olabilirler.

MAN ise, kavram olarak açıklaması zor olmakla birlikte, örneğin bir şehir ya da bir bölgenin iki ayrı LAN ile birleşmesi gibi düşünülebilir.



Örnek vermek gerekirse birden fazla şubesinin bulunan bir bankanın kullanabileceği bir yapı diyebiliriz.

Ethernet Teknolojileri

Bilgisayarların bir networke bağlanıp veri alış verişinde bulunabilmesini sağlayan elektronik devredir. Farklı yerlerde Ethernet kartı, network kartı, ağ kartı yada NIC şeklinde isimlendirmeleri yapılmıştır. NIC, İngilizce Network Interface Card'ın kısaltmasıdır.

Her Ethernet kartının üretiminden itibaren kendine ait farklı bir tanımlama numarası vardır ve bu sayede diğer bütün kartlardan ayırt edilebilir. Bu tanımlama numarasına MAC adresi (Media Access Control) ya da Fiziksel Adres denir ve 6 oktet, 48 bitten oluşur. Bu 6 oktetin ilk 3 oktetinin Internet Assigned Numbers Authority (IANA) tarafından belirlenir. Bir firma Ethernet Kartı üretmeye karar verirse ilk başvuracağı yer IANA'dır. IANA firmaya o firmanın ID'si gibi düşünebilecek 3 oktetli bir sayı verir son oktetini de firmaya bırakır. Bu şekilde bir standart sağlanırken aynı MAC adresine sahip Ethernet kartlarının üretilmesi de engellenmiş olur.



(RJ-45 ile sonlandırılmış UTP kablonun Ethernete takılması)

Bir bilgisayarın MAC adresi komut satırında "ipconfig /all" yazılarına öğrenilebilir. (Windows 9x ortamında ipconfig.exe yerine Winipcfg.exe kullanılır.)

Bir sistem yönetici kendi bilgisayarından diğer bilgisayarların MAC adreslerini de öğrenmek isteyebilir. Sözelimi DHCP ile ip konfigürasyonunu dağıtmak ve bazı bilgisayarların her seferinde aynı ip'yi almasını sağlamak için MAC adreslerini kullanarak DHCP' nin Reservations özelliğini kullanmak isteyen bir yöneticinin her bilgisayarı tek tek dolaşarak ipconfig /all komutunu kullanması ve MAC adreslerini not alması çok uzun ve yorucu olur.

Bu durumda sistem yöneticilerinin yardımına ARP (Address Resolution Protocol) protokolü yetişir.

Bir bilgisayara en az bir kere ulaşmış olmak kaydıyla, komut satırında "arp -a" yazılarak o bilgisayarın MAC adresi öğrenilebilir.

```
C:\Documents and Settings\Cisco>ping 192.168.1.4
32 bayt veri ile 192.168.1.4 'ping' ediliyor:
192.168.1.4 cevabı: bayt=32 süre=2ms TTL=60
192.168.1.4 cevabı: bayt=32 süre=2ms TTL=60
192.168.1.4 cevabı: bayt=32 süre=2ms TTL=60
192.168.1.4 cevabı: bayt=32 süre=2ms TTL=60

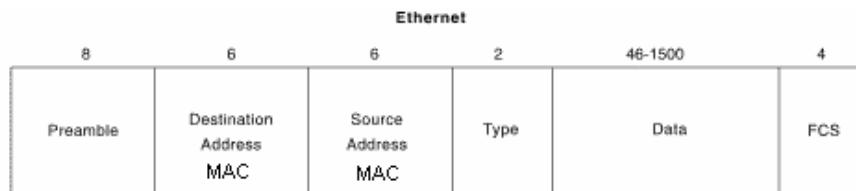
192.168.1.4 için Ping istatistiği:
  Paket: Giden = 4, Gelen = 4, Kaybolan = 0 (0% kayıp),
  Mili saniye türünden yakalılan tur süreleri:
    En Az = 2ms, En Çok = 2ms, Ortalama = 2ms

C:\Documents and Settings\Cisco>arp -a
Arabirim: 192.168.1.172 --- 0x10003
  Internet Adresi      Fiziksel Adres      Tipi
  192.168.1.1          00-13-10-3f-c8-c8      dinamik
  192.168.1.2          00-10-5a-65-7e-f0      dinamik
  192.168.1.4          00-80-77-6f-1b-89      dinamik

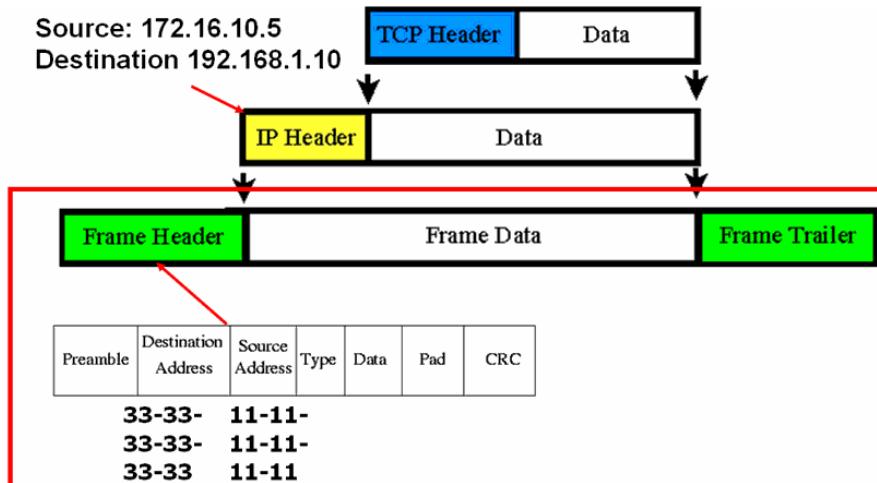
C:\Documents and Settings\Cisco>
```

Komut satırından alınmış şekilde, önce 192.168.1.4 bilgisayarına bir kez ulaşmak için ping atılmış. Daha sonra kullanılan “arp- a” komutu ile oturum boyunca ulaşılan tüm bilgisayarların bellekteki fiziksel adres tablosuna erişilmiştir.
(ARP Protokolü ilePOSITE deneysel olarak çalıştırılmıştır.)

Ethernet teknolojiler IEEE 802.3 standartı ile tanımlanmıştır ve ethernetin datayı帧 (frame) e'ler halinde taşıdığı söylenebilir. Genel olarak ethernet frame'leri aşağıdaki gibidir.



Destination ve Source adresler hedef ve kaynak cihazların fiziksel adreslerini ifade eder. FCS değeri ise datanın sağlamlı iletilip iletilmediğinin kontrol edilmesini sağlayan bir değerdir.



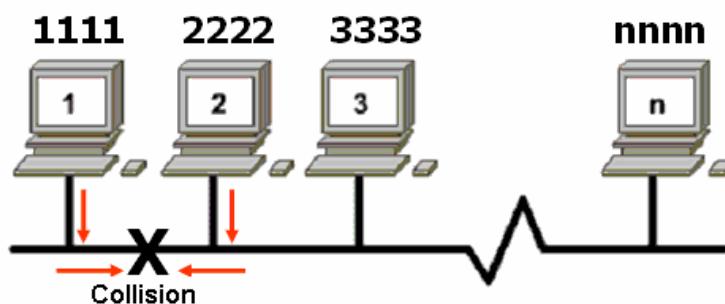
OSI Referans modeli içerisinde detaylı anlasılabiliecek datanın iletimi sırasının data katmanları ilerler. Ve her katmanda data üzerinde o katmanın çalışma mantığı içerisinde gereken bilgi etiketlenir. Sekilde simdilik onemli olan kısmın sırasıyla TCP Header ve IP Header eklenir. LAN içerisinde data 2. katmanda, yani fiziksel adresler yardımıyla haberleşecek tır. Frame Header ile bu bilgiler dataya eklenir.

IP Header ve Frame Header arasındaki en önemli fark 'frame'lerin TTL (Time To Live) değerine sahip olmamasıdır. Dolayısı ile ikinci katmanda oluşturabilecek bir dengi dengisi yasayan cihazlar kapatılmadığı sürece devam edecektir.

CSMA/CD

Ethernet networkleri belli bir anda kabloyu hangi bilgisayarın kullanacağını CSMA (Carrier Sense, Multiple Access/Collision Detection) teknigiyle belirler. Bu teknikte paket gönderilmeden önce kablo kontrol edilir. Diğer bir iletişim oluşturduğu trafik yoksa iletişime izin verilir.

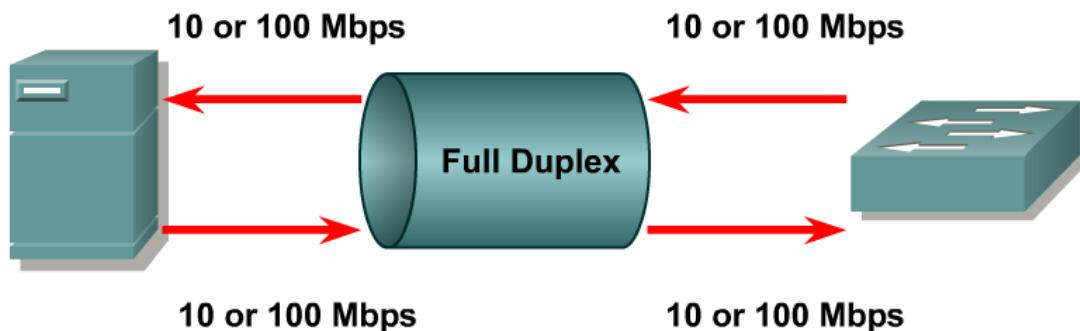
İki bilgisayarın birden kabloyu kullanmaya çalışması collision olarak adlandırılır. Her ikisinin de trafiği kaybolur. Ve hattın boş olduğu anı yakalamak için yeniden beklemeye ve hat dinlenmeye başlanır.



CSMA/CD networklerinde, beklemelerin çoğalmaması için bus olarak tanımlanan kablonun iki ucunun sonlandırılması gereklidir.

Full Duplex

Full Duplex ethernet aynı anda hem data iletimini hem de data alınmasını sağlar.



IEEE 802.3x ile tanımlanmış Full Duplex çalışma içerisinde, her iki cihaz da Full Duplex modda olduğu zaman sonuc alınabilecektir. Örneğin, host to switch, switch to switch yada switch to router bağlantılarında her iki tarafta Full Duplex modda çalışabiliyor olmalıdır.

Half Duplex çalışma içerisinde collisionlardan dolayı bant genişliğini yüzde 50 yada altı misi kullanılabılırken Full Duplex çalışma da bant genişliğinin tamamı kullanılabilir. Sozgelimi 10 Mbps olan bir hattın Half Duplex ile 5-6 Mbps' i kullanılabılır, oysa Full Duplex çalışma da ideal olarak aynı bant genişliğinde 20 Mbps'e ulaşıldığı varsayılmır.

Full Duplex içerisinde CSMA/ CD ' den bahsedilemez. Çünkü aynı anda hem iletim hem alım olabileceği için Carrier Sense olmayacağındır.

KABLO Standartları

Bilgisayarlar ethernet kartlarına takılan kablo aracılığıyla birbirine bağlanırlar. Networkün yapısına göre farklı özelliklerde kullanabilecek bir çok çeşit kablo standarı vardır. Ana başlıklar şöyledir;

- Koaksiyel (Coaxial)
- Twisted-Pair
- UTP (Unshielded Twisted-Pair / Koruyucusuz Dolanmış-Çift)
- STP (Shielded Twisted-Pair / Koruyuculu Dolanmış-Çift)
- Fiber-Optik

Koaksiyel Kablolar

Koaksiyel kablolar yaygın olarak kullanılan ağ kablolardır. Çok tercih edilmesi ve çok sık kullanılmasının başlıca nedenleri uygun fiyatı, hafifliği, esnekliği ve kolay kullanılmasıdır. Bir koaksiyel kablo bir iletken metal telin önce plastik bir koruyucu ile, ardından bir metal örgü ve dış bir kaplamadan oluşur. Bu koruma katları sayesinde iletilen verinin dış etkenlerden etkilenmesi minimuma indirgenmeye çalışılmıştır. Koaksiyel kablonun içinde kullanılan tek genellikle bakırdır.

Koaksiyel kablonun iki tipi vardır:

- Thin (thinnet)
- Thick (thicknet)

Thinnet (ince) koaksiyel kablo .25 inch genişliğindedir. Yaygın olarak kullanılır. Verileri sağlıklı olarak 185 metre uzağa iletibilirler. Thinnet koaksiyel kablolalar RG-58 standarı olarak değişik biçimde üretilmektedir.

Not: ThinNet ve ThickNet olmak üzere 2 çeşidi vardır. Thinnete 10Base2, Thicknete 10Base5 denir.

Thicknet ise daha kalın bir koaksiyel kablodur. Thicknet kablolalar 0.5 inch kalınlığındadır. Bu nedenler thicknet kablolalar daha uzun mesafe veri iletiminde kullanılırlar. 500 m mesafe için kullanılan thicknet koaksiyel kablolalar tipik olarak thinnet networkler için bir backbone oluşturmada kullanılır.

Mesafe	Koaksiyel kablo
185 m	Thinnet
500 m	Thicknet

- | Mesafe | Koaksiyel kablo |
|--------|-----------------|
| 185 m | Thinnet |
| 500 m | Thicknet |



10Base2



10Base5

Bir thinnet koaksiyel kabloyu thicknet kabloya bağlamak için ise transceiver denilen ara birim kullanılır. Transceiver'in network adaptörüne bağlanması için AUI ya da DIX

adı verilen çıkış kullanılır. AUI (Attachment Unit Interface) anlamındadır. DIX (Digital Intel Xerox) anlamına gelir.



AUI

Koaksiyel kabloların network adaptörüne bağlanması için, ayrıca iki kabloların birbirine eklenmesi BNC Konektörleri kullanılır.



BNC T Konnektörü

BNC kablo konnektörü kablonun ucunda yer alır. T konnektör ise koaksiyel kabloyu networkadaptörüne bağlamak için kullanılır. Barrel konnektör ise iki koaksiyel kablonun birbirine bağlanmasıını sağlar. Sonlandırıcılar ise kablonun sonunda yer alırlar.

Bus yerleşim biçiminde kurulan network'lerde kullanılan koaksiyel kablonun iki ucunda sonlandırıcı kullanılır. Bu sonlandırıcılar kablonun sonuna gelen sinyali yok ederler.

Twisted-Pair Kablolar

LAN'larda ve sınırlı veri iletiminde kullanılan bir diğer kablolama türü de twisted-pair kablolardır. Twisted-Pair (Dolanmış-çift) kablo iki telden oluşan bir kablodur. Twisted-pair kablolar iki türdür:

-UTP (Unshielded Twisted-Pair)



-STP (Shielded Twisted-Pair)



STP Kablo

10BaseT network'lerde ve diğer LAN ortamlarında yaygın olarak UTP kablolar kullanılır. Maksimum UTP kablo uzunluğu 100 m dir. UTP kablo iki izoleli bakır kablodan oluşur. UTP kablolar ayrıca telefon sistemlerinde de kullanılır.

10BaseT kablolar RJ-45 sonlandırıcıları ile sonlandırılırlar.



Fiber-Optik Kablolar

Fiber-optik kablolar verileri ışık olarak iletken yüksek teknoloji iletim ortamlarıdır. Fiber-optik kablolar hızlı ve yüksek kapasiteli veri iletimi için uygunudur. Özellikleri 100 Mbps hızında veri iletimi için kullanılır. Verilerin güvenliği açısından daha iyidir. Çünkü ışık olarak temsil edilen veriler başka bir ortama alınamazlar.

Fiber-optik kablo üzerinden veri aktarımı; ince fiber cam lifi (ışık iletkeni) üzerinden ışık dalgası şeklinde gerçekleştirilir. Aktarılacak her bir ışık işaretinin ayrı bir ince fiber cam kullanılır. Bu

çerçevede en basit hali ile bir Fiber-optik kablo 3 temel kısımdan oluşmaktadır: Işığın geçtiği tabaka olan Asıl Işık İletkeni, ışığı yansıma ve kırımlara karşı koruyan ve yine bir cam tabaka olan Cam Örtü ve tüm cam kısmı koruyan Koruyucu Kılıf olarak adlandırılabilenek kısımlardır. Uygulamada bunlara ilave olarak Fiber-optik kabloya; kablonun bina içi/bina dışı kullanım yeri ve şartlarına bağlı olarak çelik zırh yada jel tabakası gibi başka koruyucu ve esneklik kazandırıcı kısımlarda ilave edilebilmektedir.



Fiber Optic Kablo ve Sonlandırıcılar

Ethernet Kablolama Sistemi

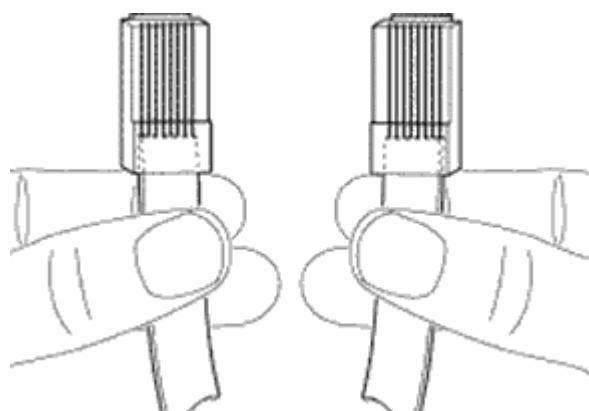
Ethernet network'lerinde dört çeşit kablolama sistemi kullanılır:

- Thick coaxial
- Thin coaxial
- Unshielded Twisted Pair
- Fiber-optic

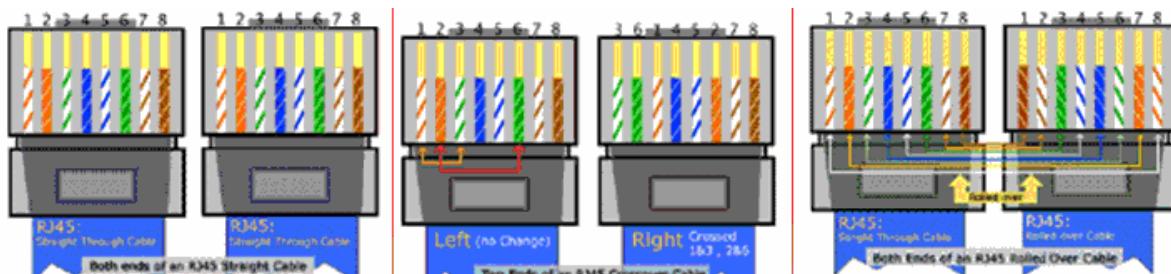
UTP Kablo Yapımı

UTP kablo yapmak hem kolay hem de eğlenceli bir istir. Bunun için kablo ve RJ-45 ile bir de Jack pensesine ihtiyacımız var.

Kablo yapılmırken dikkat edilecek unsurlardan biri de kablonun Duz, Cross yada Roll Over mi olacağı ve bu çeşitlerin renk sıraları.

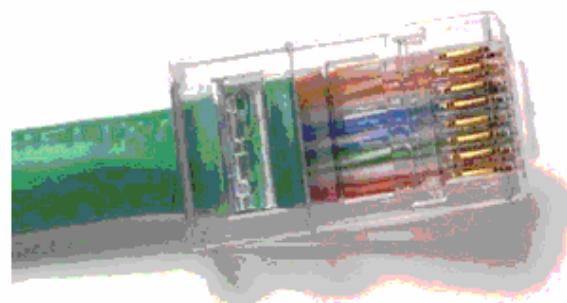


Kablolar sekilde ki gibi tutularak renk sıralarına bakılıp cinsi hakkında bilgi edinilebilir.

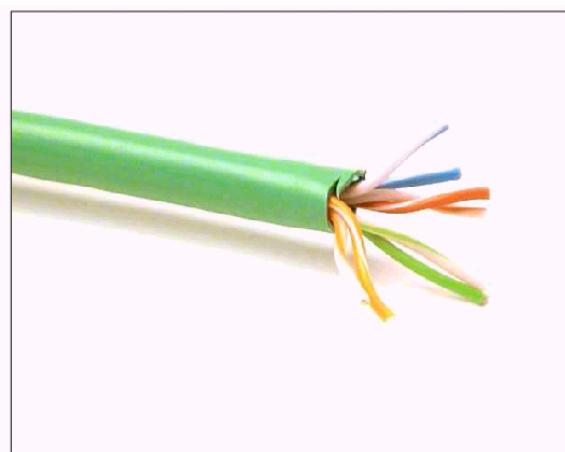
**Straight-through****Cross-over****Rollover**

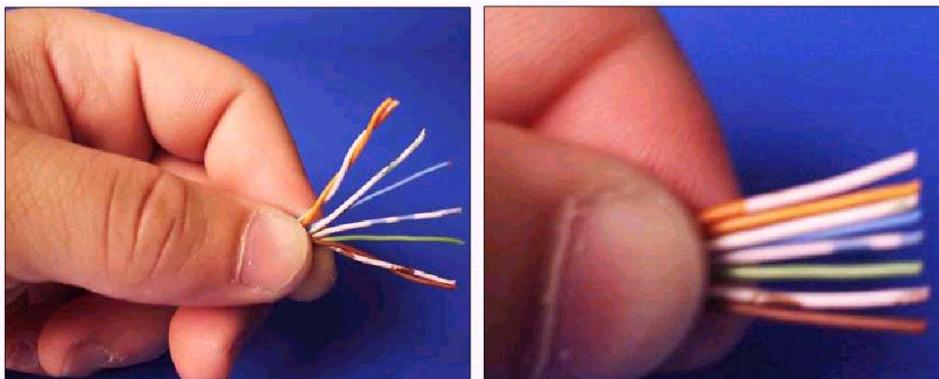
Sekilde üç çeşit kablonun renk sıraları görülmektedir.

Simdi adım adım kablomuzu yapacağız.

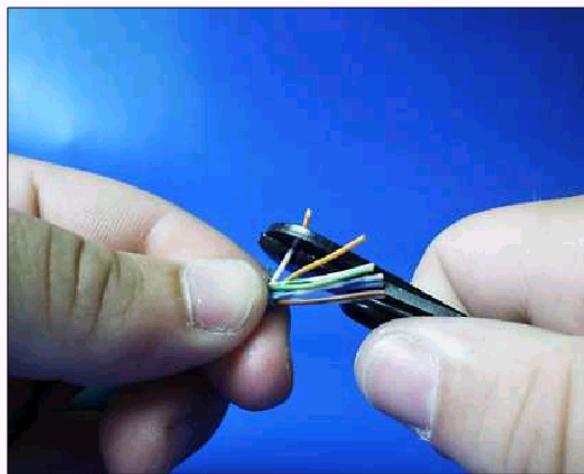
**sonra****simdi**

Kablomuzun ucundan resimdeki gibi bir kısmını soyıyoruz. Bunun için Jack Pensesi de kullanılabilir.

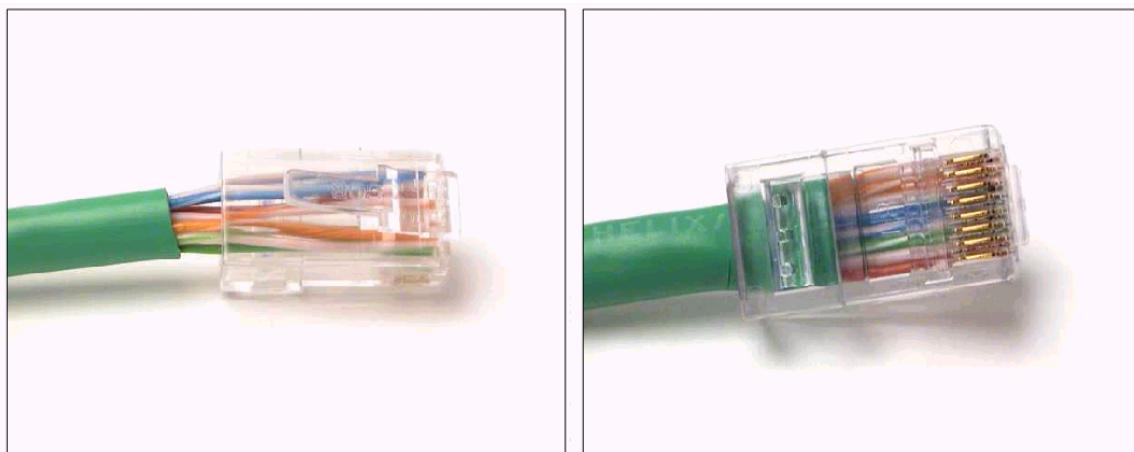




Yapacagimiz kablonun cinsine gore renk siralarina uygun sekilde kablo uclarini siralayip duzenliyoruz.



Kablo uclarini esit uzunlukta olacak sekilde kesiyoruz. Bunun icin de jack pensesi kullanilabilir. Butun bu islemler yapilirken kablonun tutugumuz yerini birakmamakta fayda var. Bir kere karisirsa bastan baslamak zorunda kalabiliriz.



İlk şekildeki gibi kablo uclarını RJ-45 içine yerlestirdikten sonra Jack pensesi ile sıkıştırıp kablonun RJ-45 e tam olarak oturmasını saglıyoruz.



Network Cihazları

Bir networkü sade bilgisayar ekleyerek genişletemeyiz. Bu bize kablolamanın zorlaşması, sinyal yayılması gibi sebeplerden sorun yaratır. Bu sebeple bir networkü genişletmek, güvenliğini sağlamak ve aynı zaman da hiyerarşî kazandırmak için bazı cihazlar kullanmalıyız. Bu cihazlar genel olarak şunlardır:

- Hub
- Switch
- Repeater
- Bridge
- Router
- Firewall
- Gateway

Hub

En basit network cihazıdır. Kendisine bağlı olan bilgisayarlara paylaşılan bir yol sunar. Yani Hub'a bağlı tüm cihazlar aynı yolu kullanırlar ve bu da aynı anda haberleşmek isteyen network cihazlarının, bir tek yol olduğu için hattın boşalmasını beklemelerine sebep olur. 8 – 12 – 16 – 24 portlu olarak üretilirler.



16 Portlu Cisco Hub

Switch

Kendisine bağlı cihazlara adından da anlaşılmacı gibi anahtarlamalı bir yol sunar. Hub ile kıyaslandığından en önemli farkı budur. İki bilgisayar kendi arasında haberleşirken başka bilgisayarlarda hattın anahtarlamalı kullanılmasından dolayı kendi aralarında iletişime gecebilirler. Bu sayede Hub'a göre daha yüksek bir performans sağlanacaktır. 8 -12 - 16 - 24 - 36 - 48 portlu olarak ya da şaseli üretilebilirler. Şaseli switchlerde boş yuvalar vardır ve gerektiğinde port eklenebilmektedir.



Cisco Switchler

Repeater

Repeater bir ethernet segmentinden aldığı tüm paketleri yineler ve diğer segmente yollar. Repeater gelen elektrik sinyallerini alır ve binary koda yani 1 ve 0'lara çevirir. Sonra da diğer segmente yollar. Bu yönyle repeater'in basit bir yükseltici olmadığını anlıyoruz. Çünkü yükselticiler gelen sinyalin ne olduğuna bakmadan sadece gücünü yükseltir. Yolda bozulmuş bir sinyal yükselticiden geçince bozulma daha da artar. Repeater ise gelen sinyali önce 1 ve 0'a çevirdiği için yol boyunca zayıflamış sinyal tekrar temiz 1 ve 0 haline dönüşmüş olarak diğer segmente aktarılır.



Cisco Repeater

Bridge

İki TCP/IP ağını birbirine bağlayan bir donanımdır. Fazla karmaşık aygıtlar olmayan bridge'ler gelen frame'leri (veri paketleri) alır ve yönlendirirler. Bridge'ler fiziksel bağlantının yanı sıra network trafigini kontrol eden aygıtlardır.

Bridge bir çeşit yönlendirme yapar diyebiliriz fakat OSI Katmanlarından 2. katman yani Data-Link Katmanında çalışmasıyla Router' dan ayrılır.

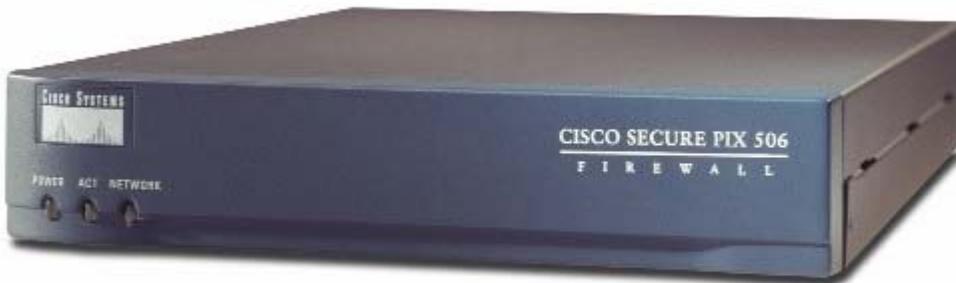


Cisco Bridge

Firewall

Türkçe güvenlik duvarı anlamına gelen firewall, özel ağlar ile internet arasında, her iki yönde de istenmeyen trafiği önleyecek yazılımsal ya da donanımsal sistemdir. Firewall'ların verimli bir şekilde kullanılabilmesi için internet ve özel ağ arasında ki tüm trafiğin firewall üzerinden geçmesi ve gerekli izinlerin / yetkilerin kısaca erişim listelerinin uygun bir stratejiyle hazırlanmış olması gereklidir.

Donanımsal firewall' lara verilebilecek en güzel örnek CSecurity derslerinde detaylı anlatılan Pix Firewall'dır.



Cisco Pix Firewall

Gateway

Network uygulamalarında farklı şekillerde kullanılan ve kapalı bir alandan dışarıya örneğim internete çıkışma olanağı sunan cihazlardır. Bir başka kullanım şekli farklı protokoller kullanan ağların birbirlerine bağlanmasıdır.

Interne ya da başka bir networke bağlanmak için kullanıldığındaki bilgisayarların TCP/IP konfigürasyonunda Gateway olarak tasarlanan cihazın ip adresi tanımlanmalıdır. Bu şekilde kullanımı yönlendiriciler (Router gibi) vasıtasyyla yapılmalıdır.

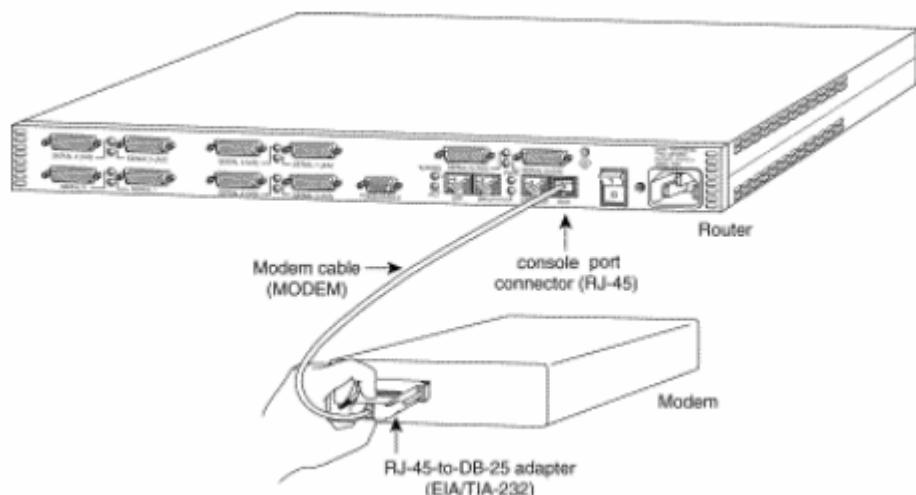


Cisco Gateway

Router

Router bir yönlendirme cihazıdır ve LAN-LAN yada LAN-WAN gibi bağlantılar da kullanılır. Router'ları basit bir yönlendirici olarak tanımlamak yetersiz olabilir. Çünkü Router'lar bir işletim sistemine sahiptirler (IOS - Internetworking Operating System) dolayısıyla programlanabilirler ve gerekli konfigürasyonlar yapıldığında bir uzak networke erişmek için mevcut birden fazla yol arasında kullanabilecekleri en iyi yolu seçimi yapabilirler (Best Path Determination).

Üzerinde LAN ve WAN bağlantıları için ayrı portla bulunur ve şaseli olarak ta üretilenlerdir. Gereksinime göre bu yuvalara LAN ya da WAN portları eklenebilir.



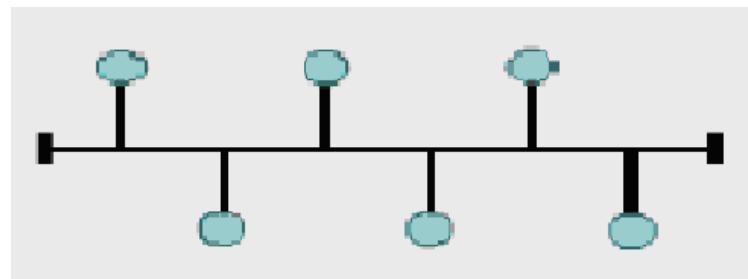
İlerleyen bölümlerde Router'ların konfigürasyonları detaylı olarak anlatılacaktır. Routerlar ve Routing işlemi CCNA sıvalarının ana omurgasıdır.

NETWORK TOPOLOJİLERİ

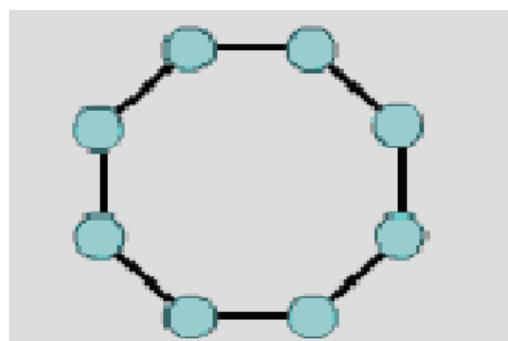
Topoloji dediğimiz de bir ağın fiziksel ya da mantıksal yapısını anlamalıyız. Networkü oluşturan cihazların fiziksel yerleri, kabloların bağlanış şekilleri, iletişimde kullanılan protokoller gibi birçok unsur network topolojilerini belirler.

Fiziksel Topolojiler:

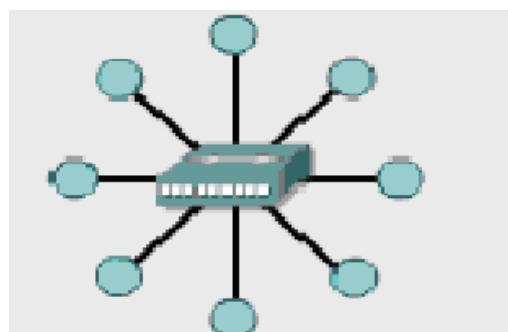
Bus Topoloji: Bütün terminallerin tek bir doğrusal bir kablo ile birbirlerine bağlanmışlardır. Burada hatta gönderilen sinyal bütün terminalere gider. Sinyal hedefe ulaşana ya da bir sonlandırıcıya gelene kadar hatta dolaşır. Çok az miktarda kablo kullanılıyor olması avantaj gibi görünse de ana kabloda meydana gelebilecek bir kopma bütün networkün çökmesine sebep olabilecektir. Ayrıca sorun giderme zorluğu ve hatta eklenen her yeni bilgisayarın networkün yükünü artırması da dezavantajlar arasında sayılabilir.



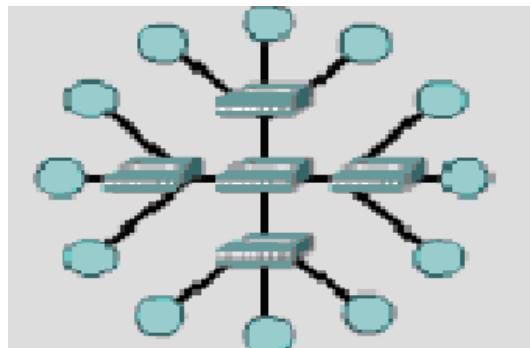
Ring Topoloji: Bu topolojide adından da anlaşılacağı gibi dairesel bir yapı söz konusudur. Hatta gönderilen sinyaller hedefe ulaşana kadar tüm terminalere uğrar. Tüm terminaller eşit haklara sahiptir.



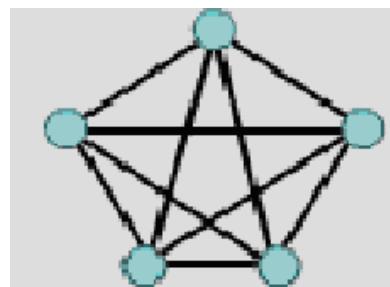
Star Topoloji: Star Topolojide her bilgisayar switch ya da hub dediğimiz network cihazlarına direk bağlıdır. Hatta gönderilen sinyal önce switch ya da hub'a gelir ve buradan hedefe gönderilir. Böyle bir yapının en büyük avantajı yeni bilgisayarlar ekleyerek büyümek çok kolaydır, yönetilmesi ve sorun giderilmesi kolaydır. Fakat diğer topolojilere göre çok daha fazla kablo kullanılmak zorunda kalınması ve switch ya da hub'ın devre dışı kalmasıyla tüm networkün çökecek olması gibi dezavantajları vardır.



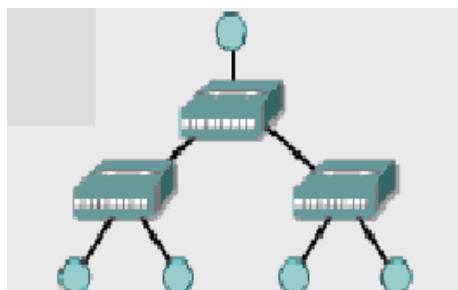
Extended Star Topoloji: Adından anlaşılacağı gibi Star Topolojinin geliştirilmesiyle ortaya çıkmıştır. Birden fazla yıldız topolojinin bir araya gelmesiyle oluşmuş bir yapıdır diyebiliriz.



Mesh Topoloji: Networkte bulunan bütün bilgisayarlarla diğer bütün bilgisayarlara direk bağlıdırlar. Uçtan uca bütün bilgisayarlar birbirine direkt bağlı olduğu için hedefe kısa zamanda ulaşılır, iki bilgisayar arasında ki bağlantının kopması durumda alternatif bir dürü olacaktır ama maliyetinin çok yüksek olması da unutulmamalıdır.



Hiyerarşik Topoloji: Üzerinde Bus topoloji ve Yıldız topolojiden özellikler taşır.



Mantıksal Topolojiler:

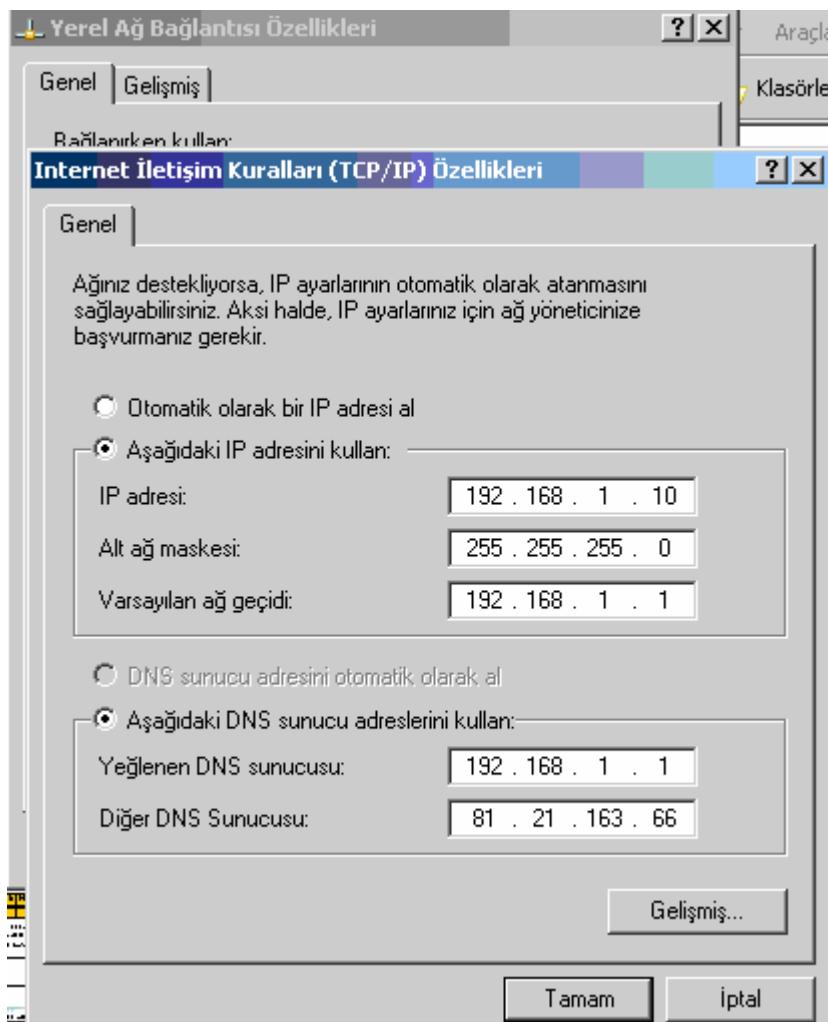
Broadcast Topoloji: Bir bilgisayar hatta gönderdiği bir sinyali diğer bütün bilgisayarların alacağı bir yayın şeklinde yapar. Yayın hedefe ulaştığı ana kadar bütün terminalleri tek tek dolaşır.

Token Passing Topoloji: Burada taşıyıcı görevinde olan bir token her bir terminale uğrayarak ağ ortamında dolaşır. Uğradığı terminal ekleyeceği bir data varsa onu token'a ekleyerek, ekleyecek bir data yoksa direkt bir sonraki terminale aktarır. Bu şekilde çalışlığında bir repeater görevi de üstlenmiştir.

IP Adresleme

Bilgisayarlar veya diğer cihazlar networklere fiziksel olarak bağlanmanın yanında mantıksal olarak da dahil olmalıdır. Bunun için aynı networkte iki cihazın aynı IP networkkunde olması gereklidir, yani Network adresleri aynı olmalıdır.

Bilgisayarlara IP adresleri Static veya Dinamik olarak verilebilir. Dinamik olarak IP adresi atanması için en güzel örnek DHCP Server' dir. Dynamic Host Configuration Protokolünün kısaltması olan DHCP konfigürasyonu Router üzerinde de yapılmaktadır. İlerleyen bölümlerde bir routerin nasıl DHCP server olarak konfigür edilebileceği anlatılmıştır.



(Statik olarak IP adresi atanması)

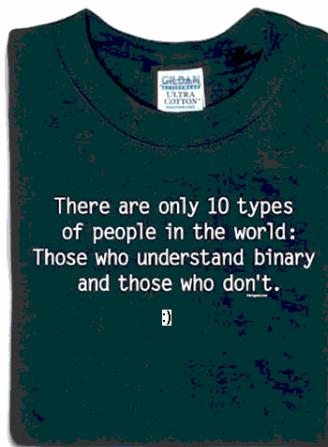
Burada Alt Ağ Maskesi ifadesi diktinizi çekmiştir. Başka bir deyişle genellikle türkçemizde de kullanılan Subnet Mask.

Subnet Mask bizim için önemlidir, çünkü daha ileride degineceğimiz bir networku alt networklere ayırmamız için Subnet Mask ile oynamamız gerekecektir. Çünkü Subnet Mask ile IP adresi binary durumda AND işlemine sokulduğunda network adresini verir.

IP adresler 4 oktetten ve her oktette 8 Bitten oluşur.

BIT – BInary digiT

10101001 11000111 01000101 10001001
 169 . 199 . 69 . 137



	1st octet	2nd octet	3rd octet	4th octet
172.0.0.0	Network	Host	Host	Host
Subnet Mask: 255.0.0.0 or /8	255	0	0	0
192.4.0.0	Network	Network	Host	Host
Subnet Mask: 255.255.0.0 or /16	255	255	0	0
192.168.1.0	Network	Network	Network	Host
Subnet Mask: 255.255.255.0 or /24	255	255	255	0
	1st octet	2nd octet	3rd octet	4th octet
172.0.0.0	Network	Host	Host	Host
Subnet Mask	11111111	00000000	00000000	00000000
192.4.0.0	Network	Network	Host	Host
Subnet Mask	11111111	11111111	00000000	00000000
192.168.1.0	Network	Network	Network	Host
Subnet Mask	11111111	11111111	11111111	00000000

İlk şekilde subnet masklar ikinci şekilde o subnet maskların Binary gösterimi mevcut. İlk resimde ki /8, /16, /24 gibi ifadeler goruyorsunuz. Bunlar Subnet Maskı ifade eder, daha doğrusu Subnet Maskin Binary gösterimi icineki yoplama 1 sayisidir.

Ornek Bazi Binary gosterimler:

192.168.1.0	11000000.10101000.00000001.00000000
255.255.255.0	11111111.11111111.11111111.00000000
192.168.1.255	11000000.10101000.00000001.11111111

192.168.0.0	11000000.10101000.00000000.00000000
255.255.0.0	11111111.11111111.00000000.00000000
192.168.255.255	11000000.10101000.11111111.11111111

192.168.0.0	11000000.10101000.00000000.00000000
255.255.255.0	11111111.11111111.11111111.00000000
192.168.0.255	11000000.10101000.00000000.11111111

Bir networkteki ilk ip adresi o networkun network adresini ve son ip adresi de Broadcast adresidir. Bu adresler network cihazlarina atanamaz.

<u>Network Address</u>	<u>Subnet Mask</u>	<u>Broadcast Address</u>
172.0.0.0	255.0.0.0	172.255.255.255
172.0.0.1 ve	172.255.255.254	
172.16.0.0	255.255.0.0	172.16.255.255
172.16.0.1 ve	172.16.255.254	
192.168.1.0	255.255.255.0	192.168.1.255
192.168.1.1 ve	192.168.1.254	
192.168.0.0	255.255.0.0	192.168.255.255
192.168.0.1 ve	192.168.0.254	
192.168.0.0	255.255.255.0	192.168.0.255
192.168.0.1 ve	192.168.0.254	

TCP / IP Modeli

TCP/IP ile olarak DARPA (Defense Advanced Research Projects Agency) ve Bekeley Software Distribution tarafından geliştirilen UNIX' de kullanılan bir protokoller gurubudur. Günümüzde internetin temel protokolü olarak yerini almış TCP/IP 'nin açılımı Transmission Control Protocol / Internet Protocol' dür.

TCP /IP modeli OSI katmanlarından çok daha önce standartlaştiği için OSI içinde referans olmuş 4 katmanlı bir yapıdır.

- Uygulama Katmanı
- Nakil Katmanı
- Internet Katmanı
- Ağa Giriş Katmanı

Uygulama Katmanı OSI modelindeki Uygulama, Oturum ve Sunum katmanlarına karşılık gelmekte ve o katmanların işlevlerini yerine getirmektedir. Bu katmanda TFTP, FTP, SMTP, SNMP gibi protokoller çalışmaktadır.

Nakil Katmanı OSI modelindeki Nakil katmanıyla bire bir eşleştirilebilir. Bu katmanda iki farklı sınıfa ayrılabilen iki protokol kullanılır. TCP ve UDP.

- Bağlantı Odaklı: TCP
- Bağlantısız: UDP

Internet Katmanı OSI modelindeki Network katmanına denktir ve adresleme, en iyi yol seçimi gibi işlevleri yerine getirir.

Bu katman da IP (Internet Protocol), ICMP (Internet Control Message Protocol), BOOTP (Bootstrap Protocol), DHCP (Dynamic Host Configuration Protocol), ARP (Adres Resolution Protocol) ve RARP (Reverse Address Resolution Protocol) gibi protokoller çalışmaktadır.

Ağa Giriş Katmanı ise OSI modelindeki Data-Link ve Fiziksel Katmana denk gelmektedir.

OSI REFERANS MODELİ

Kullanıcıların farklı talepleri ve dolayısıyla network üzerinde kullanılmak zorunda kalınan karmaşık uygulamalar, ağ kurulumlarında bir hiyerarşinin doğmasını kaçınılmaz yapmıştır. Bilgisayar ağları büyük ölçüde bu ağları yönetmek ve sorun gidermek, standart bir yapı olmadığı da düşünülürse çok daha zorlaşmaya başladı.

Uluslararası Standartlar Organizasyonu (ISO) bir çok ağ yapısını inceleyerek 1984 yılında OSI referans modelini geliştirdi. Artık donanım ve yazılım firmaları bu standarda uygun ürünler üretmeye başladılar.

OSI modelinde 7 katmanlı bir yapı kullanılmış ve bu model; karmaşıklığı azaltılmış, insanların belli katmanlarda uzmanlaşması için referans olmuş, katmanların işlevlerinin öğrenilmesi ve öğretilmesi kolaylaşmış, farklı donanım ve yazılım ürünlerinin birbirleriyle uyumlu çalışmasını sağlamış ve bir katmanda yapılan değişiklikler diğer katmanları etkilemediği için işbirliği, görev paylaşımı, problem çözümünü gibi konularda kolaylıklar getirmiştir.

Bahse konu OSI katmanlarını şu şekilde sıralayabiliriz.

7. Uygulama Katmanı (Application Layer)
6. Sunum Katmanı (Presentation Layer)
5. Oturum Katmanı (Session Layer)
4. Nakil Katmanı (Transport Layer)
3. Ağ Katmanı (Network Layer)
2. Data Link Katmanı (Data Link Layer)
1. Fiziksel Katman (Physical Layer)

Burada Uygulama, Oturum ve Sunum katmanları üst katmanlar olarak adlandırılırlar ve işlevlerini yazılımlar sağlamaktadır. (Bu katmanlar TCP/IP modelinde Uygulama Katmanı adı altında tek bir katman olarak yapıya dahil edilmiştir.) Nakil, Ağ, Data Link ve Fiziksel katmanlar ise alt katmanlar olarak adlandırılırlar ve işlevlerini bilgisayarların ve ağıda kullanılan diğer cihazların donanımları ve bu donanımlar üzerindeki yazılımlar sağlar.

Uygulama Katmanı (Application Layer)

Kullanıcıya en yakın olan katmandır ve diğer katmanlara herhangi bir servis sağlamaz. Burada kullanılan bazı uygulamalara şu örnekleri verebiliriz;

FTP
TFTP
Telnet
SMTP

SNMP
HTTP

Sunum Katmanı (Presentation Layer)

Gönderilecek datanın, datayı alacak bilgisayar tarafından da anlaşılabilen ortak bir formata dönüştürüldüğü katmandır. Bu katmanda data transferinin güvenli olması için şifreleme de mümkündür. Data formatlarına şu örnekler verilebilir;

MPEG
GIF
JPEG
ASCII

Oturum Katmanı (Session Layer)

İletişim kuran bilgisayarlar arasında oturum açar ve sonlandırır. Bu katmanda kullanılan servislere şu örnekler verilebilir;

SQL
Netbios Adları
NFS

Nakil Katmanı (Transport Layer)

Bu katman nakil edilecek datanın bozulmadan güvenli bir şekilde hedefe ulaştırılmasını sağlar. Üst katmanlardan gelen her türlü bilgi nakil katmanı tarafından diğer katmanlara ve hedefe ulaştırılır. Gönderilen datanın bozulmadan ve güvenli bir şekilde hedefe ulaşıp ulaşmadığını uygun protokollerle kontrol edebilir. Bu katmanda çalışan protokollere verilebilecek bazı örnekler şunlardır;

TCP
UDP

Bu katmanın en önemli iki fonksiyonun Güvenlilik ve Akış kontroldür.

Güvenlilik bilgisayarlar arasından gerçekleştirilen data transferinde datanın sağlıklı bir şekilde hedefe gönderip gönderilmemiş olduğunu yöneten, gönderilemediği durumlarda tekrar gönderilmesini sağlayan fonksiyondur.

İletişim halindeki bilgisayarlarda datayı gönderen bilgisayar alıcının kapasitesinden üzerinde datalar gönderebilirler. Böyle bir durumda datayı alan bilgisayar alamadığı paketleri yok edecektir ki önlemek için Nakil Katmanı Ara Bellekleme, tıkanıklıktan kaçınma ve Pencereleme metodlarını kullanarak akış kontrolünü sağlar.

Ara bellekleme de datanın akış hızına müdahale etmeden, kapasitenin üzerindeki datanın ara belleğe alınması, tıkanıklıktan kaçınma metodun da ICMP Source Quench mesajı ile gönderen bilgisayarın gönderimini yavaşlatması, Pencerelem metoduyla paketlerin gruplar halinde gönderilmesi sağlanır.

Ağ Katmanı (Network Layer)

Bu katman bir paketin yerel ağ içerisinde ya da diğer ağlar arasında ki hareketini sağlayan katmandır. Bu hareketin sağlanabilmesi için hiyerarşik bir adresleme yapısı gerekmektedir. Gelişen teknolojiyle birlikte mevcut ağlarında büyümeye eğilimde olması adresleme yapısının hiyerarşik olmasını gerektirmektedir. Ayrıca hiyerarşik sistem dataların hedef bilgisayara en etkili ve en kısa yoldan ulaşmasını da sağlar.

Bu katmanın bir özelliği olan Adresleme sayesinde bu sağlanabilmistiştir. Adresleme Dinamik ya da statik olarak yapılabilir. Sabit adresleme el ile yapılan adreslemedir. Dinamik adresleme de ise otomatik olarak ip dağıtandhcp gibi bir protokole ihtiyaç vardır.

Ayrıca bu katmanda harekete geçen bir datanın hedefine ulaşabilmesi için en iyi yol seçimide yapılır. Bu işleme Routing bu işlemi yerine getiren cihaza ise Router diyoruz. Router en basit tarif ile en iyi yol seçimini yapar ve broadcast geçirmemiş için ağ performansını olumsuz etkilemez. Bu katmanda kullanılan protokollere de şu örnekler verilebilir;

IP
ARP

RARP
BOOTP
ICMP

Data Link Katmanı (Data Link Layer)

Fiziksel adreslemenin ve network ortamında datanın nasıl taşınacağının tanımlandığı katmandır. Burada fiziksel adreslemeden kastettiğimiz şey MAC (Media Access Control) adresidir. Bu katman Hakemlik, Adresleme, Hata Saptama, Kapsüllenmiş Datayı Tanımlama fonksiyonlarına sahiptir.

Ethernet hakemlik için CSMA/CD (Carrier Sense Multiple Access with Collision Detect) adı verilen bir algoritmayı kullanır. Bu algoritma şu adımlardan oluşur;

1. Hatta boş olup olmadığını dinler
2. Boşsa data gönderir
3. Doluya bekler ve dinlemeye devam eder
4. Data transferinde çarışma olursa durur ve tekrar dinlemeye başlar.

Adresleme için, MAC adresi, Unicast adresi, broadcast adresi ve multicast adresi örnek olarak verilebilir.

Bu katman kullanılan protokollere şu örnekler verilebilir;

HDLC
PPP
ATM
Frame Relay

Fiziksel Katman (Physical Layer)

Bu katman datanın dijital rakamlara dönüştürerek aktarımın yapıldığı katmandır. Kablolar, hub, repeater cihazla bu katmanda yer alırlar. Bu katman da herhangi bir protokol tanımlanmamıştır.

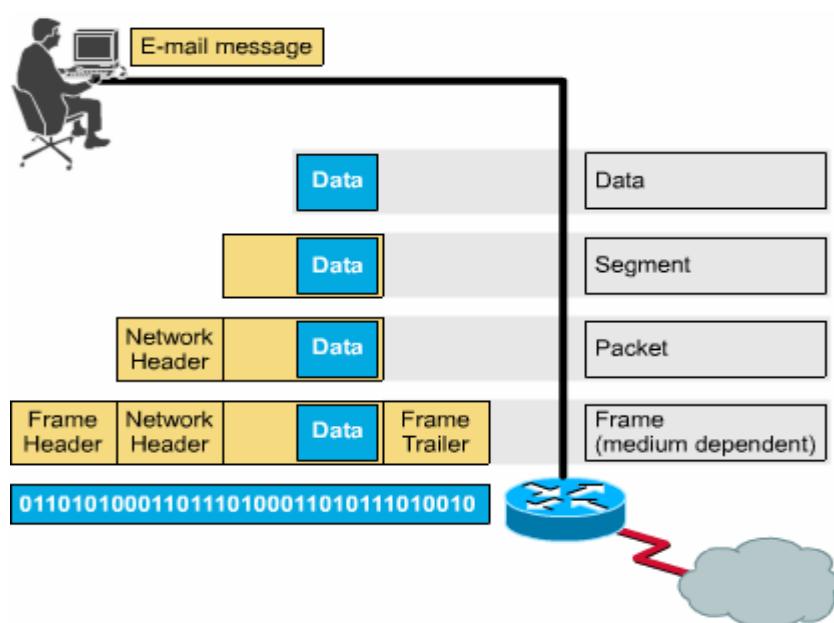
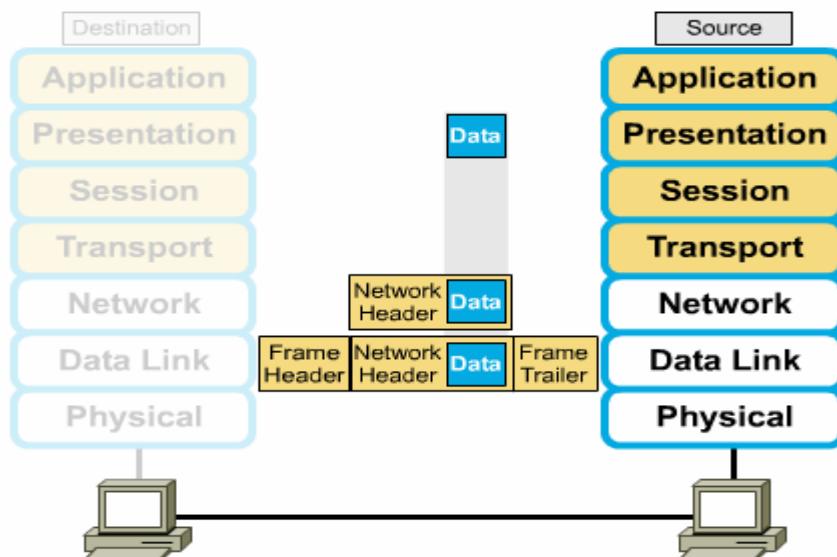
Data Encapsulation (Veri Paketleme)

Data Encapsulation 5 adımdan oluşur.

1. Uygulama, Sunum ve Oturum Katmanları kullanıcının girdiği veriyi 4. katman yani Nakil katmanına kadar getirir.
2. Nakil katmanı kendisine gelen bilgiyi segment adı verilen bölgelere ayırır ve datanın hangi protokolle gönderileceği (TCP - UDP) bilgisini de ekleyerek network katmanına gönderir.
3. Bu katmana gelen segment burada packetlere ayrılır ve IP header dene, hedef ve kaynak ipler gibi bilgileri bulunduğu başlığı ekleyerek bir alt katman olan data link katmanına gönderir.
4. Burada data artık framelere çevrilir ve mac adresleride eklenmiştir
5. Frame yapı bu katmanda bitlere ayrılır ve iletilir.

Aşağıdaki iki şkil konun daha iyi anlaşılmasını sağlayacaktır.

Data Encapsulation



TCP / IP Protokolleri

TCP (Transmission Control Protocol)

TCP, IP 'nin bir üst katmanında çalışan iki aktarım katmanı protokolünden birisidir.

TCP, güvenilir ve sanal devre üzerinden çalışan bir protokoldür. Aynı ağ üzerinde ya da farklı ağlar üzerinde iki hostun birbirleriyle güvenilir bir şekilde haberleşmesini sağlar.

TCP 'nin başlıca özellikleri şunlardır:

- Bir bağlantının (connection) kurulması ve sonlandırılması
- Güvenilir (Reliable) paket dağıtımının sağlanması
- Akış kontrolü (flow control) olanağı ile hostlarda veri taşmasının (overflow) önlenmesi
- Bozulmuş ya da ikilenmiş verinin düzeltilmesi (error recovery)
- Alıcı host içerisinde birçok uygulama arasında *demultiplexing* yapılması

TCP, internet ortamında şu işlevleri sağlar:

- Temel Veri Aktarımı (Basic Data Transfer)
- Güvenilirlik (Reliability)
- Uçtan uca Akış Kontrolü (End to end flow control)
- Çoğullama (Multiplexing)
- Bağlantılar (connections)
- Tam çift yönlü işlem (full duplex process)

TCP bağlantısının kurulması üç aşama (Three Way Handshake) sonucunda gerçekleşir:

1.Aşamada: Kaynak host bağlanmak istediği hosta bir istek paketi gönderir. Bu paketin TCP başlığında SYN = 1 ve ACK = 0 'dır. Gönderdiği paket içindeki segmente ait sıra numarası X 'tir.

2.Aşamada: Bu paketi alan hedefe TCP başlığında SYN = 1, ACK = 1 bitlerini kurarak kendi paketini sıra numarasına SEQ Numarası=Y ve onay numarası, ACK Numarası = (X + 1) 'i gönderir.

3.Aşamada: İsteğine karşılık bulan istemci son aşamada hedefe onay paketi gönderir ve bağlantı kurulmuş olur.

Sonra kaynak, hedefe göndermek istediği veri paketlerini gönderir.

TCP ve UDP üst protokollerle bağlantıda portları kullanırlar. 65535 adet port vardır ve IANA ([Internet Assigned Numbers Authority](#)) ilk 1024 portu Well-known portlar olarak ilan etmiştir. Bu portlardan bazıları şunlardır:

- FTP: 21
- Telnet: 23
- SMTP: 25
- DNS: 53

UDP (User Datagram Protocol)

UDP, TCP / IP protokol grubunun iki aktarım katmanı protokolünden birisidir. UDP, onay (acknowledge) gönderip alacak mekanizmalara sahip değildir. Bu yüzden veri iletiminde başarayı garantileyemez. Yani güvenilir bir aktarım servisi sağlamaz. Uygulamalar güvenli ve sıralı paket dağıtımını gerektiriyorsa UDP yerine TCP protokolü tercih edilmelidir. Bazı UDP port numaraları şunlardır;

- Who Is: 43
- DNS: 53
- NTP: 123
- SNMP: 161

FTP (File Transfer Protocol)

TCP tabanlı dosya transfer protokolüdür. FTP bağlantı kurulurken FTP sunucunun 21 numaralı portu kullanılır.

TFTP (Trivial File Transfer Protocol)

UDP tabanlı Cisco IOS tarafından desteklenen bir protokoldür. Router ve switchlerde dosya transferi için kullanılır, daha az hafıza ve işlemci gücü gerektirir. UDP tabanlı olduğu için hızlı bir iletişim söz konusudur fakat hata telfisi yoktur.

SMTP (Simple Mail Transfer Protocol)

Mail göndermek için sunucu ile istemci arasındaki iletişim şeklini belirleyen protokoldür. Sadece mail yollamak için kullanılan bu protokolde, basitçe, istemci bilgisayar SMTP sunucusuna bağlanarak gerekli kimlik bilgilerini gönderir, sunucunun onay vermesi halinde gerekli maili sunucuya iletir ve bağlantıyı sonlandırır.

SNMP (Simple Network Management Protocol)

SNMP protokolü ağlar üzerindeki birimleri denetlemek amacıyla geliştirilmiştir. Bir network cihazı üzerindeki sıcaklığından o cihaza bağlı kullanıcılar, internet bağlantı hızından sistem çalışma süresine kadar bir çok bilgi SNMP protokolünde tanımlanmış bir yapı içerisinde tutulur.

IP (Internet Protocol)

Bağlantısız bir protokoldür. Bu protokol datanın hedefe ulaşması için gidebileceği en iyi yolu seçer ve gelen paketleri IP başlıklarını okuyarak networkteki bilgisayarların yerlerini belirler. IP başlıklarında gönderilecek datanın yaşam süresi, datanın gönderilmesini sağlayacak protokol, kaynak ve hedef ip adresleri, kullanılan ip versiyonu gibi bilgiler bulunur.

ICMP (Internet Control Message Protocol)

Internet protokolünün control ve yönetimine yardımcı olan bir protokoldür. Bu protokol sayesinde network üzerindeki sorunla kolaylıkla tespit edilebilmektedir. RFC 792 standartı ile belirlenmiştir ve iki bilgisayar arasındaki iletişimde, hedef bilgisayarda, varsayılan ağ geçidinde veya routlerlarda oluşan hatalar ICMP mesajı olarak kaynak bilgisayara bildirilir.

Farklı durumlara göre farklı hata mesajları vardır. Bunlardan bazıları şunlardır:

Hedefe Ulaşılamıyor: Kaynak bilgisayara datanın gönderilmesiyle ilgili bir problem olduğu bilgisi döner.

Zaman Aşımı: Gönderilen datanın hedefe ulaşması için gereken zamanın dolduğunu ve bu sebeple paketin yok edildiğini belirten mesajdır.

Source Quench: Kaynak bilgisayara yönlendirmeyi yapan cihazdan daha hızlı data gönderdiğini ve yavaşlaması gerektiğini belirtir.

Tekrar Yönlendirme: Bu mesajı gönderen yönlendirici hedef için daha iyi bir yola sahip yönlendiricinin var olduğunu belirtir.

Yankı: Ping komutu tarafından bağlantıyı onaylamak için kullanılır.

Parameter Problem: Parametrenin yanlış olduğunu belirtmek için kullanılır.

Address Mask Request / Reply: Doğru Subnet Maskın öğrenilmesi için kullanılır.

Bolum Sonunda ICMP detaylı inceleneciktir.

RARP (Reverse Address Resolution Protocol)

Sabit diski olmayan aptal terminaller tarafından otomatik olarak ip adresi almak için kullanılan protokoldür. RARP istemci kendisiyle aynı segmentte bulunan RARP sunucudan ARP paket formatını kullanarak broadcast yapar ve ip adresi ister. RARP sunucu da uygun bir ip adresini istemciye gönderir.

BOOTP (Bootstrap Protocol)

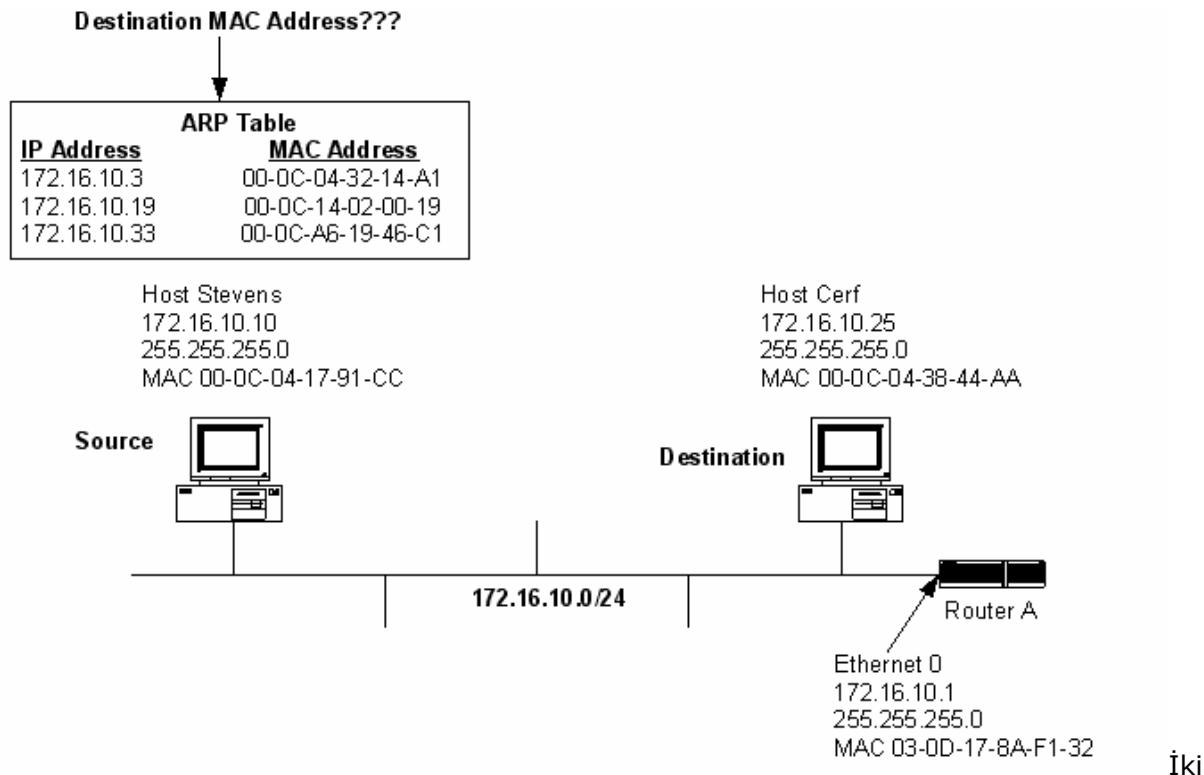
UDP tabanlıdır ve RARP protokolü gibi suncu - istemci ortamında çalışır. IP adresi isteyen bilgisayarlar bu isteklerini bir broadcast ile bildirirler. BOOTP sunucusu ise ip adresini, kendi ip adresini ve varsayılan ağ geçidi adresini bir broadcast ile networke gönderir. İstemciler MAC adreslerine balkarlar ve kendi MAC adreslerini gördüklerinde bu bilgileri alırlar.

DHCP (Dynamic Host Configuration Protocol)

BOOTP protokolünün gelişmiş hali olan bu protokol ile tam dinamik ip konfigürasyon dağıtıımı yapılabilir. Sunucu – istemci ortamında çalışanlar ve istemcilerde ip adreslerini otomatik olarak alacaklarına dair bir konfigürasyon yapılmalıdır. DHCP ile belirlene ip adresleri, subnet maskalar, dns server adresleri, varsayılan ağ geçidi gibi adresler dağıtılabılır, ip adresleri MAC adreslerine reserve edilebilir veya bazı ip adresleri tamamen kullanıma kapatılabilir. DHCP' den alınan ip adresleri DHCP server tarafından istemciye belirli sürelerle kiralanır ve istemci belirlili aralıklara ile DHCP serverdan kira süresini yenilemesini ister. Yenilenme kira süresine dolana kadar yapılamazsa DHCP server tarafından istemciye yeni bir ip adresi verilir.

ARP (Address Resolution Protocol)

ARP protokolü ip adresi bilinen bir bilgisayarın MAC adresini öğrenmede kullanılır.



İki bilgisayar iletişim kuracağı zaman kaynak bilgisayar hedef bilgisayara MAC adresini sorar ve bu işlem ARP Request denen be broadcast olan mesajla gerçekleşir. İlgili ip adresine sahip olan bilgisayar içinde MAC adresinin bulunduğu cevap paketini istemciye gönderir. Bu cevap mesajına ARP Reply denir. ARP protokolü Internet Katmanında çalışır. Kaynak bilgisayar ip adresi ve edindiği mac adresini eşleştirerek ön belleğinde saklar. "ARP -a" komut satırı komutu ile ön bellekte bulunan MAC adresleri görüntülenebilir.

IP HESAPLARI VE SUBNETTING

TCP/IP protokolünde tüm bilgisayarlar 32 bitlik "özgün" bir IP numarasına sahip olacak şekilde adreslenirler.

IP adresleri sınıflara ayrılmıştır, bu sınıflar şunlardır;

Class A :0.0.0.0 - 127.255.255.255 arasındaki ip adresleri.

Class B: 128.0.0.0 - 191.255.255.255 arasındaki ip adresleri.

Class C: 192.0.0.0 - 223. 255.255.255 arasındaki ip adresleri.

Class D: 224.0.0.0 - 239.255.255.255 arasındaki ip adresleri.

Class E: 240.0.0.0 – 255.255.255.255 arasındaki ip adresleri.

Her ip sınıfının subnet maskıda belirlenmiştir buna göre;

A sınıfı için subnet mask: 255.0.0.0,

B sınıfı için subnet mask: 255.255.0.0,

C, D, E sınıfları için subnet mask: 255.255.255.0 'dır.

NOT: Bir ip adresi yada protokol sınıfından bağımsız olarak bir subnet mask ile .alıyor veya çalışabiliyorsa “classless” aksi durumda “classfull” denir.

Bilgisayarımızdan komut sistemini açıp "ipconfig /all" komutunu verdığımızda kullandığımız bilgisayarın ip konfigürasyonunu görebiliriz.

Göründüğü gibi kullandığım bilgisayarın ip adresi C Class bir ip ve 192.168.2.155. Peki ip adreslerinin özel olması gerekiğine göre bütün dünyada bu ip adresinin aynısı kullanan bir başka bilgisayar yok mu ?

Gerçekten de böyle olsaydı mevcut ip adreslerimiz çöktan bitmiş olurdu. Belki de bunu önlemek için bazı ip aralıkları iç networkte kullanılmak üzere ayrılmıştır ve herhangi bir şekilde dış networkte (internette) kullanılamaz.

Bu ip aralıkları şunlardır:

10.0.0.0 – 10.255.255.255
 172.16.0.0 -172.31.255.255
 192.168.0.0 – 192.168.255.255

Internet ortamında bu ip adresleri kesinlikle kullanılmaz, iç network kullanıcıları internete çökarken, ISP tarafından sağlanan static veya dynamic bir ip adresine dönüşürler. İşte bu ip adresi tüm dünya da tek olacaktır.

Burada aklımıza şöyle bir soru gelebilir; Neden özel olarak ayrılmış ip sınıflardan kullanmalıyım, söz gelimi benim 212.212.212.212 gibi bir ip adresi kullanmama engel olan şey nedir?

Eğer firmanız internete hiçbir şekilde çıkmıyorsa istediğiniz ip adresini kullanabilirsiniz fakat çıkmıyorsa bu ip adresi belki de sizin o an ziyaret etmek istediğiniz bir sitenin ip adresi olabilir ve siz browser'ınız a sitenin adını yazdığınızda bir sonuç alamazsınız. Zira ip adresi sizinle aynı networkte.

Yerel networkler de ip adresi manuel olarak static konfigüre edilebileceği gibi örneğin DHCP gibi bir yazılımla dinamik olarak da dağıtılabılır.

Ip adreslerinin dağıtılması sırasında subnet maskların standart verilmesi ciddi sorunlara sebep olacaktır. Örneğin bir ISP firması söz gelimi 150 adet ip adresi almak istiyorsunuz. Bu durum standart subnet mask kullanılarak size verilebilecek minimum ip sayısı 255'dir. Daha vahim bir senaryo ise siz söz gelimi 500 tane ip adresi isteseniz ortaya çıkar çünkü o zaman size verilebilecek minimum ip sayısı $255 \times 255 = 65025$ 'dir.

Bunun önüne geçebilmek için yapılabilecek tek şey ise subnet masklar ile oynamaktan geçer. Bu sayede networkler sub-networklere bölünebilir ve ip israfın biraz olsun azalabilir.

Örnek:

Elinizde adresi 192.168.1.0 olan C Class bir network var ve bunu 4 subnete bölmek istiyorsunuz;

Bu durumda $256/4 = 64$ 'er tane ip adresiniz olacak.

Subnet Mask'ın son oktetini 256-64 yaparsanız bunu gerçekleştirmiş olursunuz. Bu durumda subnet mask=255.255.255.192 olacaktır ve elimizde subnet maskı 255.255.255.192 ve network adresleri sırasıyla;

192.168.1.0
 192.168.1.64
 192.168.1.128
 192.168.1.192

Olan 4 adet networkümüz, her networkte 64'er tane ip adresimiz olacak.

Bir networkün ilk ip adresi network adresini, son ip adresi broadcast adresini gösterdiği için kullanılamaz dolayısıyla bir networkte "useable" olarak adlandırılan, yani kullanılabilen ip sayısı toplam ip sayısından 2 eksikltir.

Useable Ip sayısı = toplam ip sayısı – 2

Network adresleri örneğin /24 şeklinde gösterilebilirler. /24 ip adresinin binary yazılımında soldan sağa 24 tane 1 olduğu anlamına gelir. Bu şekilde yazılımına CIDR denir. (Classless)

Örneğin;

255.255.255.0 binary olarak
11111111.11111111.11111111.00000000 'e eşittir ve 24 tane 1 den dolayı /24 olarak gösterilebilir.

Yukarıdaki örneğimizdeki subnet mask ise binary olarak;
11111111.11111111.11111111.11000000 'a eşit olacak dolayısıyla /26 olarak gösterilebilecektir.

Örnekler:

Subnet Mask	Binary Yazılım	CIDR İfade
255.255.128.0	11111111.11111111.10000000.00000000	/17
255.255.255.128	11111111.11111111.11111111.01000000	/25
255.255.255.252	11111111.11111111.11111111.11111100	/30

Elimizde bir ip adresi ve onun subnet maskı varsa ikisinin binary yazışmasını AND' leyerek network adresini bulabiliriz.

Örneğin elimizde subnet maskı 255.255.255.128 olan 192.168.1.141 gibi bir ip var.

$$\begin{array}{ll} 192.168.1.141 & = 11000000.10101000.00000001.10001101 \\ 255.255.255.128 & = 11111111.11111111.11111111.10000000 \\ \text{AND (çarpiyoruz)} & = 11000000.10101000.00000001.10000000 \\ \text{Network Adresi} & = 192. \quad 168. \quad 1. \quad 128 \end{array}$$

NOT: IP hesapları CCNA sınavına hazırlanan öğrenciler için son derece önemlidir. CCNA sınavlarında ip hesaplarıyla direkt ilgili yada içerisinde ip hesaplarını içeren bol sayıda soru çıkmaktadır.

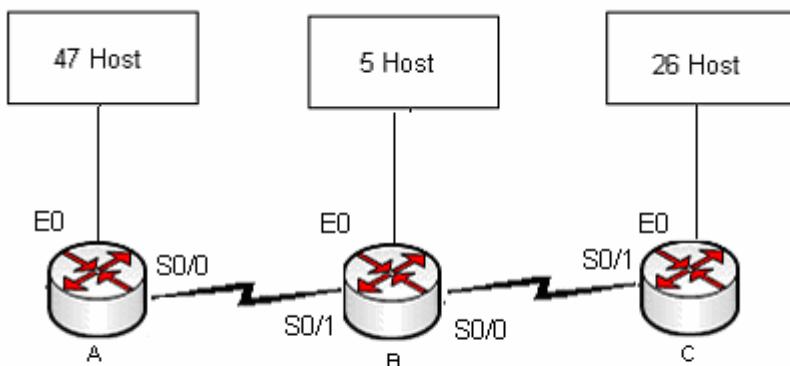
NOT: Router'in Ethernet interface' ine ip adresi atanırken önerilen networkün ilk ip adresini atamaktır.

Classfull - Classless IP Adresleri

Classfull adreslerde subnet masklar ip adresinin hangi sınıfa ait olduğunu direkt ilgilidir. Ip adreslerinin ilk oktetleri sınıflarını belirlerler ve her sınıf için subnet mask belirlenmiş durumdadır. Örnek vermek gerekirse 10.x.x.x gibi bir ip adresi A sınıfı bir ip adresidir ve Classfull olarak çalışan bir sistem de bu adresin subnet maskı her zaman 255.0.0.0 olacaktır. Routing protokoller anlatılırken detaylı değinilecek Rip ve Igrp protokoller Classfull protokollerdir ve subnet mask sınıflarına göre kendileri belirlerler.

Classless adreslerde ise subnet mask sınıftan bağımsızdır. Şöyleki 10.x.x.x gibi bir ip adresine istendiğinde 255.255.255.0 gibi bir subnet mask verilebilir ve Classless olan sistemlerde bunu algılarlar. Ospf, Eigrp gibi protokoller classless' dir. Classless adreslemeye VLSM (Variable Length Subnet Mask) veya CIDR (Classless Inter Domain Routing) denir.

IP Subnetting Örnek Çalışma



Elimizde 192.168.1.0 networkü var ve bu networkün 192.168.1.0 /25 lik kısmı daha sonra kullanılmak üzere ayrılmış durumda. Kalan ip adreslerini uygun şekilde dağıtmamız gerekiyor.

A, B, C Routerlarının Ethernet Interface' lerine bağlı 3 network ve router'ların birbirleriyle bağlantısında oluşan 2 (2' şer useable ip gereken) network olmak üzere elimiz toplam 5 network var.

Burada ilk yapmamız gereken host sayılarına bakara kaçar ip içeren networklere böleceğimize karar vermek.

A Routerı için 64,

C Routerı için 32,

B Routerı için 8 ve

Routerlar arasında ki networkler için 4'er ip içeren gruplar olmalı.

Dolayısıyla A routerı için 192.168.1.128 /26 networkü kullanımlı. Çünkü 192.168.1.0 ' dan 192.168.1.127 ' ye kadar olan ip ler daha sonra kullanılmak üzere ayrılmış durumda.

C routerı için 192.168.1.192 /27

B Routerı için 192.168.1.224/29

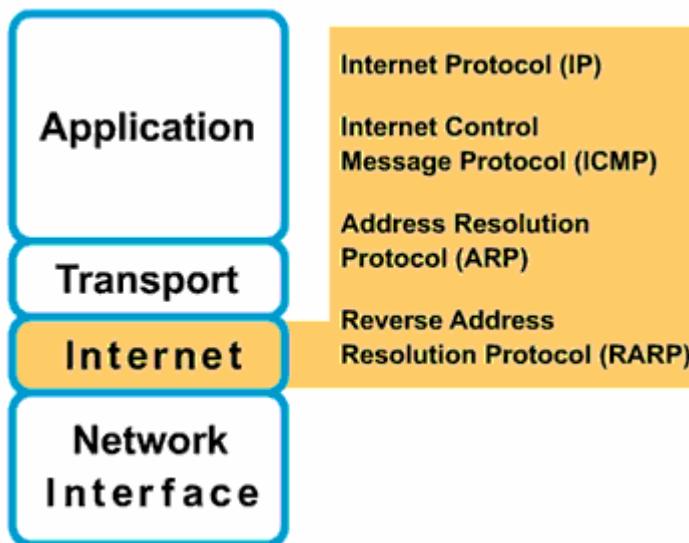
Diğer networkler içinde 192.168.1.232 / 30 ve 192.168.1.236 /30 networkleri kullanılmalıdır.

Network aralıklarımızı detaylı incelersek;

Son Oktet	Yeri	İçerdeği Ip Sayısı	Network Adresi	S.M.
0 - 127	Ayrılmış	128	192.168.1.0	/25
128-191	A Routerı	64	192.168.1.128	/26
192-223	C Routerı	32	192.168.1.192	/27
224-231	B Routerı	8	192.168.1.224	/29
232-235	A-B Arası	4	192.168.1.232	/30
236-239	B-C Arası	4	192.168.1.236	/30

CCNA sınavlarında Subnetting ile ilgili Sürükle-Bırak şeklinde ve bu çalışmaya benzer sorular çıkmaktadır. Bu sorularda yapılması gereken network aralıklarını bulup seçenekler arasından ki uygun ip adreslerini ilgili yerbere atamaktır.

ICMP (Internet Control Message Protocol)



Ucuncu katman yani Internet yada Network katmanı olarak adlandırdığımız katman IP bazından yönetildiğinin yapıldığı katmandır. IP data iletimi ve yönlendirme için belki de en iyi çözümüdür.

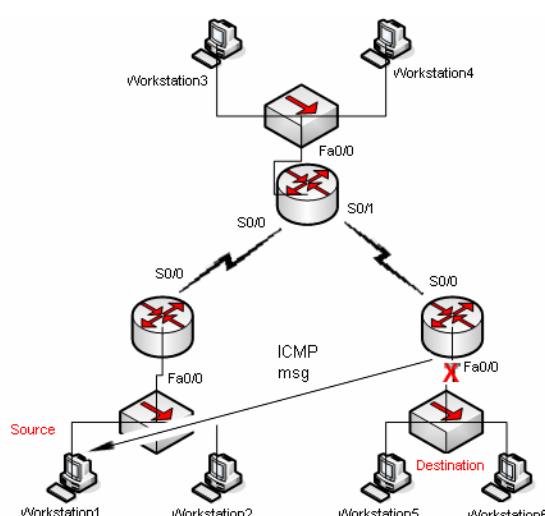
Fakat IP ile ilgili datanın iletimi sırasında herhangi bir sebeple fail olma durumu olduğunda bu durumu kontrol edecek hata mesajlarına sahip olmaması gibi bir sorun vardır. Sozgelimi yanlış konfigürasyonlar, donanımsal arızalar yada Routing Table'lar ile ilgili sorunlarda IP bir hata mesajı dondurmez.

ICMP IP'ının bu acısını gidermek üzere geliştirilmiş bir protokoldür. Bahsedilen durumlarda ICMP ilgili bir mesajı dondurur ve problem çözümlerinde Network Mühendislerine yardımcı olur.

Ancak burada IP bazında iletimin güvenilir olmadığını, ICMP mesajları ile bu güvenilirliğin sağlandığını söylemek yanlış olur. Datanın güvenilir şekilde iletilmesi bir üst katman olan Transport katmanı ve bu katmanda çalışan protokoller tarafından sağlanmalıdır.

Genel olarak ICMP mesajları iki ana başlık altında incelenebilir.

1. Hata Raporlama Mesajları
2. Durum Kontrol Mesajları



Sozgelimi Workstation1 den Workstation6 ya bir data gonderildigini ve bu C Routerinin da Fa0/0 interface' inin down oldugunu varsayalim. Bu durumda C routeri datanin ulastirilamadi ile ilgili bir mesaji geri dondurecektir. Burada bu bilgi sadece kaynaga yani Workstation1 e gonderilecektir.

ICMP mesajlari kendi frame yapisina sahip degildir. Bu mesajlar IP ile encapsule edilmiş frameler icerisine gomulmuslerdir. Dolayisiyla ICMP mesajlari tarafindan olusturulmus hata mesajlari kendi ICMP mesajlarini yaratmazlar.

ICMP mesajlari Type' lardan ve Code' lardan olusur.

Type	Name	Type	Name
0	Echo Reply	17	Address Mask Request
1	Unassigned	18	Address Mask Reply
2	Unassigned	19	Reserved (for Security)
3	Destination Unreachable	20-29	Reserved (for Robustness Experiment)
4	Source Quench	30	Traceroute
5	Redirect	31	Datagram Conversion Error
6	Alternate Host Address	32	Mobile Host Redirect
7	Unassigned	33	IPv6 Where-Are-You
8	Echo	34	IPv6 I-Am-Here
9	Router Advertisement	35	Mobile Registration Request
10	Router Solicitation	36	Mobile Registration Reply
11	Time Exceeded	37	Domain Name Request
12	Parameter Problem	38	Domain Name Reply
13	Timestamp	39	SKIP
14	Timestamp Reply	40	Photuris
15	Information Request	41-255	Reserved
16	Information Reply		

Type 3: Destination Unreachable

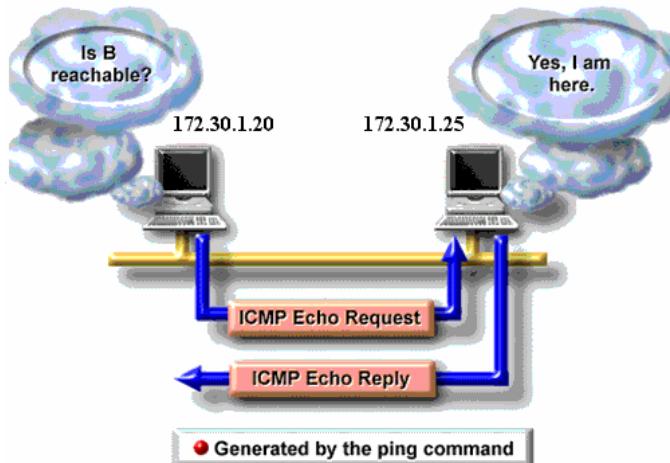
Codes

- 0 Net Unreachable
- 1 Host Unreachable
- 2 Protocol Unreachable
- 3 Port Unreachable
- 4 Fragmentation Needed and Don't Fragment was Set
- 5 Source Route Failed
- 6 Destination Network Unknown
- 7 Destination Host Unknown
- 8 Source Host Isolated
- 9 Communication with Destination Network is Administratively Prohibited
- 10 Communication with Destination Host is Administratively Prohibited
- 11 Destination Network Unreachable for Type of Service
- 12 Destination Host Unreachable for Type of Service
- 13 Communication Administratively Prohibited
- 14 Host Precedence Violation
- 15 Precedence cutoff in effect

(Type 3, Hedefe ulaslamiyor mesaj code'lari)

Ping ve Trace

Ping ve Trace komutları network mühendislerine bir çok problemin teshisinde yardımcı olar. Her iki komutta ICMP Echo Request ve ICMP Echo Reply mesajları ile çalışır.



Ping komutu ile ping istegini gönderen cihaz ICMP Echo Request' te bulunur. ICMP mesajlarındaki Echo Request Type'i 8 ve Code'u 0' dir.

Hedef ip adresi Echo Request mesajını gönderen cihaza Echo Reply ICMP mesajını gönderir. Bu mesajın Type'i 0 ve Code'u da 0' dir.

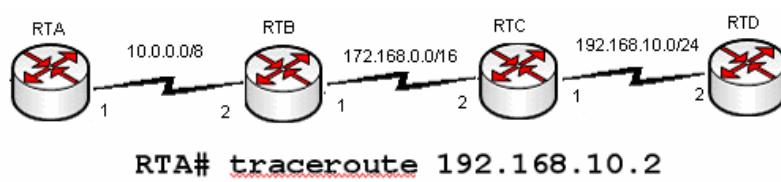
Trace komutu ise kaynak ve hedef ip adresleri arasında ki olası problemleri anlamaya yarar. Burada olası problemler dememizin sebebi kaynak ve hedef ip adresleri arasından birden fazla yol varsa her defasından farklı yollar izlenebilir.

Trace komutu bilgisayarlarda,
traceroute (*ip adresi*)

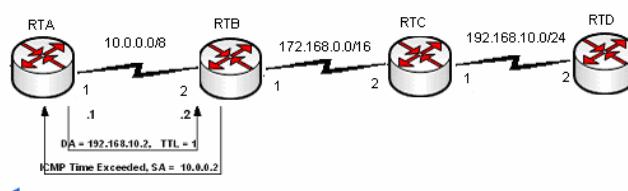
Routerlarda,
traceroute (*ip adresi*)

seklinde kullanılır. Traceroute çalışırken ping (ICMP Echo) mesajlarını kullanır.

Traceroute Örneği



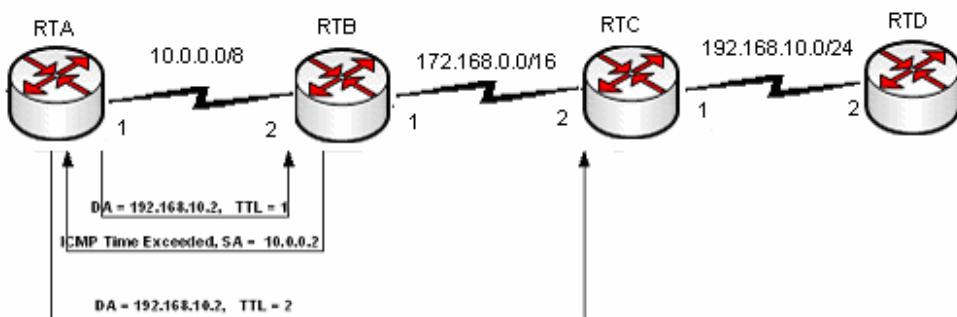
RTA# traceroute 192.168.10.2



Data Link Header (Layer 2)			IP Header (Layer 3)	ICMP Message - Time Exceeded					DataLink Tr.
Data Link Destination Address	Data Link Source Address	...	Source IP Add. 10.0.0.2 Dest. IP Add. 10.0.0.1 Protocol field 1	Type 11 Code 0	Chk sum	ID	Seq Nu m.	Data	FCS

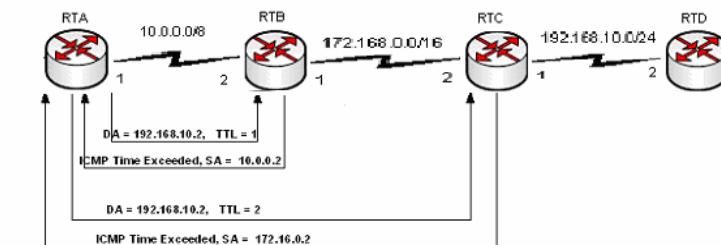
Traceroute basladiktan sonra IP basligindaki TTL degerini 1 yaparak ICMP Echo Requestte bulunur. RTB bu istegi aldigı zaman TTL degerine bakar ve bu deger 1 ise bir sonraki Routera gonderir, 0 ise İstek Zaman Asimi mesajini geri gonderir.

Bu durumda RTA İstek zaman asimi mesajini aldiktan sonra TTL degerini 1 artirarak yani 2 yaparak yeni bir Echo Requestte bulunur.



Data Link Header (Layer 2)		IP Header (Layer 3)	ICMP Message - Echo Request (trace)					UDP (Layer 4)	DataLink Tr.
Data Link Destination Address	Data Link Source Address	Source IP Add. 10.0.0.1 Dest. IP Add. 192.168.10.2 Protocol field 1 TTL 2	Type 8 Code 0	Chk sum	ID	Seq. Num	Data	DestPort 35000	FCS

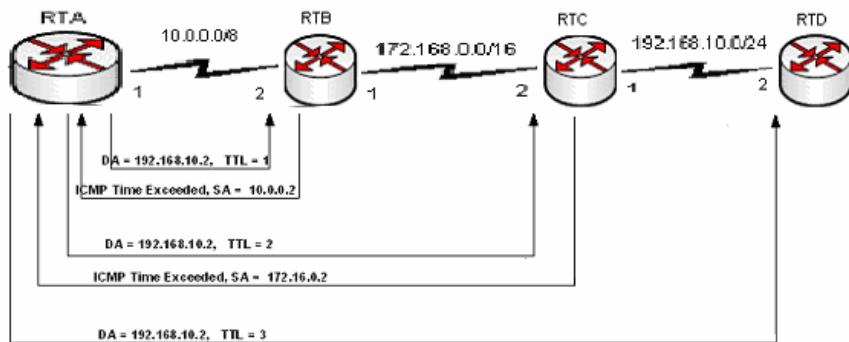
Artık RTB mesajı alip TTL degerine baktığında 0 değil 1 gorecek ve dolayısıyla bu istegi RTC routerina gönderecektir. RTB ile yasanınanlar bu sefer RTC ile de yasanacak ve TTL degeri 0 olarak gelen Echo Requeste RTC İstek zaman asimi mesajını geri gönderecek. Burada RTC nin dondureceği istek zaman asimi mesajında source ip adresi olarak RTC' nin adresi gorunecektir.



Data Link Header (Layer 2)		IP Header (Layer 3)	ICMP Message - Time Exceeded					DataLink Tr.
Data Link Destination Address	Data Link Source Address	Source IP Add. 172.168.0.2 Dest. IP Add. 10.0.0.1 Protocol field 1	Type 11 Code 0	Chk sum	ID	Seq. Nu. m.	Data	FCS

```
RTA# traceroute 192.168.10.2
Type escape sequence to abort.
Tracing the route to 192.168.10.2
```

```
1 10.0.0.2 4 msec 4 msec 4 msec
2 172.16.0.2 20 msec 16 msec 16 msec
```



Link Header (Layer 2)		IP Header (Layer 3)	ICMP Message - Echo Request (trace)					UDP (Layer 4)
a Link Layer Address	Data Link Layer Address	Source IP Add.	Type	Chk sum	ID	Seq. Num	Data	DestPort
RTA	10.0.0.1	10.0.0.1	8					35,000
	RTB	172.168.0.1	Code					
		1	0					
		TTL						
		3						

Bu sefer RTA TTL değerini 3 e çıkararak yeni bir Echo Requestte bulunacaktır. Dolayısıyla paket RTD routerına kadar gidebilecektir. Burada TTL değerini 0 olarak alan RTD hedef ip adresi kendine direkt bağlı olan networkte bulunduğu için artık zaman asımı mesajı göndermez, ICMP Port Unreachable Mesajını geri döndürür. (Type=3, Code=3)

RTA routeri port unreachable mesajını trace ettiği network olarak algılar.

```
RTA# traceroute 192.168.10.2
Type escape sequence to abort.
Tracing the route to 192.168.10.2
```

```
1 10.0.0.2 4 msec 4 msec 4 msec
2 172.16.0.2 20 msec 16 msec 16 msec
3 192.168.10.2 16 msec 16 msec 16 msec
```

ROUTER

Network katmanında bulunan ve temel işlevi farklı networklere erişimde en iyi yol seçimini (Best Path Determination) yapan cihaza Router denir.

Router Bileşenleri

RAM: Random Access Memory' nin kısaltmasıdır. Router'ın running-configuration adı verilen ve çalıştığı andaki konfigürasyonunu içeren bilgileri bulundurur. Bazı kaynaklarda RAM' a Dinamik RAM anlamında DRAM, running-configuration dosyasına da active-configuration denir. Router kapatıldığında ya da yeniden başlatıldığında RAM' de bulunan bilgiler silinir.

ROM: Read Only Memory' nin kısaltmasıdır. Yani sadece okunabilir kesinlikle silinemez ve değiştirilemez. ROM' un ayrı başlıklarda incelenmesi gereken bileşenleri vardır. Bunları şöyle sıralayabiliriz;

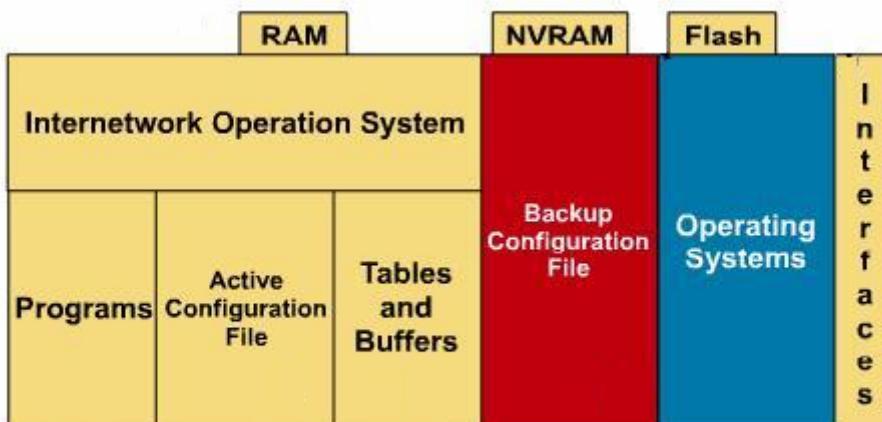
Post; Router' in power tuşuna basıldığı anda devreye girer ve donanım testini gerçekleştirir.

MiniOS; Konsoldan giriş yapılarak ulaşılabilen IOS' de bir sorun ile karşılaşlığımızda sorun çözmemize yetecek kadar içeriğe sahip bölümdür. Burada TFTP servera erişilerek çeşitli yüklemeler yapılabilir.

Bootstrap; Router' in çalışmasını sağlayan bir yazılımdır. Microsoft işletim sistemlerindeki "boot.ini" dosyasına benzetilebilir.

ROM Monitör; Router' in BIOS' u gibi düşünülebilir. Düşük seviyede hata ayıklama ve özellikle ilerde detaylı anlatacağımız şifre kırma işlemlerinde kullanılır. Kısaca Rommon olarak adlandırılır.

FLASH: Silinebilir, değiştirilebilir, yeniden yüklenebilir (EEPROM) bir hafıza kartıdır. IOS burada bulunur. Flash üzerine yüklemeler yapmak için TFTP Server adındaki programdan faydalанılır.



NVRAM: Non-Volatile Ram' in kısaltmasıdır. Yani kalıcı, silinmez bir RAM' dir. Startup-Configuration denen başlangıç konfigürasyon dosyaları burada bulunur. Router açıldığından buradaki dosyayı alıp RAM' de çalışmasını sağlar. NVRAM boş ise konfigürasyon için bir sihirbaz kullanmayı isteyip istemeyeceğimizi soracaktır.

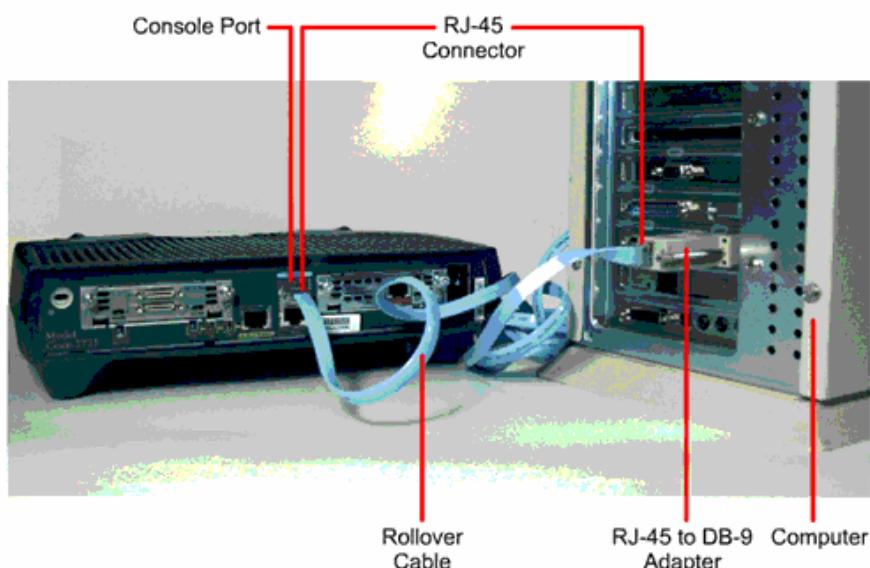
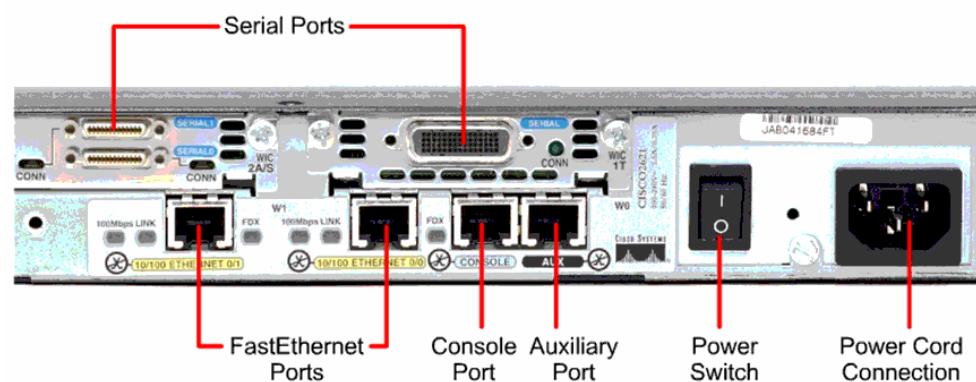
--- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]: _

[Giriş](#) [Kullanıcılar](#) [Dönüşüm Aletleri](#) [Yardımcılar](#) [İzleme](#) [İçerik](#) [Makale](#) [Makalelerin varlığı](#)

CPU: İşlemci.

INTERFACE: Router'a erişmek ya da çeşitli fiziksel bağlantıları yapmak için kullanılan fiziksel arabirimlerdir. CCNA eğitimleri boyunca kullanılacak interfaceleri "Serial Interface" ve "Ethernet Interface" ler olarak sınıflandırabiliriz. Bu interfaceler default olarak kapalı durumdadır.



IOS (Internetworking Operating System)

Adından da anlaşılacağı gibi IOS, Router ve Switch'lerin yönetilmesinde kullanılan işletim sistemidir. IOS bize CLI (Command Line Interface) denen text görünümünde bir arayüz sağlar.

Bu arayüze erişmenin üç temel yolu vardır. Consol Port, Auxilary Port ya da Telnet vasıtısıyla erişmek mümkündür.

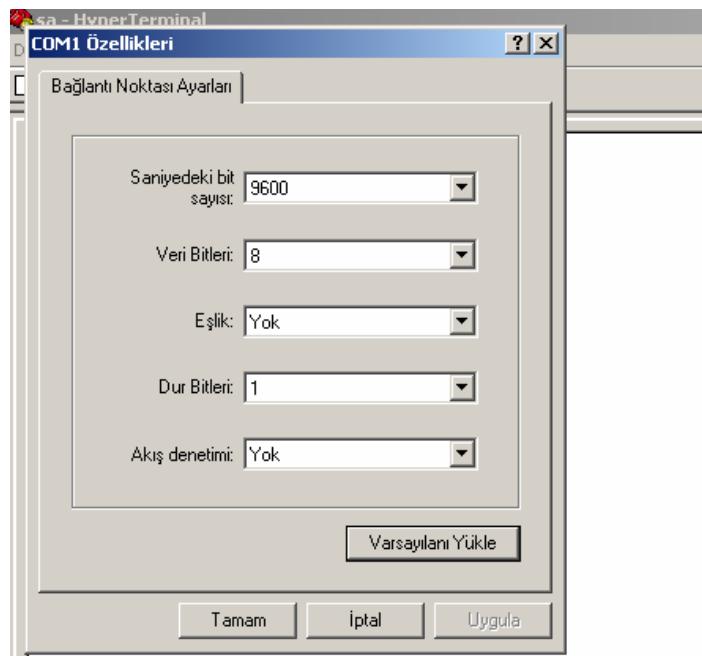
Consol port ile erişmek için, Roll Over denen, her iki ucu RJ45 ile sonlandırılmış ve bilgisayarımızın com portundan girilmesi için bir dönüştürücüye sahip özel kablolar kullanılır. Bunlara Konsol kablosu denir. Hyper Terminal yardımıyla CLI'ye erişilebilir.

Auxilary Port veya Yardımcı portta denilen bu port modem aracılığı ile asenkron çevirmeli bağlantı kullanarak erişmemizi sağlar.



Buraya herhangi bir isim verip geçiyoruz.

Burada COM1'in seçili olduğuna dikkat edin.



Burada "Varsayılanı Yükle" dediken sonra Tamam'a basıyoruz ve routemizə erişimimiz tamamlanıyor.

Telnet ile Router'ımıza erişebilmemiz için öncelikle Telnet oturumunun aktif hale getirilmesi gereklidir. Bunun için Telnet ve enable şifreleri verilmelidir. Bu şifrelerin nasıl verileceğini daha detaylı inceleyeceğiz.

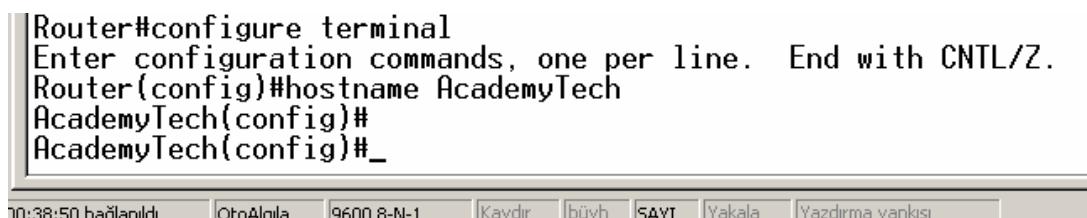
Router Çalışma Modları

User Mod: Router'ı açıp arayüze eristiğimiz anda karşımıza çıkan moddur. Burada yönetimsel işlemler yapılamaz. Bir sonraki modlara geçiş için kullanılır.

Privileged Mod: User modda iken "enable" yazıp entera bastığımızda bu moda geçeriz. Bu moda enable moda denir ve önerilen davranış bu moda geçerken şifre konulmasıdır. Zira bir kullanıcı bu moda geçtikten sonra Router'a tamamen hakim olur.



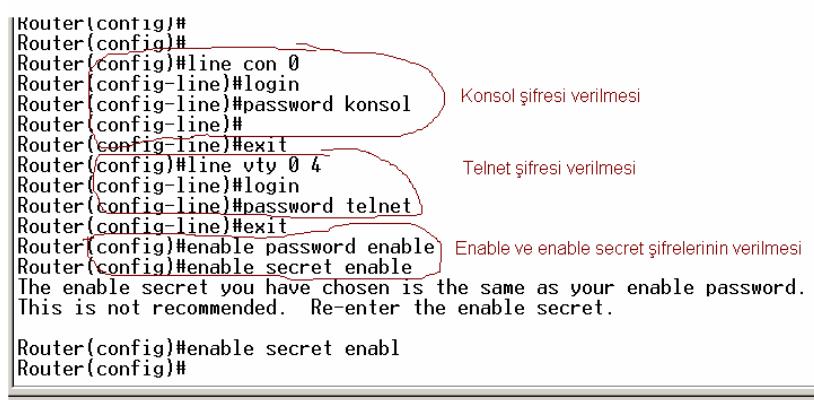
Global Configuration Mod: Config Mod diye de anılan bu moda geçmek için enable moda iken "configure terminal" yazılır ve entera basılır. Bu modda yapılan değişiklikler bütün Router'ı etkiler. Örneğin bu modda iken bir router'a isim verilebilir. Bu mod ileride detaylı anlatacağımız alt modlara ayrılr.



Enable, Telnet ve Konsol Şifreleri verme

Enable şifresi Global Configuration modda verilirken konsol ve telnet şifreleri line Configuration mod denilebilecek alt modlarda verilebilir. Enable şifre "enable secret" komutu kullanılarak 5. leveldan şifrelenebilirken telnet ve konsol şifrelerinde bu mümkün değildir. Fakat 7. leveldan şifrelenebilirler ve bunun için gerekli komutumuz "service-password encryption" dır.

Bir Router' a "enable secret" ve "enable" şifreleri, aynı olmamak şartıyla birlikte verilebilir. Bu durumda "enable secret" şifresi geçerli olacaktır.



(Dikkat edilirse enable ve enable secret şifrelerinin aynı olmasına izin verilmiyor)

```
|Router(config)#service password-encryption
Router(config)#

```

(Şifrelerin 7. leveldan encrypted edilmesi)

Şifre verirken kullandığımız "login" komutu dikkatinizi çekmiştir. Default olarak şifresiz kabul edilen Router'a bu komut ile artık kendisine şifre vasıtıyla erişileceği bilgisini vermiş oluyoruz. Bütün komutlar başına "no" yazılarak geçersiz hale getirilebilir.

" no enable secret" gibi bir komut ile enable secret şifresini kaldırabiliriz.

```
|Router(config)#
Router(config)#no enable password
Router(config)#no enable secret
Router(config)#line con 0
Router(config-line)#no pass
Router(config-line)#no password
Router(config-line)#
Router(config-line)#

```

Yardım Alma

Router konfigürasyonu sırasında kullanacağınız komutun ilk harflerini yapış tab tuşuna bastığınızda, yazdığınız komut bulunduğu mod için geçerliyse ve o harflerle başlayan başka bir komut yoksa, Router sizin için komutu tamamlayacaktır.

```
|Router#conf
Router#configure
Router#sh
Router#show

```

Ve yine devamını hatırlamadığınız komutlar için sonuna "?" koymak suretiyle yardım alabilirsiniz.

```
|Router#co?
configure connect copy
```

```
Router#sh?
show
```

```
Router#sh
```

Konuyu tam olarak kavrama da AcademyTech laboratuarlarında sıkça uyguladığımız bir çalışma da (Clock uygulaması) aşağıda detaylı bir şekilde gösterilmiştir.

```
|Router# clear clock
Router#cl?                                Router bize cl ile başlayan clear ve clock komutları olduğunu söyledi.
clear clock
Router#clock ?                               clock komutunu seçip ? yaptığımızda ise set komutunu
set Set the time and date                  kullanabileceğimizi gördük.
Router#clock set ?
hh:mm:ss Current Time
Router#clock set 17:23:51 ?
<1-31> Day of the month
MONTH Month of the year
Router#clock set 17:23:51 24 feb ?
<1993-2035> Year
Router#clock set 17:23:51 24 feb 2006 ?
<cr>
Router#clock set 17:23:51 24 feb 2006
Router#
Router#_
```

Router# clear clock
Router#cl? Router bize cl ile başlayan clear ve clock komutları olduğunu söyledi.
clear clock
Router#clock ? clock komutunu seçip ? yaptığımızda ise set komutunu
set Set the time and date kullanabileceğimizi gördük.
Router#clock set ?
hh:mm:ss Current Time
Router#clock set 17:23:51 ?
<1-31> Day of the month
MONTH Month of the year
Router#clock set 17:23:51 24 feb ?
<1993-2035> Year
Router#clock set 17:23:51 24 feb 2006 ?
<cr>
Router#clock set 17:23:51 24 feb 2006
Router#
Router#_
Router#_

Show Komutları

Show komutu Router ile ilgili bir çok şeyi görüntüleme de bize yardımcı olur. Show komutları Enable Moda çalışır ve yardım alındığında görünecektir ki bir çok uygulaması vardır.

AcademyTech#show ?	
access-expression	List access expression
access-lists	List access lists
accounting	Accounting data for active sessions
adjacency	Adjacent nodes
aliases	Display alias commands
alps	Alps information
arp	ARP table
async	Information on terminal lines used as
backup	Backup status
bridge	Bridge Forwarding/Filtering Database
bsc	BSC interface information
bstun	BSTUN interface information
buffers	Buffer pool statistics
c2600	Show c2600 information
call	Show Calls
cdp	CDP information
cef	Cisco Express Forwarding
clock	Display the system clock
cls	DLC user information
compress	Show compression statistics
configuration	Contents of Non-Volatile memory
context	Show context information
--More--	

:08:14 bağlandı OtoAlgıla 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı

Görünende çok daha uzun bir listeyi Routerlarda inceleyebilirsiniz. Burada önemli ve bizlere CCNA eğitimi boyunca yardımcı olacak belli başlı show komutları, yeri geldikçe gösterilecektir.

Konfigürasyon Dosyaları

Routerın açılış konfigürasyonunun tutulduğu startup-config ve çalışan konfigürasyonunun tutulduğu running-config adı altında iki dosyası vardır. Bir router'ın running-config ve startup-config dosyalarını "show" komutu ile görebilir, "copy" komutu ile birbirleri üzerine kopyalayabilir, "erase" komutu ile silebiliriz.

Startup-Config: NVRAM'da bulunur, yeni alınmış bir Router için üzerinde hiçbir bilgi bulunmaz. Ve böyle bir Router açılışta startup ve running konfigürasyonunun bir sihirbaz yardımıyla yapıp yapmayacağımız sorusunu sorar. Bu sihirbaz gereksiz ve boş zaman harcatan bir çok soru ile doludur ki önerdiğimiz ve uyguladığımız konfigürasyonu manuel yapmaktadır.

Router#show startup-config

-

Running-Config: RAM'da bulunur ve Router'ın çalıştığı andaki konfigürasyonunu tutar. Router kapatıldığında buradaki bilgiler gider.

```
Router#show running-config
Building configuration...
```

Bir Router yeniden başlatıldığı zaman startup-config dosyası dolu ise, IOS tarafından bu dosya alınıp RAM'a aktarılır ve dolayısıyla o artık Running-config olmuştur.

Bir router' in running-config ve startup-config dosyalarını "show" komutu ile görebilir, "copy" komutu ile birbirleri üzerine kopyalayabiliriz.

```
Router#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router#
```

```
Router#erase nvram:
Erasing the nvram filesystem will remove all files! Continue? [confirm]
[OK]
Erase of nvram: complete
Router#
Router#
```

Write Komutu

Kopyalama ve silme işlemlerinde "Write" komutu da kullanılabilir. Write komutu ile birlikte kullanılabilecek komutlar aşağıdadır.

```
AcademyTech#write ?
  erase      Erase NV memory
  memory    Write to NV memory
  network   Write to network TFTP server
  terminal  Write to terminal
<cr>
```

```
AcademyTech#write _
```

NOT: Kısaca "wr" yazmak Running Konfigürasyonumuzu NVRAM'a kayıt edecektir.

```
AcademyTech#wr
Building configuration...
[OK]
AcademyTech#_
```

Şifre Kırma

Router'ın şifrelerini unuttugunuza ya da ikinci el bir Router aldığınızı ve bu router'ın konfigürasyon dosyalarının hala üzerinde olduğunu dolayısıyla şifrelerini bilmediğinizi varsayıyalım. Böyle bir durumda şifreyi değiştirmek ve istersek eski konfigürasyonun bozulmamasını da sağlayarak bunu yapmak mümkündür. Bu ilk bakışta bir güvenlik açığı gibi görünse de, bu işlemin yapılabilmesi için konsoldan router'a bağlanmamız, dolayısıyla fiziksel olarak router'ın yanında olmamız gerekeceği için açık denilemez. Zira fiziksel olarak erişilebilen bir router'ın şifreleriyle oynayabilmenin bir sakıncası yoktur.

Adım adım şifre kırma işlemini inceleyecek olursak;

1. Router açılırken Ctrl+Break tuşlarına basılarak Rom Monitöre girilir. Burada "Router>" yerine "rommon>" ifadesiyle karşılaşacağız.

```
monitor: command "boot" aborted due to user interrupt
rommon 1 >
rommon 1 >
rommon 1 >
rommon 1 >
```

2. "confreg" komutu ile başlangıç register'i değiştirilir ve NVRAM yerine direk RAM'dan çalışmaya başlaması sağlanır. Bu sayede mevcut konfigürasyon NVRAM'da bulunmaya devam ederken Router RAM'dan sıfır konfigürasyon ile açılacaktır. 0x2102 olan register 0x2142 olarak değiştirilmelidir.

```
rommon 1 >
rommon 1 > confreg 0x2142
```

```
You must reset or power cycle for new config to take effect
rommon 2 > _
```

3. Router yeniden başlatılır. Açıldığında Router'ın herhangi bir şifre sormadığını göreceksiniz.
4. Enable moda geçilir. Bu moda geçtikten sonra artık istediğimiz her şeyi yapabileceğimize göre, eski konfigürasyonu kaybetmek istemiyorsak, "copy startup-config running-config" komutu ile o dosyayı alır ve şifreleri değiştirip yeniden NVRAM'a kaydederiz.

```
Router#copy startup-config running-config
Destination filename [running-config]?
499 bytes copied in 0.889 secs
Router#
```

Bundan sonra istediğimiz değişiklikleri yapıp running-config dosyasını tekrar Startup-config üzerine yeni haliyle kopyalayabiliriz.

5. Son olarak Rom Monitör'e girip değiştirdiğimiz register'i eski haline getirip (0x2102) getirip Router'ımıza yeniden başlatabilir ve eski konfigürasyon ve yeni şifreyle router'ın açıldığını görebiliriz.

```

Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#conf
Router(config)#config-register 0x2102
Router(config)#

```

Temel Router Konfigürasyonu

Bir router'ın çalışması için şifre vermekten çok daha fazlası gereklidir. En temel gereklilik ise Router'ın interface'lerine ip adresi atamaktır. Router'ın interfaceleri default olarak shutdown durumdadır ve bunun kaldırılması gereklidir ki bu da ip adresinin atadıktan sonra ilgili interface'ye "no shutdown" komutu vermek ile mümkündür.

Bir router'ın interfacelerinden herhangi birine ip adresi atanmanın diğerinde farklı yoktur. Yapılacak işlemler sırasıyla interface konfigürasyon moduna geçmek, ip adresini subnet maski ile birlikte yazmak ve "no shutdown" ile interface'i aktif hale getirmektir.

Örnek bir çalışma olarak Router'ımıza şu ip adreslerini atayalım.

Ethernet Interface Ip adresi : 192.168.1.1 / 24
 Serial (0/0) Interface Ip Adresi: 192.168.2.1 /24
 Serial (0/1)Interface Ip Adresi : 192.168.3.1 /24

```

Router(config)#interface et
Router(config)#interface ethernet 0/0
Router(config-if)#ip addr
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
(Ethernet 0/0 interface'ine ip adresi verildi)

```

```

Router(config)#interface serial 0/0
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
(Serial 0/0 interface'ine ip adresi verildi)

```

```

Router(config)#interface serial 0/1
Router(config-if)#ip address 192.168.3.1 255.255.255.0
Router(config-if)#no shut
Router(config-if)#no shutdown
(Serial 0/1 interface'ine ip adresi verildi)

```

Buradaki 0/0, 0/1 gibi ifadeler standart olmamakla birlikte Router'ımızın üzerinde yazıyor olmalı. Eğer yazmıyorsa, Router'ımıza "show running-config" komutunu verip hangi interface'in hangi numaraya sahip olduğunu öğrenebiliriz.

Router'ımıza gerekli şifreleri verip interfacelerine de gerekli ipleri atadıktan sonra "Show running-config" ile göreceğimiz text ifade şu şekilde olacaktır.

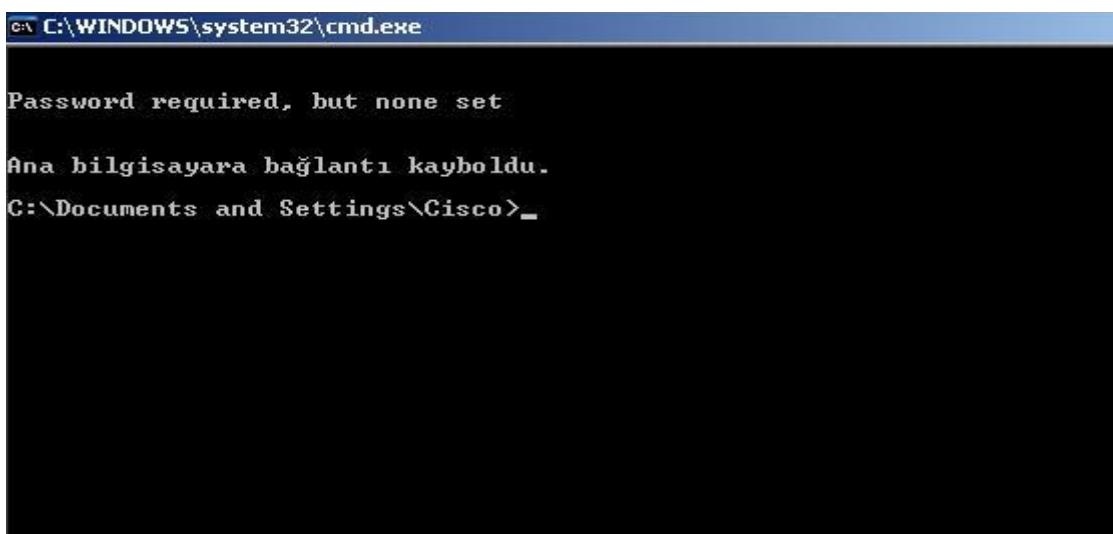
```

Router#sh running-config
Building configuration...
Current configuration : 526 bytes
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
hostname Router
memory-size iomem 10
ip subnet-zero
interface Ethernet0/0
  ip address 192.168.1.1 255.255.255.0
  half-duplex
interface Serial0/0
  ip address 192.168.2.1 255.255.255.0
  no fair-queue
interface Serial0/1
  ip address 192.168.3.1 255.255.255.0
!
ip classless
ip http server
dial-peer cor custom
!
gatekeeper
shutdown
line con 0
line aux 0
line vty 0 4
!
end
-----
```

Router'a Telnet İle Bağlanma

Router üzerinde bir konfigürasyon yapılacak olmas mutlaka Router'a fiziksel olarak erişmeyi yani Konsol' dan bağlanmayı gerektirmez. Router'a Telnet ile de bağlanılabilir.

Fakat bunun için bazı şartların yerine gelmesi gereklidir. Öncelikle Router'ın ethernet interface' i up olmalıdır ve Telnet, Enable şifreleri verilmiş olmalıdır. Telnet şifresi verilmediğinde "Password Required, but none set" şeklinde bir hata mesajı alınacak ve bağlan gerçekleştirilemeden kaybolacaktır.



Şekilde ki gibi bir tabloyla karşılaşıldığında anlaşılması gereken gerekli şifrelerin verilmemiş olduğunu söyleyebiliriz. Önceki bölümlerde öğrendiğimiz gibi şifreleri verdikten sonra bağlantımızı gerçekleştirebiliriz.

```
Router(config)#
Router(config)#enable pass
Router(config)#enable password academytech
Router(config)#line vty 0
Router(config-line)#pass
Router(config-line)#password cisco
Router(config-line)#login
Router(config-line)#exit
Router(config)#

```

(Telnet ve Enable Şifrelerinin Verilmesi)

```
Telnet 192.168.1.175

User Access Verification

Password:
Password:
Router>enable
Password:
Password:
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router<config>#hostname AcademyTech
AcademyTech<config>#exit
AcademyTech#copy run
AcademyTech#copy running-config start
AcademyTech#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
AcademyTech#
```

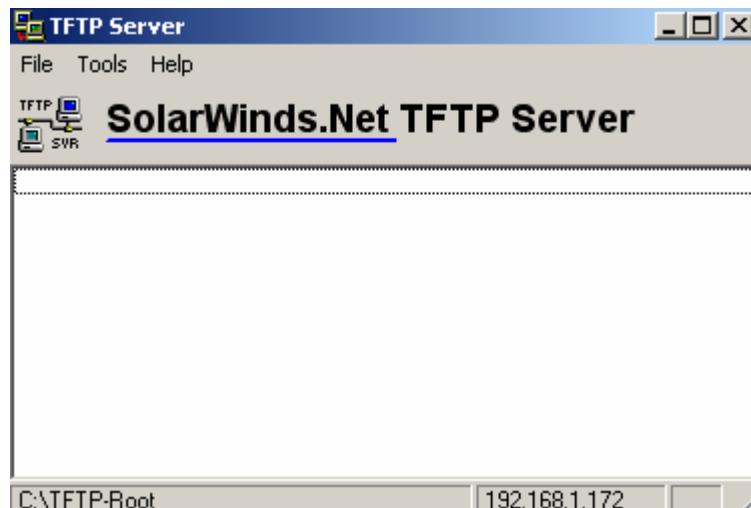
Göründüğü gibi şifreler verildikten sonra bağlantı gerçekleştirilebilir ve her türlü konfigürasyon yapılabılır. Komutlarda herhangi bir farklılık söz konusu değildir.

TFTP Server'a Yedek Alma

Konfigürasyonu yapılmış bir Router'ın startup ve running-config dosyalarının yedeklerini almak akıllıca bir harekettir. Bu TFTP Server sayesinde mümkün. Ve yine TFTP sayesinde Flash'ın yedeği alınabilir, güncellemesi yapılabilir.

TFTP Server normal bir PC'ye yükleyeceğimiz UDP protokolünü kullanan ufak bir programdır. Bu program network üzerinden TFTP isteklerini karşılamak için devamlı networkü dinler.

TFTP Server'a yedek alınabilmesi için kurulu olduğu bilgisayarın ip adresini, flaş'ın yedeği alınacaksa onun tam adını bilmek gerekir. Flash'ın tam adını "Show version" komutu ile öğrenebiliriz. "copy" komutu bundan sonrası kendisi halledecektir.



Copy startup-config tftp:

Copy running-config tftp:

Copy flash tftp

gibi bir komut yazdığımızda bize ilk olarak TFTP Server'ın ip adresi ve şayet Flaş'ın yedeğinin alacaksa onun tam adını soracaktır. Ve bütün bunlar yapılırken TFTP Server çalışıyor durumda olmalı.

TFTP Serverdan geri yüklemelerde ise komut tam tersi yazılarak çalıştırılacaktır.

Copy tftp startup-config

Copy tftp running-config

Copy tftp flash

```
Router#
Router#
Router#
Router#
Router#
Router#
Router#ping 192.168.1.175
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.1.175

!!!!!

Success rate is 100 percent (5/5), round-trip

Router#

Router#copy start

Router#copy startup-config tftp

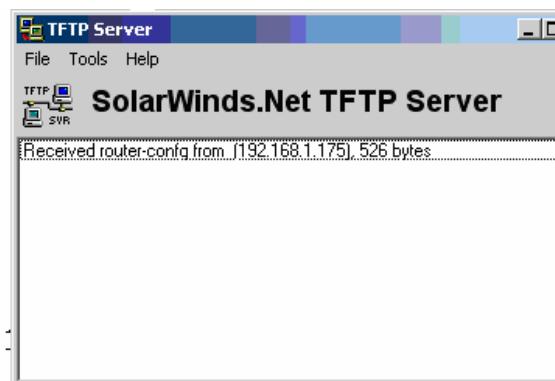
Address or name of remote host []? 192.168.1.172

Destination filename [router-config]?

.!!

526 bytes copied in 4.348 secs (121 bytes/sec)

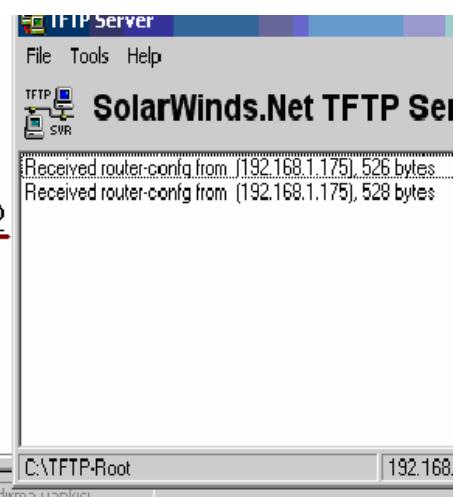
Router#



```

Router#
Router#
Router#
Router#copy
Router#copy run
Router#copy running-config tftp
Address or name of remote host []? 192.168.1.172
Destination filename [router-config]?
!!
528 bytes copied in 1.583 secs (334 bytes/sec)
Router#
Router#
Router#

```



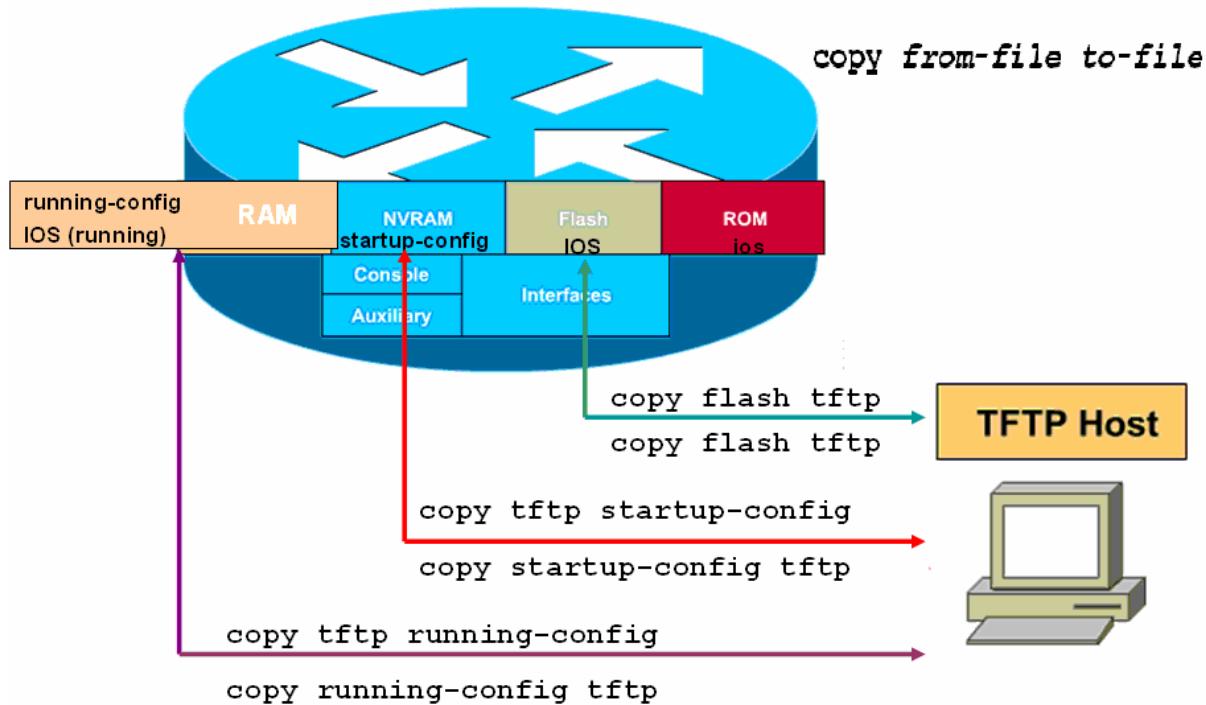
```

Router#copy tftp running-config
Address or name of remote host []? 192.168.1.172
Source filename []? router-config
Destination filename [running-config]?
Accessing tftp://192.168.1.172/router-config...
Loading router-config from 192.168.1.172 (via Ethernet0/0): !
[OK - 528 bytes]

528 bytes copied in 0.934 secs (565 bytes/sec)
Router#

```

Copy Komutları Özeti



IOS Yedek Alma ve Yükleme

TFTP Server kullanarak IOS' in yedeği alınabilir veya IOS yüklenebilir. Bunun için Sistem Image File' in tam dosya adı bilinmelidir ve bu "show version" komutu ile öğrenilebilir. Alınan bütün yedekler gibi IOS' in yedeği de TFTP Server tarafından TFTP-Root klasörünün altına atılır.

```

ROM: System Bootstrap, Version 11.3(2)XA3, PLATFORM SPECIFIC RELEASE SOFTWARE (f
c1)
ROM: C2600 Software (C2600-IX-M), Version 12.2(28), RELEASE SOFTWARE (fc5)

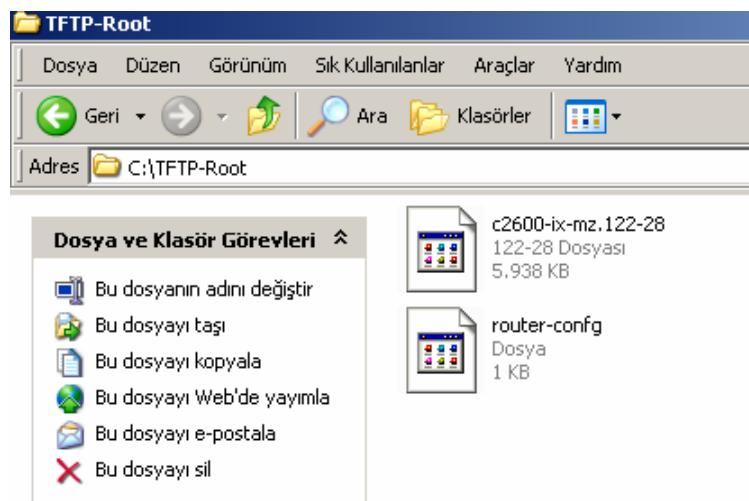
Router uptime is 34 minutes
System returned to ROM by power-on
System image file is "flash:c2600-ix-mz.122-28"

cisco 2610 (MPC860) processor (revision 0x202) with 36864K/4096K bytes of memory
Processor board ID JAB024903E2 (2074409390)
M860 processor: part number 0, mask 49
Bridging software.
X.25 software, Version 3.0.0.
1 Ethernet/IEEE 802.3 interface(s)
2 Serial network interface(s)
--More--

```

Yedek alırken **startup-config** ve **running-config**' den farklı olarak dikkat edilecek tek konu hedef dosya adıdır ve şekilde belirtildiği gibi tam adı olmalıdır.

00:37:21 bağlandı OtoAlqila 9600 8-N-1 Kaydır büy SAYI Yakala Yazdırma yankı ("copy flash tftp" komutuyla yedek alınması)



(Yedekleri alınan dosyalar TFTP-Root klasörünün altında)

IOS olmadığında router ethernet interface' ine ip adresi veremeyeceğimizde IOS' in konsoldan yüklenmesi gerekmektedir. Bunun için kullanılacak iletişim kuralı "xmodem" dir ve konsol hızı 115200 bps' a çıkarılmalıdır. Tabii ki bu işlemlerin tamamı Rom Monitör (Rommon) kullanılarak yapılabilir.

Bunun için Router açılırken Ctrl+Break tuşlarını basılılara Rommon' a girilir ve konsol hızı 115200 bps' a .İkartılır. (Flash tamamen boş ise, IOS yoksa CTRL+Break tuşlarına basmaya da gerek yoktur. Zira IOS olmadığı zaman Router direk Rommon' dan açılır.) Bu durumda ilk bağlantımız 9600 bps ile yapıldığı için kopacaktır. Hyper Terminal' de bağlantı hızı 115200' e çıkarılarak yeniden bağlanılır.

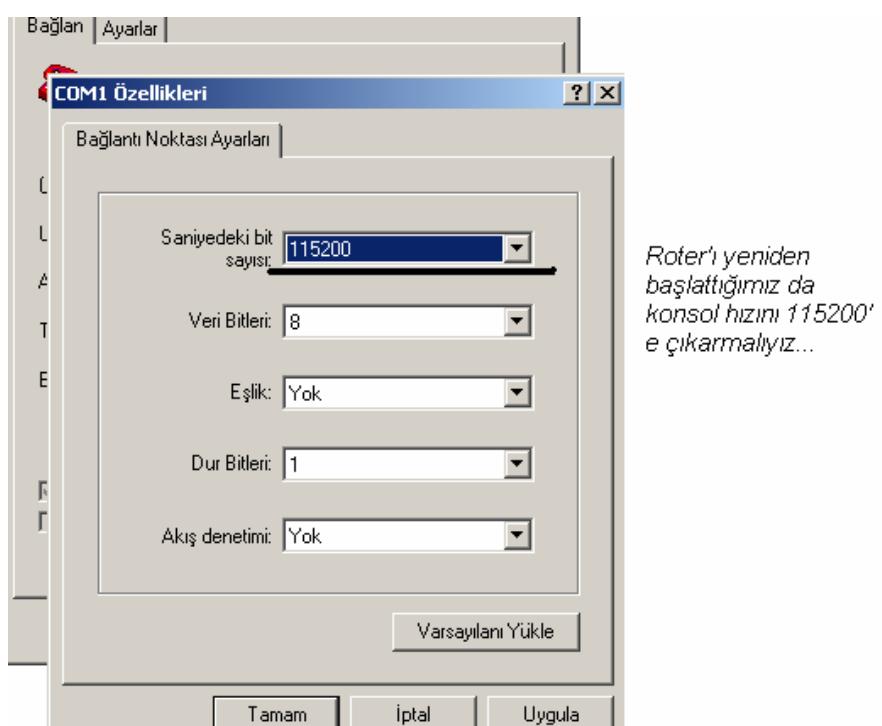
Rommon açılıp komut satırında "confreg" yazıldığında router bize değiştirmek istediğimiz böülümleri sıralayacak ve burada sadece konsol hızı için evet deyip uygun hızı seçeceğiz. Ve router'ı yeniden başlatmamız istenecek.

rommon 1 > confreg

```
do you wish to change the configuration? y/n [n]: y
enable "diagnostic mode"? y/n [n]: n
enable "use net in IP bcast address"? y/n [n]: n
enable "load rom after netboot fails"? y/n [n]: n
enable "use all zero broadcast"? y/n [n]: n
disable "break/abort has effect"
enable "ignore system config info"? y/n [n]: n
change console baud rate? y/n [n]: 7
change console baud rate? y/n [n]: y
enter rate: 0 = 9600, 1 = 4800, 2 = 1200, 3 = 2400
4 = 19200, 5 = 38400, 6 = 57600, 7 = 115200 [0]: 7
change the boot characteristics? y/n [n]: n
```

You must reset or power cycle for new config to take effect

Artık routerımız 115200 konsol hızıyla açılacak ve Xmodem iletişim kuralı kullanılarak Flash'ın yüklemesi yapılabilecektir. Bunun için Hyper Terminal'in "Dosya Gönder" özelliğinden faydalananacağız.



```
rommon 3 > xmodem -c c2600-ix-mz.122-28.bin → Komut Satırı
Do not start the sending program yet...
    File size      Checksum   File name
    6080092 bytes (0x5cc65c)  0xd773   c2600-ix-mz.122-28

WARNING: All existing data in bootflash will be lost!
Invoke this application only for disaster recovery.
Do you wish to continue? y/n [n]: y
Ready to receive file c2600-ix-mz.122-28.bin ...
```

rommon 2 > xmodem ?
Do not start the sending program yet...
 File size Checksum File name
 6080092 bytes (0x5cc65c) 0xd773 c2600-ix-mz.122-28

Dosya adı ve yeri seçildi

WARNING: All existing data in bootflash will be lost!
Invoke this application only for disaster recovery.
Do you wish to continue? y/n [n]: y
Ready to receive file c2600-ix-mz.122-28.bin ...

Xmodem İletişim Kuralı Seçildi

monitor: command "C:\TFTP-Root\c2600-ix-mz.122-28" sa için Xmodem dosya gönderilmesi
rommon 2 > xmodem ?
Do not start the sending program yet...
 File size Checksum File name
 6080092 bytes (0x5cc65c) 0xd773 c2600-ix-mz.122-28

Gönderilen: C:\TFTP-Root\c2600-ix-mz.122-28
Paket: 3720 Hata denetimi: CRC
Deneme: 0 Toplam deneme: 0
Son hata:

Dosya: 465k / 5938K
Geçen: 00:01:56 Kalan: 00:22:48 Üretilen iş: 4097 cps

iptal *cps/bps*

WARNING: All existing data in bootflash will be lost!
Invoke this application only for disaster recovery.
Do you wish to continue? y/n [n]: y
Ready to receive file c2600-ix-mz.122-28.bin ...

Şu anda yüklemeye yapılmıyor

IOS yüklenmesi tamamlandıktan sonra Routerımız açabiliriz. Fakat yapmamız gerekenler henüz bitmiş değil. Routerımızın konsol hızı hala 115200' de. İlk açtıığımızda Hyper Terminal ile bu hızı göz önüne alarak bağlanıp, standartı sağlama için konsol hızını tekrar 9600 bps olarak değiştirmemiz gereklidir.

Bunun için Konsol-line konfigürasyonuna girip "speed" komutuyla gerekli düzenlemeyi yapmalıyız.

Ve bağlantımız kesildi çünkü Hyper Termina ile bağlantımızı oluştururken konsol hızı olarak 115200 bps'ı seçmiştik. Bunu da eski haline getirmemiz gereklidir.

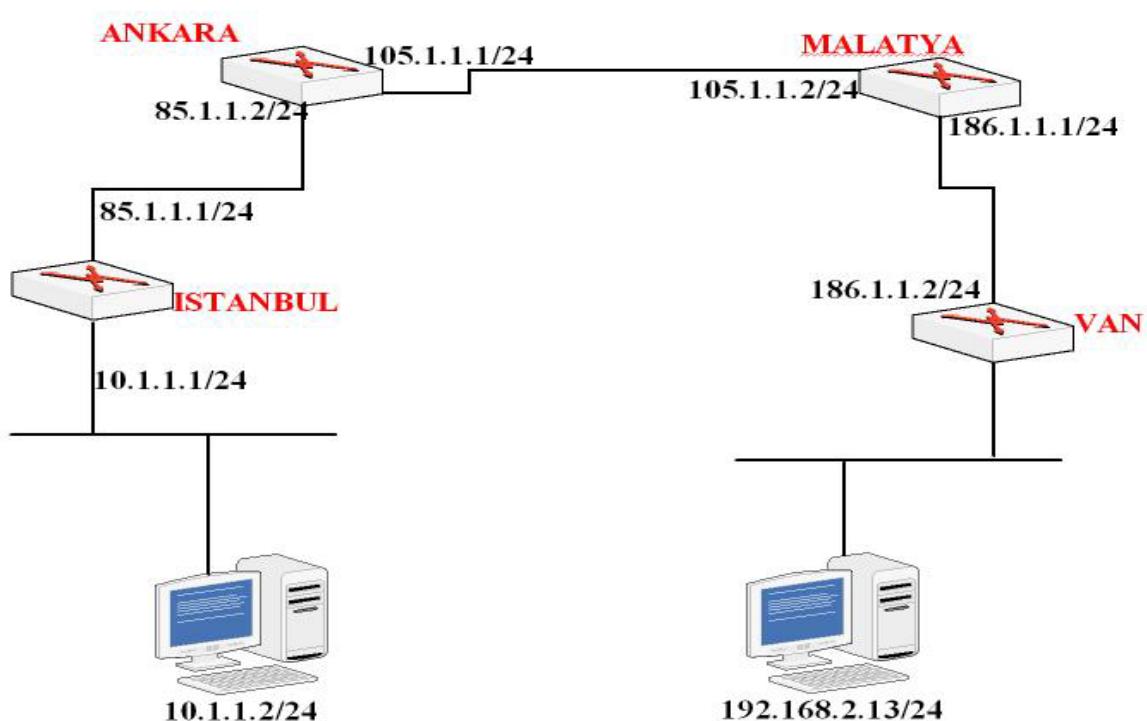
```
Router(config)#  
Router(config)#line con 0  
Router(config-line)#speed ?  
<0-4294967295> Transmit and receive speeds  
Router(config-line)#speed 9600
```

00:53:20 saatlerde 00:00:00 115200 S-N-1 Kavşak İstihb SAVT Makala Yazdırma seçenekleri

ROUTING GİRİŞ

Routing en basit ifadeyle bir uzak networke gitmek için gereken yol bilgisinin Router' lar tarafından sağlanmasıdır. Routerlar kendilerine gelen paketlerde, hedef ip adresi olarak, nerede olduğunu ve nasıl gidileceğini bildikleri bir networkten adres bulunduğuunda, hedefe yönlendirme yaparlar. Aksi takdirde paketi yok ederler.

Aşağıdaki senaryoyu biraz incelersek daha iyi fikir sahibi olabiliriz.



Burada routerlar üzerinde hiçbir yönlendirme konfigürasyonu yapmadığımızda, 10.1.1.2 bilgisayarından 192.168.2.13 bilgisayarına ping atarsak başarısız oluruz. Peki neden ?

Çünkü İstanbul Router' i 192.168.2.13 bilgisayarının bulunduğu network hakkında hiçbir bilgiye sahip değil. Router' lar üzerlerinde konfigürasyon yapılmadığından sadece kendileri (interface' lerine) direk bağlı olan network' leri bilirler. Bu durumda İstanbul Router' iin sadece 10.1.1.0 ve 85.1.1.0 network' lerini bildiğini söyleyebiliriz.

Eğer Router'ın gideceği ip numarası directly connected değil ise Router'a gideceği ip adresine nereden ulaşacağını belirtmemiz gereklidir.

- Routing işlem, Bir paketin bir Networkdeki bir aygıtın diğer Networkdeki bir aygıta gönderilmesidir.
- Routerlar destination adrese sahiptirler.
- Routerlar; bütün uzak Networklerin olası yollarını (routes) bilirler.
- Routerlar ; Uzak Networklerin en iyi(en kısa) yolunu kendileri seçerler. Bunu seçerken o anki duruma bakarlar ve belli bir kriter yoktur. O anki hattın yoğunluğuna bakabilir , aradaki mesafeye bakabilir.... En iyi yolu kendisi seçmekteidir.
- Routerler uzak Networklerin adreslerini oluşturdukları bir "Routing" tablosunda tutarlar. Bu bilgiler manuel olarak yada otomatik olarak tutulur. Manuel olarak tutulmasına Static Routing , Otomatik olarak tutulmasına Dynamic Routing denir.

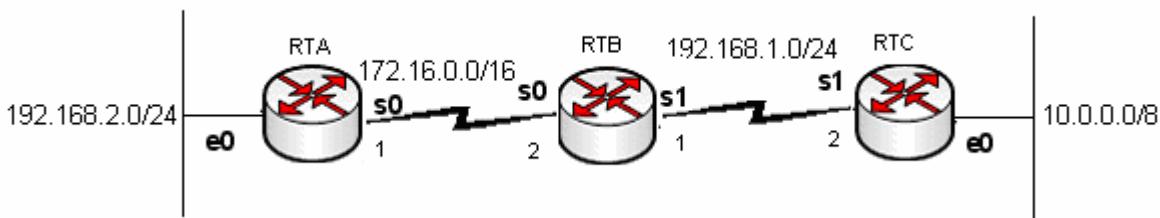
Bu senaryo da İstanbul Router'ının bilmediği networkler uygun tanımlamalar yapılarak Router'a öğretilebilir.

Peki İstanbul Router'ına bütün tanımlamaları yaptıktan sonra uzak bilgisayara ping atabilir miyiz?

Hayır...

Biz sadece İstanbul Router'da Static Routing yaptık. Malatya Router'da hiçbir işlem yapmadığımızdan dolayı Malatya Routerı ping işlemine cevap vereceği ip adresine nasıl ulaşacağını bilemediği için Ping işlemi gerçekleşmeyecektir. (Ping işlemi iki yönlüdür, paket hedefe gider ve gelir.)

Daha oncede belirttiğimiz gibi Routerlar için Directly Connected networklerine herhangi bir yönlendirme yazmaya gerek yoktur. İki Routeri birbirine bağladığınızda ve interfacelerini uygun şekilde konfigüre edip up durumuna getirdiğinizde Routing Table'ında o networkler ile ilgili bilgileri goruruz.

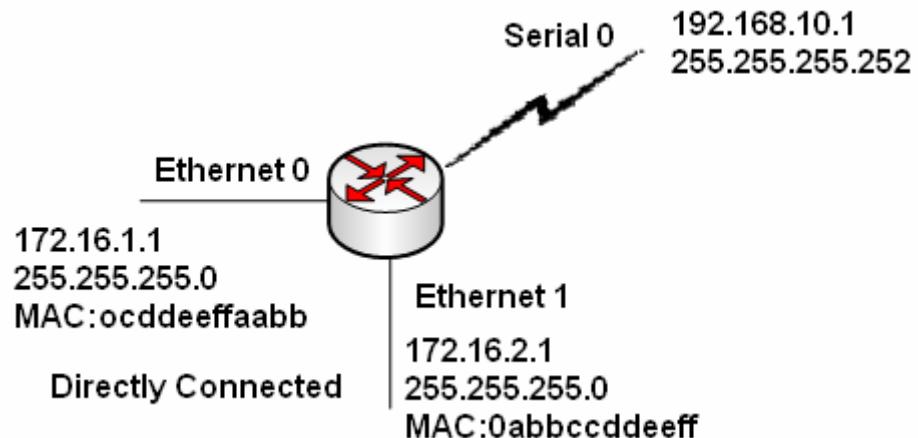


Boyle bir networkte interfaceleri up duruma getirdigimizde Routing Table'ı lar asagidaki gibi olacaktir.

```
RTA#show ip route
Codes: C - connected,..... .
C    172.16.0.0/16 is directly connected, Serial0
C    192.168.2.0/24 is directly connected, Ethernet0
```

```
RTB#show ip route
Codes: C - connected,..... .
C    172.16.0.0/16 is directly connected, Serial0
C    192.168.1.0/24 is directly connected, Serial1
```

```
RTC#show ip route
Codes: C - connected,..... .
C    10.0.0.0/8 is directly connected, Ethernet0
C    192.168.1.0/24 is directly connected, Serial1
```



ROUTING BASICS

Routerların temel işlevi yönlendirmek yapmaktadır. Bunu yaparken Router Routing Table'ında bulunan bilgilerle hareket eder. Routing table'ı bizler static olarak tanımlayabildiğimiz gibi Routing Protokoller vasıtasıyla oluşmasını da sağlayabiliriz. Anlaşılacağı gibi Routing işlemi iki ana başlık altında toplanabilir.

1. Static Routing
2. Dynamic Routing

Static Routing Ip Route komutu ile gerçekleştirilirken Dynamic Routing protokoller yardımıyla gerçekleşir.

Static Routing özellikle küçük ölçekli networklerde kullanıldığında ideal bir çözüm olarak karşımıza çıkabilir fakat büyük ölçekli networklerde çalışmaya başladığımız andan itibaren hata yapma olasılığımız artacaktır.

Dynamic Routing ise konfigürasyonu çok çok kolay olduğu için, mantığı anlaşıldığı andan itibaren birçok fayda sağlayacaktır.

Static Routing

Az önce de bahsettiğimiz gibi static Routing "ip route" komutu ile Global Configuration modda yapılır ve küçük ölçekli networklerde ideal çözümüdür.

Static Routing yapılrken hedef network adresi, subnet maskı ve bizi o hedefe götürecek bir sonraki routerın ip adresi bilinmelidir. Burada bir sonraki router ile ilgili bir kavram ortaya çıkıyor; "next hop". Bunlar bilindiğinde komut şu şekilde kullanılacaktır.

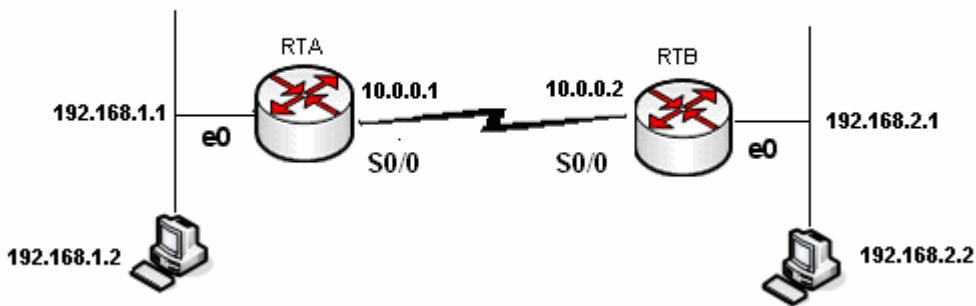
Router(config)#ip route [hedef adres][subnet mask][Next Hop] [distance]

Bu komut yönlendirme tablosundan silinmek istendiğinde ise başına "no" ifadesini yazmak yeterli olacaktır. Distance ifadesi seçimlik olup gereği durumlarda Routingler arasında önceliği belirlemeye yarayan Administrative Distance değerini değiştirmek için kullanılır. Static Routing için Administratif Distance default olarak "1" dir.

Default Administrative Distance değerleri sunlardır:

Administrative Distance Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IGRP summary route	5
External BGP	20
Internal Enhanced IGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP external route	170
Internal BGP	200
Unknown	255

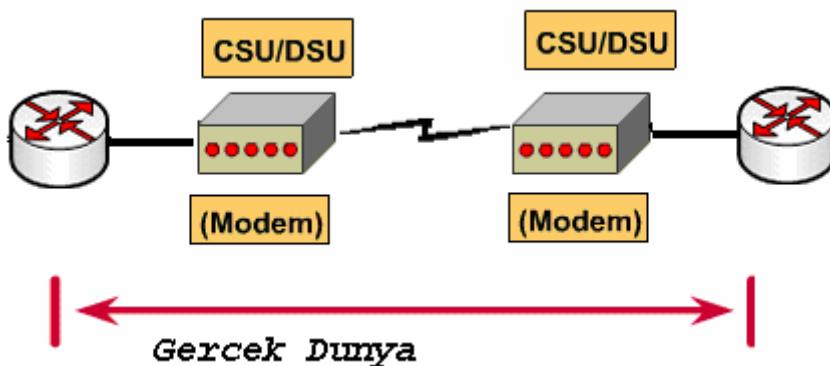
Static Routing, i ornek bir calisma ile inceleyecek olursak;



192.168.1.0 ve 192.168.2.0 10.0.0.0 networklerimiz var. Bütün subnet masklarımız 255.255.255.0 olsun. Bu durumda Static Routing işlemi her iki router için şu şekillerde gerçekleştirilmelidir. A Routerının serial 0 adresi 10.0.0.1 ve B routerının serial 0 adresi 10.0.0.2' dir.

NOT: Cisco Router'ların seri interface'leri DTE veya DCE olarak konfigure edilebilir. Bu özellik kullanılarak WAN bağlantıları simüle edilebilir. Bunun için birbirine bağlı Router'ların interface'lerinden bir tanesini DCE diğer Router'ın interface'sini ise DTE olarak kabul ediyoruz. Ardından DCE olarak kabul ettiğimiz interface'in DTE olan interface clock sağlaması gerekiyor. DCE olarak kullanabileceğimiz interface'de "**clock rate**" komutunu kullanarak bir değer atamamız gerekiyor. Aksi halde bağlantı çalışmayaçaktır. Örneğin;

RouterA(conf-if)#clock rate 64000





(CSU/ DSU)

Artık konfigurasyonumuza gecebiliriz.

Router A için;

```
H(config)#
A(config)#ip route 192.168.2.0 255.255.255.0 10.0.0.2
A(config)#

```

Router B için;

```
B(config)#
B(config)#ip route 192.168.1.0 255.255.255.0 10.0.0.1
B(config)#
B(config)#
B(config)#

```

Burada Next hop olarak her iki konfigurasyonda da bir sonra ki routerin ip adresi secildi. Zaten sistem de 2 tane Router oldugu icin bir sorun yasamadik. Bu noktada hedef networke ulasmak icin birden fazla Router gecildigi zaman next hop olarak hangisi secilmelidir sorusu aklimiza gelebilir.

Next Hop olarak o routelardan herhangi biri secilebilir, burda onemli olan konfigurasyonlar bittiği zaman Routerimizin next hop adresine nasıl ulasacagini bilip bilmediğidir.

Routerların konfigurasyonları ve problem cozumu aşamasında running-config dosyalarının incelenmesi onemlidir, cunki bu dosyada yaptigimiz her konfigurasyon adimini gorebiliriz.

Simdi topolojimizdeki A routeri icin running-config dosyalarina bir goz atalim.

```

hostname A
!
enable secret 5 $1$gZBQ$yyxVv/2B4uq7pROiHGRhg/
!
!
!
!
memory-size iomem 10
ip subnet-zero
!
!
!
voice-port 1/0/0
!
voice-port 1/0/1
!
!
interface Ethernet0/0
 ip address 192.168.1.1 255.255.255.0
!
interface Serial0/0
 ip address 10.0.0.1 255.255.255.0
 no fair-queue
 clockrate 64000
!
interface BRI0/0
 no ip address
 shutdown
 isdn x25 static-tei 0
!
ip classless
ip route 192.168.2.0 255.255.255.0 10.0.0.2
ip http server
!
!
line con 0
line aux 0
line vty 0 4
!
no scheduler allocate
end
-
```

NOT: CCNA Sınavlarında DCE ve DTE olacak interface belirtilmektedir ve konfigürasyon sorularında DCE olan interface' lere Clock Rate verilmelidir.

"Show ip route" komutu ile yönlendirme tablosunu görebiliriz.

```

RouterA#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

```

Gateway of last resort is not set

```

10.0.0.0/24 is subnetted, 1 subnets
C    10.0.0.0 is directly connected, Serial0/0
C    192.168.1.0/24 is directly connected, Ethernet0/0
S    192.168.2.0/24 [1/0] via 10.0.0.2

```

RouterA#_

(RouterA için yönlendirme tablosu)

Yönlendirme tablosunda "S" ile başlayan satırlar Statik bir yönlendirme yapıldığını ve şekeiten hareketle bu yönlendirmenin 192.168.2.0 network' üne, 10.0.0.2 next hop'undan giderek olduğunu söyler. Bu tabloda C ile başlayan satırlar ise A router'ının interfacelerine direkt olarak bağlanmış networkleri gösterir ve bu networklere "Directly Connected" networkler denir. Routerlar kendi Directly Connected networklerini bilirler ve bu networkler ulaşmak için yönlendirme yapılmasına gerek yoktur.

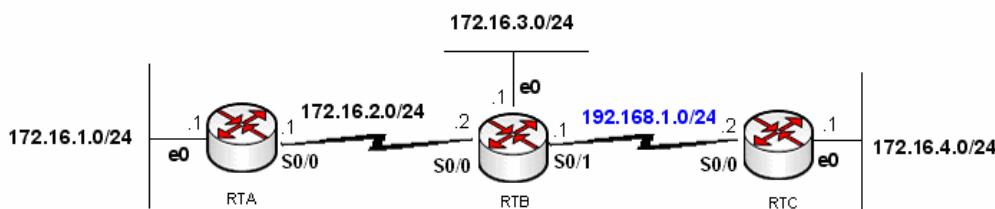
Senaryomuzda hiçbir yönlendirme yapmasaydık bile A router'ının ethernet interface' ine bağlı bir bilgisayardan B router'ının serial interface' ine ping atabiliirdik. Bunun için tek yapmamız gereken şey, o bilgisayarda Default Gateway' i (Varsayılan Ağ Geçidi) 192.168.1.1 olarak konfigüre etmektir.

RouterA#show ip interface brief					
Interface	IP-Address	OK?	Method	Status	Prot
Ethernet0/0	192.168.1.1	YES	NVRAM	up	up
Serial0/0	10.0.0.1	YES	manual	up	up

("show ip interface brief" komutu ile interface' leri durumunun görüntülenmesi)

Routing Table

Routing Table konfigurasyonlarımız ve projelerimiz sırasında problem teshisimiz açısından çok önemlidir. İyi bir network yöneticisi running-config ve routing table' a hakim olmalıdır. Isimizi Routing oladugu durumda Routing Table bir numarali yardımçımız olacaktır.



RouterB#show ip route

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
 D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
 N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
 i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter:
 area
 * - candidate default, U - per-user static route, o - ODR
 P - periodic downloaded static route

Gateway of last resort is not set

RouterB#

Iste ornek bir network topolojisi B routerinin Routing Table'si. Routing Table gorundugu üzere bombos. Burada Directly Connected networklerin bile gorunmemesinden dolayi interfaceler ile ilgili bir sorun oldugundan bahsedilebilir. Sorunun ne oldugu ile ilgili bilgiyi "show ip interfaces brief" komutu ile goruntuleyebiliriz. Su anda anlamamiz gereken nokta, eger bir interface sebebi ne olursa olsun "down" ise o interface bagli network Routing Table, da gorunmez!

```

RouterB(config)#inter s 0
RouterB(config-if)#ip add 172.16.2.2 255.255.255.0
RouterB(config-if)#end
  
```

```

RouterB(config)#interface s 1
RouterB(config-if)#ip add 192.168.1.1 255.255.255.0
RouterB(config-if)#no shutdown
  
```

```

RouterB(config)#interface fastethernet 0
RouterB(config-if)#ip add 172.16.3.1 255.255.255.0
RouterB(config-if)#no shutdown
  
```

Router B icin interfaceleri up duruma getirdikten sonra artik en azindan Routing Table' imizda Directly Connected networklerimizi gormemiz gerekir.

```

RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
<text omitted>

Gateway of last resort is not set

  172.16.0.0/24 is subnetted, 2 subnets
C      172.16.2.0 is directly connected, Serial0
C      172.16.3.0 is directly connected, FastEthernet0
C      192.168.1.0/24 is directly connected, Serial1
RouterB#
  
```

Burada 172.16.0.0 networkkune Parent route ve o networkun subnetworku olan 172.16.2.0 - 172.16.3.0 networklerine Child Route denir.

Bilindiği gibi Static Route yazılırken Routerin interface' i yada Next Hop ip adresi kullanılabilir. Routerin interface' i kullanıldığında o static route satırı interface ile direk bağlı bir network gibi gorunecektir.

```
RouterB(config)#ip route 172.16.1.0 255.255.255.0 serial 0
RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
<text omitted>
```

Gateway of last resort is not set

```
172.16.0.0/24 is subnetted, 3 subnets
S      172.16.1.0 is directly connected, Serial0
C      172.16.2.0 is directly connected, Serial0
C      172.16.3.0 is directly connected, FastEthernet0
C      192.168.1.0/24 is directly connected, Serial1
RouterB#
```

Next Hop ip adresi kullanıldığında ise Routing Table o networke verilen ip adresi ile ulaşabileceğini gösteren satır yer alacaktır.

```
RouterB(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.1
RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
<text omitted>
```

Gateway of last resort is not set

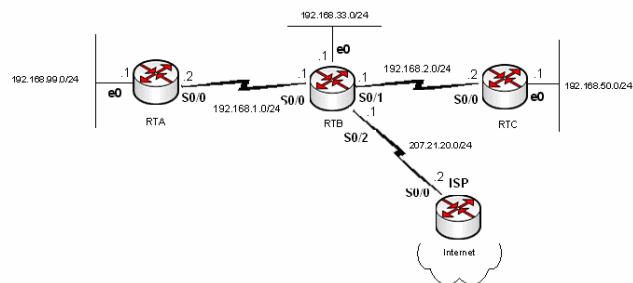
```
172.16.0.0/24 is subnetted, 3 subnets
S      172.16.1.0/24 [1/0] via 172.16.2.1
C      172.16.2.0 is directly connected, FastEthernet1
C      172.16.3.0 is directly connected, FastEthernet0
C      192.168.1.0/24 is directly connected, Serial1
RouterB#
```

Interfaceler "up" olduğu surece Routing Tablelarda bozulma olmayacağı.

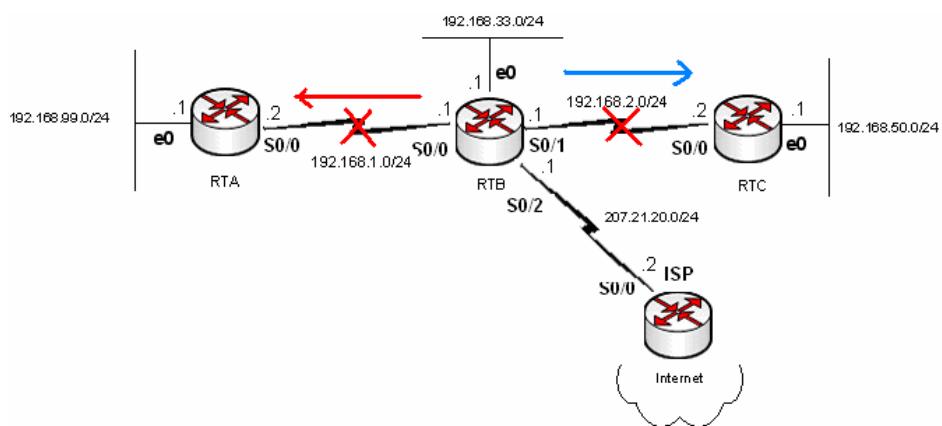
```

S 192.168.99.0/24 [1/0] via 192.168.1.2
S 192.168.50.0/24 is directly connected, Serial2
C 192.168.1.0/24 is directly connected, Serial1
C 192.168.2.0/24 is directly connected, Serial2
C 192.168.33.0/24 is directly connected, FastEthernet0

```



Ornegin sekildeki yapı içerisinde RTB Routeri için butun interfaceler up durumdayken Routing Table sorunsuz gorunuyor. 192.168.1.0 ve 192.168.2.0 networklerinin bulunduğu interfacelerin bir an icin down oldugunu dusunelim.



Bu durumda Routing Table RTB için şu şekilde olacaktır.

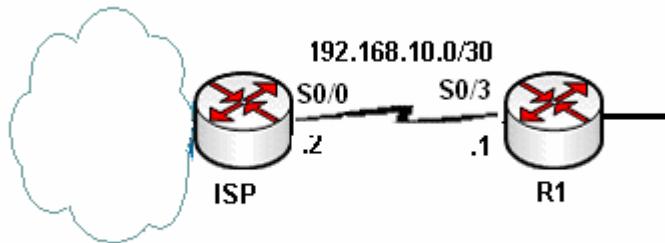
```

RTB#show ip route
C 207.21.20.0/24 is directly connected, Serial0
C 192.168.33.0/24 is directly connected, FastEthernet0

```

Şu konusu Interfacelere direkt bağlı olan networkler ve o interfaceleri kullanarak yazılan Static Route satırları artık Routing Table'da yoklar.

Default Routing



Burada ISP Routeri ile baglanilan networke (Internet) R1 üzerinde default route yazilarak ulasilabilecektir.

Default hedefi bilinmeyen paketleri yonlendirmek icin uazilabilecek Route satiridir seklinde tanimlanabilir.

```
R1(config)#ip route 0.0.0.0 0.0.0.0 Serial3/0 yada
```

```
R1(config)#ip route 0.0.0.0 0.0.0.0 192.168.10.2
```

Seklinde yazilabilir.

Routing Table' da asagidaki gibi gorunecektir.

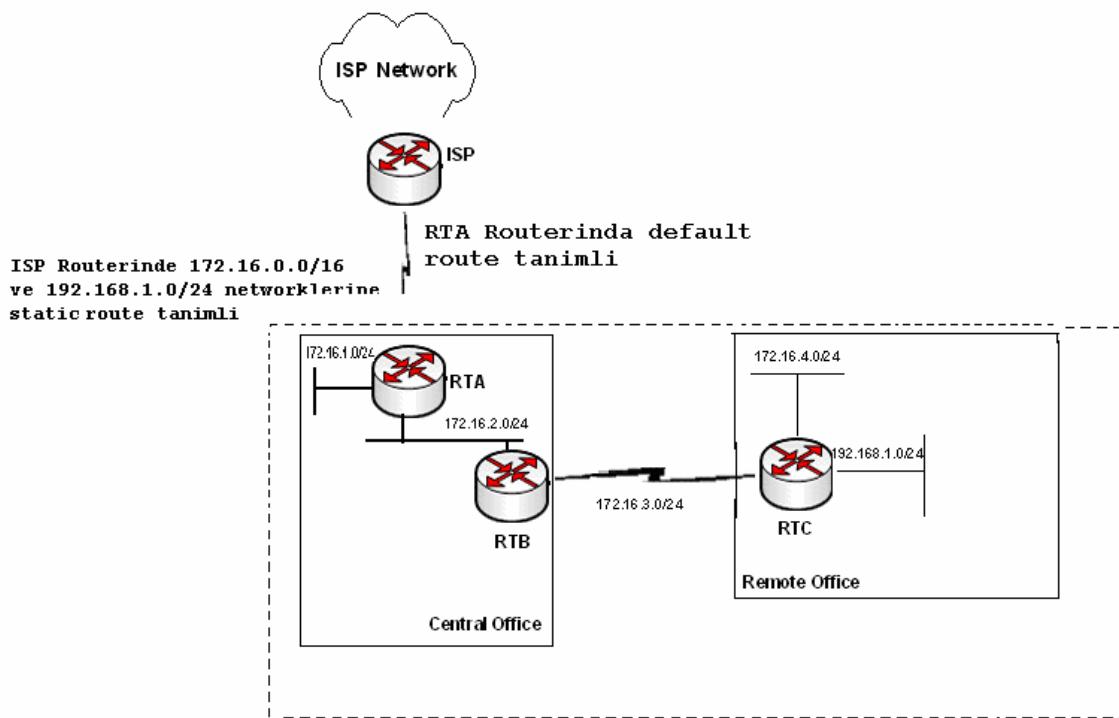
```
192.168.10.0/30 is subnetted, 1 subnets
C      192.168.10.0 is directly connected, Serial3/0
S*    0.0.0.0/0 is directly connected, Serial3/0
```

Extra

Default tanimlamak bazen sorunlar ile birlikte gelebilir.

Cunki uzerinde Default tanimli ve bu route satiri ile paketleri interne gonderen bir router, sisteminde bulunan diger networklere olan yolu down oldugunda o networklere gelen paketleri de default route satirina gore degerlendirecektir.

Ornek uzerinde incelemek gerekirse;



Boyle bir yapı içerisinde RTB ve RTC arasındaki bağlantının down olduğunu düşünürsek Routing Table lar update edildikten sonra RTA ve RTB routerları 172.16.4.0 ve 192.168.1.0 hedef networklerine giden yolları bilmeydikleri için hedefinde bu networkler bulunan paketleri default route satırından hareketle ISP routerına gönderecekti.

ISP Routeri da kendine gelen bu paketleri üzerinde tanımlı static route satırlarından hareketle tekrar geri gönderecek ve bu sebeple bir dongu olusmasına sebep olacaktır. Bu dongu IP basılığında TTL (Time – to – live) alanı sıfırlanana kadar devam edecektir.

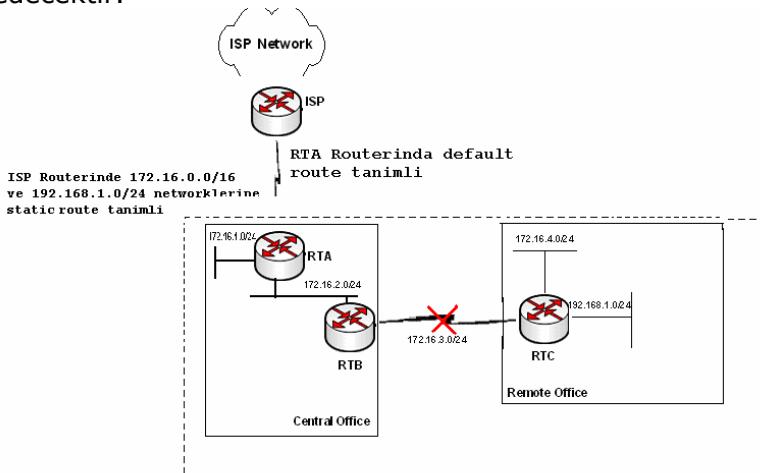
Bunun için kullanılacak çözüm RTA üzerinde Discard Route denen tanımlamayı yapmaktadır.

`Discard Route routing table' da bir eşlesme olmadığına ve default route' un işletilmesi istenmediğinde kullanılır ve paketler null0'a gönderilir.`

Orneğin;

```
RTA(config)#ip route 172.16.0.0 255.255.0.0. null0
```

Satırı ile RTA Routeri kendisine gelen hedefinde 172.16.0.0 networku bulunan paketleri drop edecektir.



Boyle bir durumda bir baska cozumde "no ip classless" komutunu kullanmaktir.

Bu komut kullanildiktan sonra Router soz gelimi hedef ip adresi 172.16.4.9 olan bir paket icin routing table' ina bakacak ve en uygun yolu arayacak. Bu durumda parent network olarak 172.16.0.0. networkunu ve bu networkun altında bilinen 172.16.1.0, 172.16.2.0 networklerini bulacak.

"no ip classless" ile konfigure edilmis bir router her ne kadar parent networklerde 172.16.0.0 olsa da 172.16.4.9 ip adresini iceren 172.16.4.0 networku bilinen networkler arasında olmadigi icin paketi drop edecektir.

Fakat bu cozum onerilen bir cozum degildir. Ornegimiz iceriside de var olan ornegin 192.168.1.0 gibi bir networkte ise yaramaz. Cunki bu network herhangi bir parent networkun subnetworku degildir.

Dolayisiyla bu network icin null0 kullanilmalidir.

Dynamic Routing

Static Routing ile çalışmalarımız sırasında Router' a ihtiyacı olan balıkları verdik ama artık balık ihtiyacı arttı yani networkler büyümeye başladılar. Dolayısıyla artık onlara balık tutmayı öğretmenin zamanı da geldi. :=)

Dynamic Routing' te Static Routing' de olduğu gibi sabit bir tanımlama yapmak yerine her Router' a kendi Directly Connected networklerini, çeşitli Routing Protokoller ile tanımlıyoruz. Ve ilgili Routing protokolün çalışma mantığına göre en iyi yol seçimi (Best Path Determination) Router tarafından gerçekleştiriliyor.

Burada bahsettiğimiz Routing Protokolleri üç başlık altında incelememiz mümkün.

1. Distance Vector Protokoller (RIP, IGRP)
2. Link State Protokoller (OSPF)
3. Hybrid Protokoller (EIGRP)

Distance Vector protokoller routing table update mantığıyla çalışırlar. Yani belirli zaman aralıklarında sahip oldukları network bilgilerini komşu routerlarına gönderirler ve komşu routerlarından da aynı bilgileri alırlar. Bu döngünün sonunda her router sisteme ki bütün networkler öğrenmiş olur ve uygun yol seçimini yapar.

Link State Protokoller ise sürekli bir update yapmak yerine, komşu routerlarının up olup olmadıklarını anlamak için küçük "Hello" paketleri gönderirler. Sadece gerektiği zamanlarda, yeni bir router ortama eklendiğinde veya bir router down olduğunda, sadece o bilgi ile ilgili update gerçekleştirirler.

Hybrid Protokoller hem Distance Vector hem de Link State protokollerin bazı özelliklerini taşır. Bu gruba üye olan EIGRP Cisco tarafından ortaya çıkarılmıştır ve sadece Cisco routerlarda çalışır.

Her gruba üye olan protokoller ile ilgili detaylı bilgi ilerleyen başlıklar altında verilecektir.

Distance Vector Protokoller

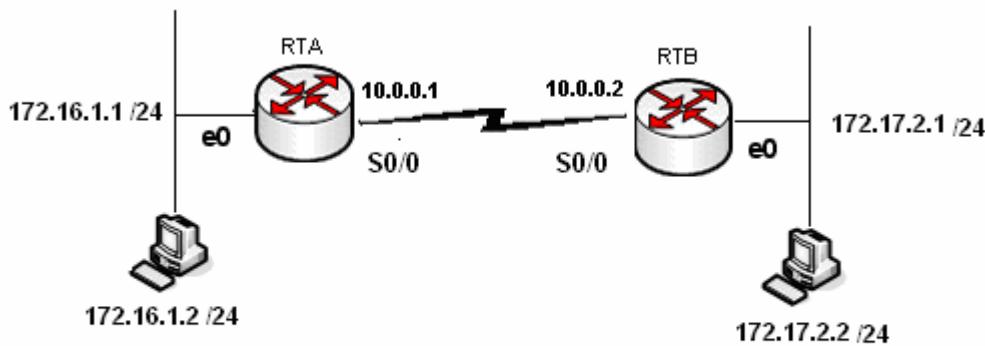
RIP (RIPv1)

Rip (Routing Information Protocol) en iyi yol seçimi yaparken tek kriter olarak hop sayısına bakar. Rip tanımlanarak oluşturulmuş bir networkte maksimum hop sayısı 15' dir be 16. hop' tan sonra Destination Unreachable hatası verecektir.

Rip ile tanımlanan routerlar her 30 saniyede bir kendisinde tanımlı olan networkleri komşu routerlarına iletirler. Burada dikkat edilmesi gereken bir konu, RIP ile tanımlanan bir networkün bağlı bulunduğu interface' i, aynı zaman da routing update gönderilecek bir interface olarak seçiyor olmamızdır.

Rip classfull bir routing protokoldür. Yani konfigürasyon sırasında subnet mask girilemez ve subnet masklar update sırasında ip adresinin sınıfına ait subnet mask seçilerek gönderilir.

Rip konfigürasyonu diğer bütün routing protokoller de olduğu gibi oldukça basittir. (Bütün subnet masalar 255.255.255.0)



Bu senaryoyu Rip ile konfüre edecek olursak;

```

RouterA(config)#router rip
RouterA(config-router)#net
RouterA(config-router)#network 172.16.1.0
RouterA(config-router)#net
RouterA(config-router)#network 10.0.0.0
RouterA(config-router)#exit
RouterA(config)#_

```

(RouterA Rip Konfigürasyonu)

```
RouterA#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
```

Gateway of last resort is not set

```
R    172.17.0.0/16 [120/1] via 10.0.0.2, 00:00:17, Serial0/0
    172.16.0.0/24 is subnetted, 1 subnets
C      172.16.1.0 is directly connected, Ethernet0/0
    10.0.0.0/24 is subnetted, 1 subnets
C      10.0.0.0 is directly connected, Serial0/0
RouterA#_
```

(RouterA Yönlendirme Tablosu)

```
RouterB(config)#router rip
RouterB(config-router)#network 172.17.2.0
RouterB(config-router)#net
RouterB(config-router)#network 10.0.0.0
RouterB(config-router)#
RouterB(config-router)#exit
RouterB(config)#
```

(RouterB Rip Konfigürasyonu)

```
RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route
```

Gateway of last resort is not set

```
    172.17.0.0/24 is subnetted, 1 subnets
C      172.17.2.0 is directly connected, Ethernet0/0
R    172.16.0.0/16 [120/1] via 10.0.0.1, 00:00:00, Serial0/0
    10.0.0.0/24 is subnetted, 1 subnets
C      10.0.0.0 is directly connected, Serial0/0
RouterB#_
```

(RouterB Yönlendirme tablosu)

Routing Table' ımıza "Show ip route" komutu ile baktığımızda başında R harfi bulunan satırlar görüyoruz. Buradan çıkartacağımızı anlam şu: Bu satırlarda belirtilen networklerin bilgisi Rip protokol sayesinde başka routerlardan update yoluyla gönderildi.

Yine Routing Table dikkatli izlendiğinde köşeli parantez içindeki [120/1] gibi ifadeler görünüyor. Burada 120 Rip protokol için Administrative Distinct denen ve routing protokoller arasında ki önceliği belirleyen değerdir. Diğer ifade da "n" gibi bir sayıdır (burada 1) ve hedef networke ulaşmak için aşılabilecek hop sayısıdır.

```

RouterA#sh ip interface brief
Interface          IP-Address      OK? Method Status      Prot
ocel
Ethernet0/0        172.16.1.1    YES manual up       up
Serial0/0          10.0.0.1      YES manual up       up
Serial0/1          unassigned    YES unset administratively down down

RouterA#ping 172.17.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/29/33 ms
RouterA#

```

Rip protokolü update'lerini broadcast adresi olan 255.255.255.255 ip'inden yapar. "debug ip rip" komutunu verdığımızda bunu açıkça görürüz.

```

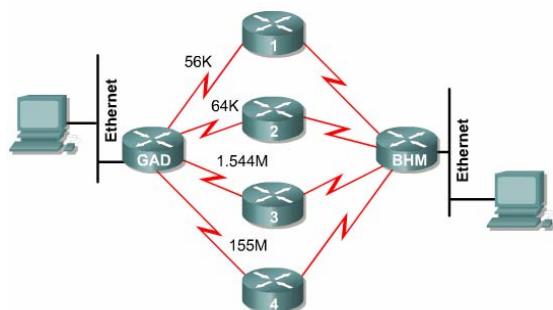
RouterA#debug ip rip
RIP protocol debugging is on
RouterA#
00:37:18: RIP: sending v1 update to 255.255.255.255 via Ethernet0/0 (172.16.1.1)

00:37:18: RIP: build update entries
00:37:18:   network 10.0.0.0 metric 1
00:37:18:   network 172.17.0.0 metric 2
00:37:18: RIP: sending v1 update to 255.255.255.255 via Serial0/0 (10.0.0.1)
00:37:18: RIP: build update entries
00:37:18:   network 172.16.0.0 metric 1_

```

Rip Load Balancing

Load Balancing tam olarak yükü birden fazla yol arasında dağıtmak demektir.



(Routerlar metricler eşit olduğu için load balancing yapar)

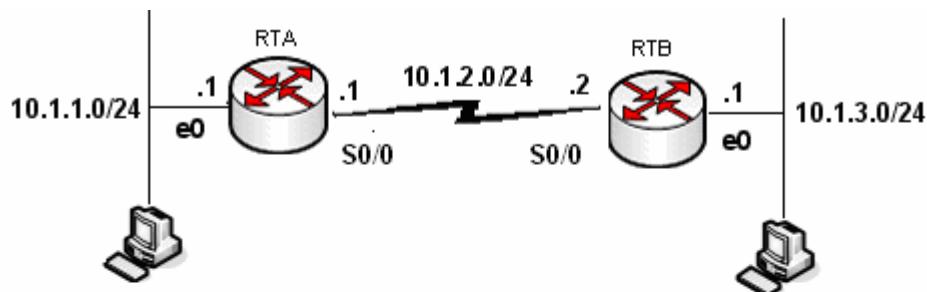
Mantıksal olarak düşündüğümüz de Rip'ın load balancing yapma ihtimali her zaman vardır. Çünkü referans olarak bir tek hop sayısına bakar. Oysa diğer protokoller de load balancing ihtimali en iyi yol seçimi sırasında bir çok kriter göz önüne alındığı için mucize derecesinde zayıf bir ihtimaldir. Fakat ileride deagineceğimiz IGRP ve EIGRP protokollerinde fazladan bir komut kullanarak Routerin load balancing yapması sağlanabilir.

Split Horizon

Bir Router kendi directly connected networkünü başka bir router'dan da öğrenirse öğrendiği bilgiyi çöpe atar.

Ayrıca router'ın ağ üzerinde herhangi bir değişiklik olduğunu anladığında bu değişikliği, öğrendiği interface haricindeki interface'lerden yayılmasını sağlar. Böylece router'lar değişikliği sadece bir yönde yayırlarlar.

Aşağıdaki örnek ile Split Horizon kuralını detaylı anlayabiliriz.

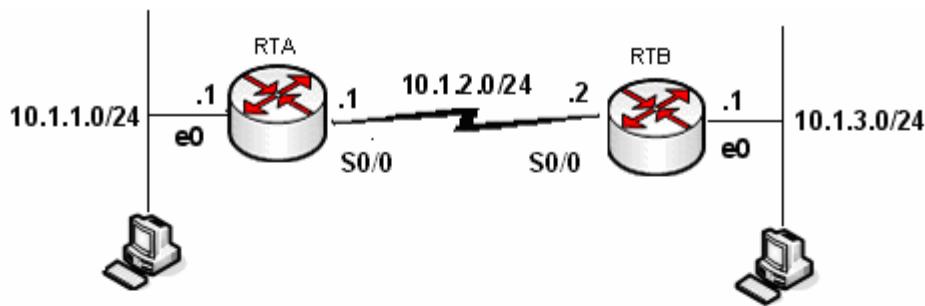


Routing Table			
Net.	Hops	Ex-Int	
10.1.1.0/24	0	e0	
10.1.2.0/24	0	s0	

Routing Table			
Net.	Hops	Ex-Int	
10.1.2.0/24	0	s0	
10.1.3.0/24	0	e0	

İki adet Routerimiz var ve başlangıçta Routing Table'lar şekildeki gibi oluşmuş durumda yani sadece Directly Connected networkleri biliyorlar.

Split Horizon Disable edildiği zaman Routerlar Routing Table'larında ki bütün networkleri ve herhangi bir interfacelerinden öğrendikleri bütün networkleri update edeceklerdir.



Routing Update		
	Next-hop	
Net.	Hops	Address
10.1.1.0/24	1	10.1.1.1
10.1.2.0/24	1	10.1.1.1

Routing Update		
	Next-hop	
Net.	Hops	Address
10.1.2.0/24	1	10.1.2.2
10.1.3.0/24	1	10.1.2.2

Routing Table		
Net.	Hops	Ex-Int
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

Routing Table		
Net.	Hops	Ex-Int
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

Routerlar şekilde gösterilen update'leri komsu Routerlarına yapacaklar. Burada kırmızı ile gösterilmiş Directly Connected networklerinde update edildigine dikkat edin. Bu update Split Horizon'un disable olmasının sonucudur.

Routing Table incelendiginde bir sorun yok gibi gorunuyor. Gercekten de yok, cunki split horizon'un disable olmasından kaynaklanan update'ler daha yüksek metrig'e sahip olduğu için Routing table' larda yer almıyor.

Simdi bir sonraki update'lere bakalım.

Routing Update		
	Next-hop	
Net.	Hops	Address
10.1.1.0/24	1	10.1.1.1
10.1.2.0/24	1	10.1.1.1
10.1.3.0/24	2	10.1.1.1

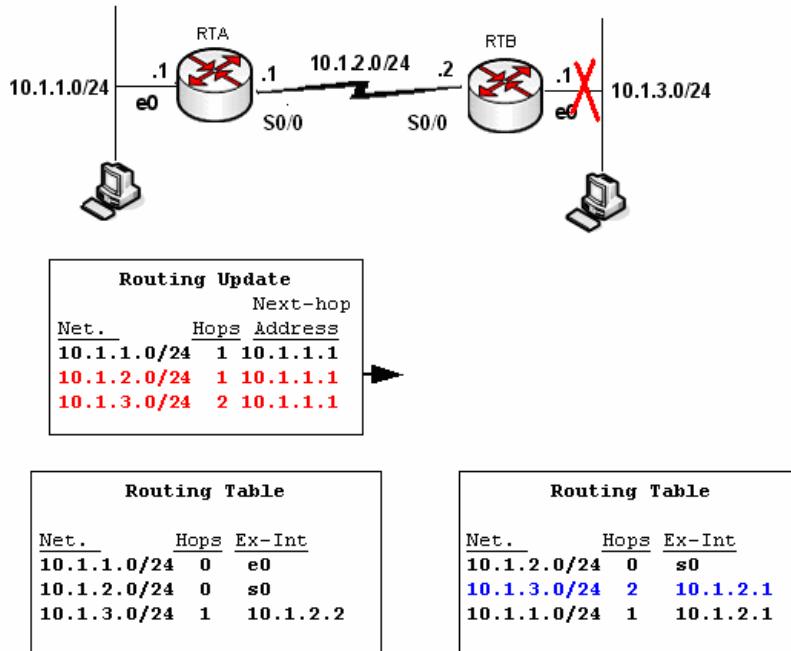
Routing Update		
	Next-hop	
Net.	Hops	Address
10.1.2.0/24	1	10.1.2.2
10.1.3.0/24	1	10.1.2.2
10.1.1.0/24	2	10.1.2.2

Routing Table		
Net.	Hops	Ex-Int
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

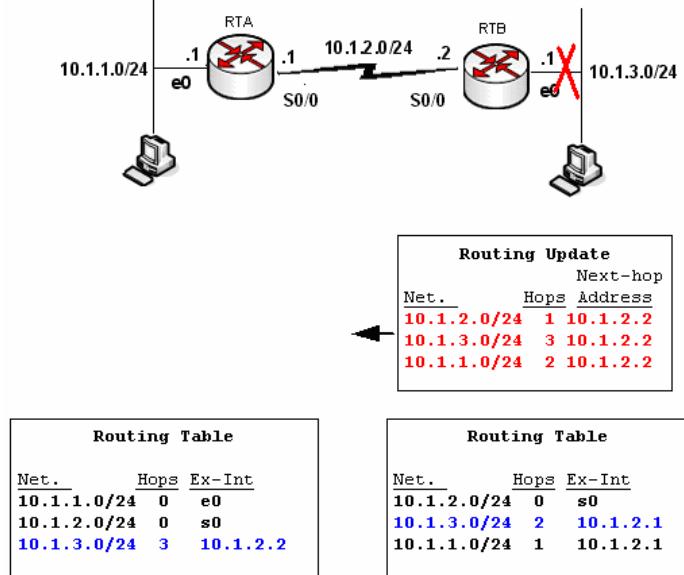
Routing Table		
Net.	Hops	Ex-Int
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

Burada daasinda update edilmemesi gereken networkler update edilmiş olmasına rağmen bir sorun yok cunki o networkler daha büyük metric ile update ediliyor. Örneğin RTA Routeri 10.1.3.0 networkunu serial 0 interfaceinden aldığı için split horizon disable edilmemiş olsaydı o interfaceden geriye update etmeyecekti.

Bu ana kadar bir sorun olmadı ama bir an için RTB Routerina bağlı olan 10.1.3.0 networkünün down olduğunu varsayıyalım.

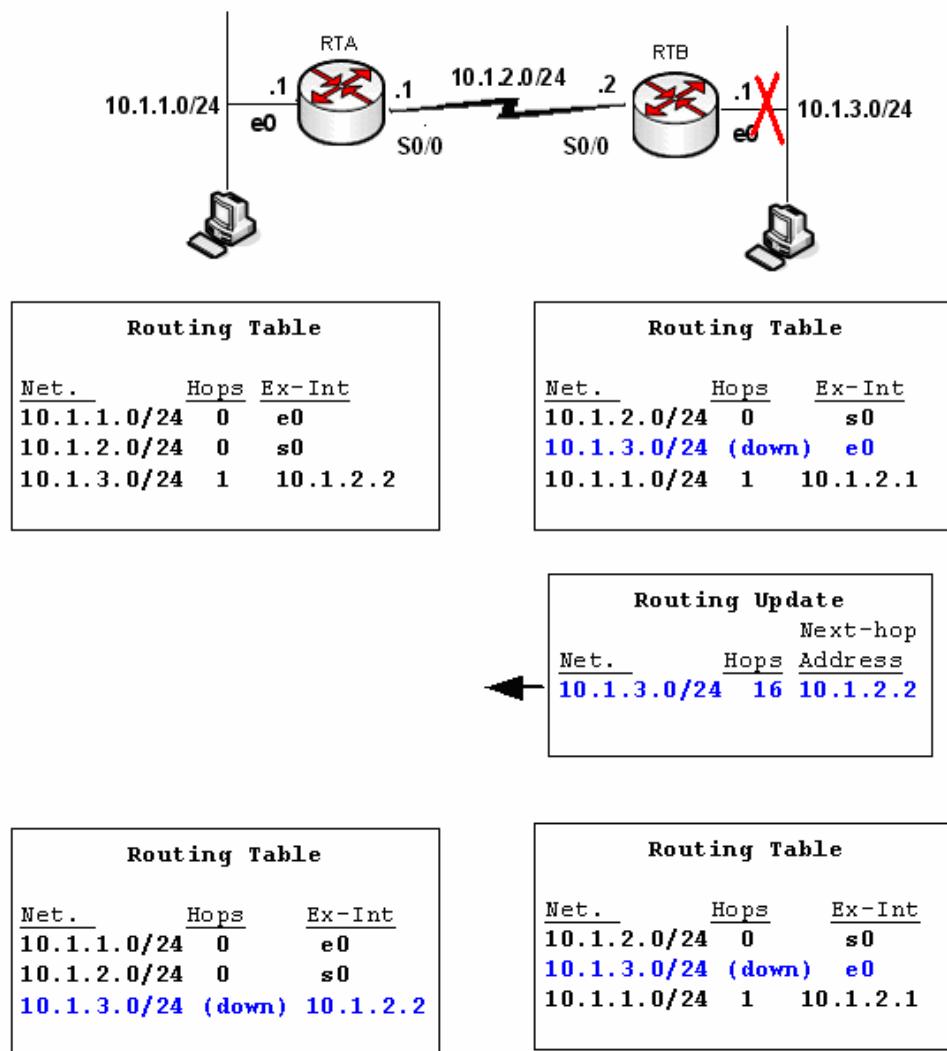


Bu durumda RTB routerına RTA routerinden 10.1.3.0 networku kendisine bağlı olan network down olduğu için daha küçük metric ile geliyormus gibi olacak ve RTB routerının Routing Table'ında sekildeki gibi yer alacak. Su anda RTB Routeri bir sure önce kendisine direkt bağlı olan networke diğer router üzerinden 2 hop gecerek gidebileceğini sanıyor. RTB updatelarını üstelik yanlış olan Routing Table'ına dayanarak yapacaktır.



Ve bu updatelerden sonra RTA routerinin Routing Table, ida 10.1.3.0 networkune 3 hop ile gidilebilecegi kanisinda. Bu dongu ta ki hop sayisi 16 oluncaya kadar devam edecktir. (Rip maximum 16 hop ilerleyebilir)

Bu dongunun engellenmesi Split Horizon ile mumkundur.



Split Horizon enable oldugunda RTB routeri aninda Triggered Update gonderir komsu routerina ve bu update bilgisi soz konusu networkun 16 hop ile ulasilacagi seklindedir ki rip soz konusu oldugunda bu RTA nin da o networku down olarak varsayacagi anlamina gelir.

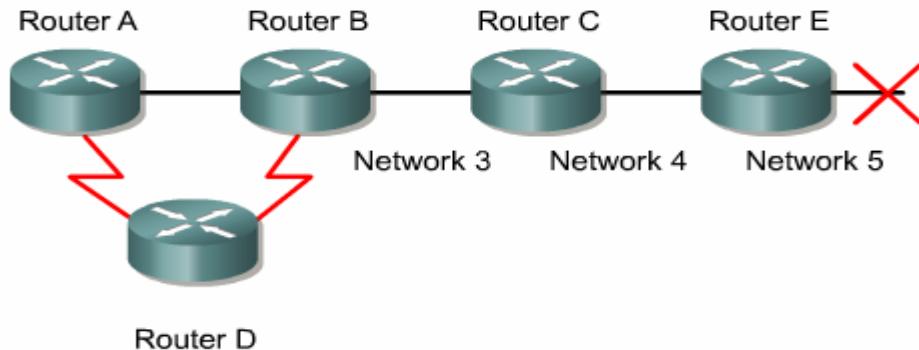
Dolayisiyla her ikir router da 10.2.3.0 networku icin Hold Down Timer' i baslatirlar.

Bu calisma yapısına Split Horizon with Poison Reverse denir ki Routerlar, da default olarak enable durumdadır. Disable edilme gereken zamanlar da ki CCNA 4 içerisinde bu konudan bahsedegiz, asagidaki komut kullanilabilir.

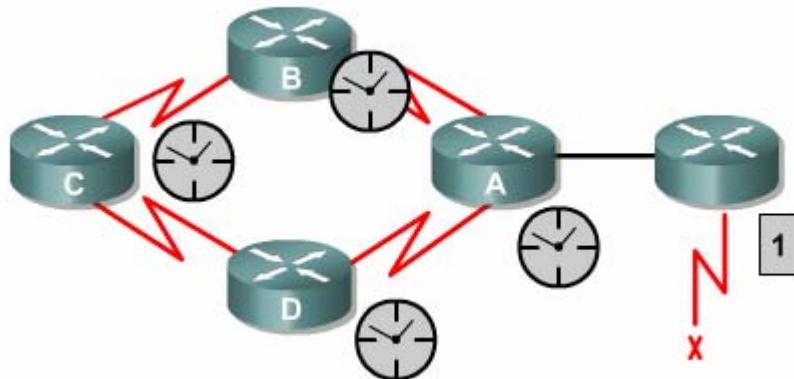
Router (config-if) # no ip split-horizon

Route Poisoning

Router'ların yönlendirme tablosuna hop count değer 16 olarak yazılan bir yönlendirmedir ve hedef adresin erişilemez olduğunun router'lar arasında bilinmesini sağlar.



Holddown Timers



Bu teknikte hold-down sayıcıları router'ın komşusundan aldığı ulaşılamaz bir ağa ait güncelleme ile başlar. Eğer aynı komşudan aynı ağa ait daha iyi bir metric değerine sahip bir güncelleme bilgisi alırsa hold-down kaldırılır. Fakat hold-down değeri dolmadan aynı komşudan daha düşük bir metric değerine sahip bir güncelleme gelirse bu kabul edilmez.

Triggered Updates

Routing Table, da bir değişiklik olduğu anda Routerlar tarafından gönderilen update'lerdir. Topoloji değiştiği anda bunu farkeden Router periodic update süresini beklemeden değişikliği komşu Routerlarına bildirir.

Triggered Update'ler Route Poisoning ile同步 çalışırlar.

Extralar

Timers Basic ve update timer komutları ile Rip update, holdwoen v.s süreleri değiştirilebilir.

```
Router(config-router)#timers basic update invalid holddown flush
```

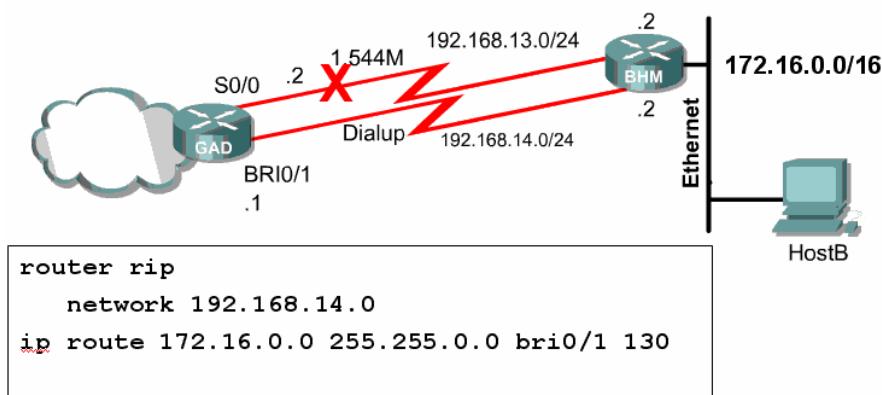
```
Router(config-router)#update-timer seconds
```

Rip ve Floating Static Route

Floating Static Routelar backup route olarak tanımlanmış route' lardır. Bu Route' lar reel olarak çalışan route lara göre daha yüksek bir Administrative Distance ile konfigüre edilmelidir. Rip ile çalışan bir Floating Static Route için Administrative Distance değeri 120' den büyük olmalıdır.

Bu durumda Rip sorunsuz çalıştığı surece Floating Static Route Routing Table' da görülmeyecek ancak Rip devre dışı kaldığında çalışmaya başlayacaktır.

Ve Rip tekrar aktif olarak çalışmaya başlarsa devre dışı kalacaktır. WAN bağlantısının sürekli up olmasını isteyen müşteriler için ideal çözümdür. Alternatif olarak örneğin ISDN bağlantıları Floating Static Route ile backup için önerilir.



Ornegimiz de iki nokta arasında 1,5 Mbitlik bir baglanti var ve bu baglanti Rip ile konfiguredilmiştir. Ayni iki nokta arasında dial-up bir baglanti var bu da Floating Static Route ile konfiguredilmiştir ve Administrative Distance için Ripinden daha büyük olan "130" seçilmiştir.

(Burada ki bri 0/1 portu ISDN bağlantıları için kullanılan porttur, CCNA 4 içerisinde detaylı olarak anlatılacaktır.)

Dolayısıyla Rip ile çalışan hat down olduğunda dial-up bağlantı devreye girecek ve hat tekrar aktif olduğunda devreden çıkışacaktır.

IGRP (Interior Gateway Routing Protocol)

IGRP yol seçimi yaparken K1' den K5' e kadar 5 ayrı değere bakar. Burarda kullanılan en etkin değer bant genişliğiyle ifade edilen K1 değeridir.

K1: Bant Genişliği

K2: Yuk

K3: Gecikme

K4: Güvenilirlik

K5: MTU (Maximum Transmission Unit)

```
Router> show interfaces s1/0
Serial1/0 is up, line protocol is up
Hardware is QUICC Serial
Description: Out to VERIO
Internet address is 207.21.113.186/30
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
  rely 255/255, load 246/255
Encapsulation PPP, loopback not set
Keepalive set (10 sec)
<output omitted>
bandwidth
delay
reliability
load
```

Burada büyük cogulukla etki eden degen bant genisligi degeridir. Routerlar bant seri interface'lerindeki genisliklerini anlayamazlar bu yuzden bizim verdigimiz yada default olan degerleri kullanırlar. Default olarak bir Cisco Router' in seri interface' i 1,5 M.bit olarak calisir, daha dogrusu hesaplarini bu deger ile yapar. Bu 1,5 Mbit ile calisildifi anlamina gelmez.

Metric degerlerinin anlamlı olması için gerçek bant genisligi interfacelere atanmalıdır. Bunun için "bantwidth *bantgenisligi(kbit)*" komutu kullanılır.

```
Router(config-if)# bandwidth kilobits
```

IGRP AD ve Timers

IGRP' nin Administrative Distance' i 100 ' dır ve dolayısıyla aynı routerda Rip ile birlikte kullanılacak olursa önceliğe sahip olacak, Router en iyi yol seçimini IGRP mantığından hareketle yapacaktır.

IGRP default olarak 90 saniyede Routing Table' ini komsu Routerlarina 255.255.255.255 broadcast adresi üzerinden update eder.

Yine default olarak 3x90 yani 270 saniye sonra hala update gelmeyen networklerini invalid varsayar fakat bu network bilgisini Routing Table; indan silmez, ayrıca bu network ile ilgili daha büyük metricli update'leri kabul etmez.

Daha büyük metrike sahip update'leri ancak Holddown Timer suresinin sonunda kabul eder ki bu sure 280 saniyedir. Artık bu noktadan sonra IGRP ile kofigured edilmiş Router kaybettigi network bilgisini silmese bile daha büyük metric ile gelebilecek update'leri kabul edecektir.

Kaybettigi networkun bilgisini ise Flush Timer suresinin sonunda silecektir. Bu sure de default olarak 630 saniyedir.

"show ip protocols" komutu ile bu sureler görüntülenebilir.

```

RouterB#show ip protocols
Routing Protocol is "igrp 101"
  Sending updates every 90 seconds, next due in 51
  seconds
  Invalid after 270 seconds, hold down 280, flushed
  after 630
    Outgoing update filter list for all interfaces is
    Incoming update filter list for all interfaces is
    Default networks flagged in outgoing updates
    Default networks accepted from incoming updates
    IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
    IGRP maximum hopcount 100
    IGRP maximum metric variance 1
    Redistributing: igrp 101
    Routing for Networks:
      192.168.2.0
      192.168.3.0

```

Tipki Rip' te oldugu gibi timers basic komutu ile default olan bu sureler degistirilebilir. Takrar default degerlere donulek istendiginde ise "no timers basic" komutu kullanilmalidir.

```

Router(config-router)#router igrp 100
Router(config-router)#timers basic update invalid holddown
  flush [sleeptime]
Router(config-router)# no timers basic

```

IGRP maksimum hop sayisi yonunden Rip'e gore ustundur, maksimum 255 hopa kadar calisir. Fakat Cisco ozel olmasi yuzunden dezavantajlidir, farkli ureticilere ait routerlarin oldugu sistemlerde kullanilamaz.

IGRP Load Balancing

Her Routing protocol esit metricli yollara yuk dagitimi yapar ancak IGRP konusan Routerlardan esit olmayan yollar icin load balancing yaptirilabilir. (Bu durum EIGRP tarafindan da desteklenmektedir.)

Bunun icin "variance" komutu kullanilir.

Ornek;

```

Router(config)#router igrp 102
Router(config-router)#network 10.1.1.0
Router(config-router)#network 192.168.1.0
Router(config-router)#network 172.16.1.0
Router(config-router)#variance 2

```

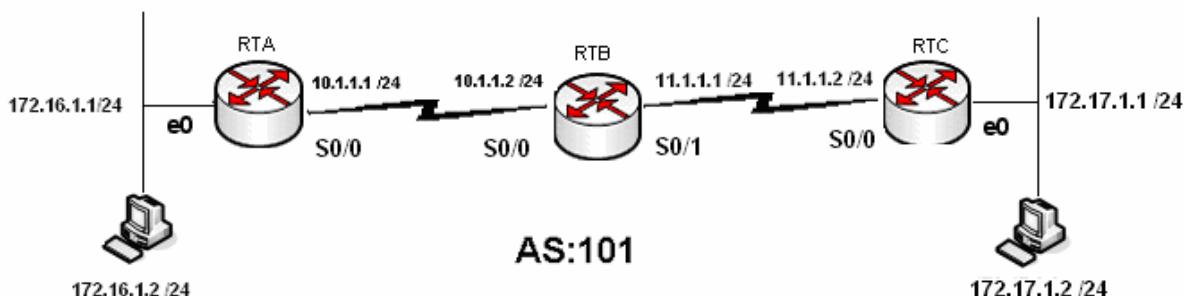
Burada Router variance ile belirttilmis sayisi alip en kucuk metric degeri ile carpar ve o degerin altinda metrige sahip yollar arasında load balancing yapar.

IGRP Konfigürasyonu

IGRP' de tipki Rip gibi classfull bir routing protokoldür.

IGRP konfigürasyonu Rip' in ki ile büyük ölçüde aynıdır. Burada tek fark aynı sistemde çalıştığımızı belirtmek için kullanacağımız Autonomous System numarasıdır. Kısaca AS denebilir. Bütün Routerlarda aynı AS kullanılmaz ise routerlar arasında iletişim olmaz.

Ornek bir senaryo ile konfigürasyonu yapmak gereklir;



Konfigürasyon yapılırken DCE ve DTE uçlar düzgün belirlenmelii ve gereken yerlere "clock rate" verilmelidir.

```
A(config)#router igrp 101
A(config-router)#net
A(config-router)#network 172.16.1.0
A(config-router)#network 10.1.1.0
A(config-router)#_
C(config)#router igrp 101
C(config-router)#net
C(config-router)#network 172.17.1.0
C(config-router)#network 11.1.1.0
C(config-router)#

```

(A ve C router' larının konfigürasyonu)

Routing Table incelendiğin administatice distinct' in 100 olduğu görünecektir. Yine dikkat edilirse metric değerlerinin Rip' inkinden çok farklı ve yüksek değerlerde olduğu gözden kaçmaz. İşte bu metrik değerleri bahsettiğimiz K1' den K5' e kadar değerlerle hesaplanmıştır.

```
A#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
```

Gateway of last resort is not set

```
I   172.17.0.0/16 [100/91056] via 10.1.1.2, 00:00:13, Serial0/0
    172.16.0.0/24 is subnetted, 1 subnets
C     172.16.1.0 is directly connected, Ethernet0/0
    10.0.0.0/24 is subnetted, 1 subnets
C     10.1.1.0 is directly connected, Serial0/0
I   11.0.0.0/8 [100/90956] via 10.1.1.2, 00:00:13, Serial0/0
A#
```

(A router'ı Yönlendirme Tablosu)

```
C#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route
```

Gateway of last resort is not set

```
    172.17.0.0/24 is subnetted, 1 subnets
C      172.17.1.0 is directly connected, Ethernet0/0
I   172.16.0.0/16 [100/10576] via 11.1.1.1, 00:01:13, Serial0/0
I   10.0.0.0/8 [100/10476] via 11.1.1.1, 00:01:13, Serial0/0
    11.0.0.0/24 is subnetted, 1 subnets
C      11.1.1.0 is directly connected, Serial0/0
C#
```

(C Router'ı yönlendirme tablosu)

```
B(config)#router igrp 101
B(config-router)#net
B(config-router)#network 10.1.1.0
B(config-router)#network 11.1.1.0
B(config-router)#
B(config-router)#_
```

00:02:09 bağlandı OtoAloala 9600 8-N-1 Kaydır büyük

B Router'ı Konfigürasyonu ve Yönleendirme Tablosu

```
B#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
       D - EIGRP, EX - EIGRP external, 0 - OSPF, IA - OSPF inter
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external t
       E1 - OSPF external type 1, E2 - OSPF external type 2, E -
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - ca
       U - per-user static route, o - ODR

Gateway of last resort is not set

I    172.17.0.0/16 [100/89056] via 11.1.1.2, 00:00:07, Serial3
I    172.16.0.0/16 [100/8576] via 10.1.1.1, 00:00:57, Serial1
      10.0.0.0/24 is subnetted, 1 subnets
C      10.1.1.0 is directly connected, Serial1
      11.0.0.0/24 is subnetted, 1 subnets
C      11.1.1.0 is directly connected, Serial3
B#
```

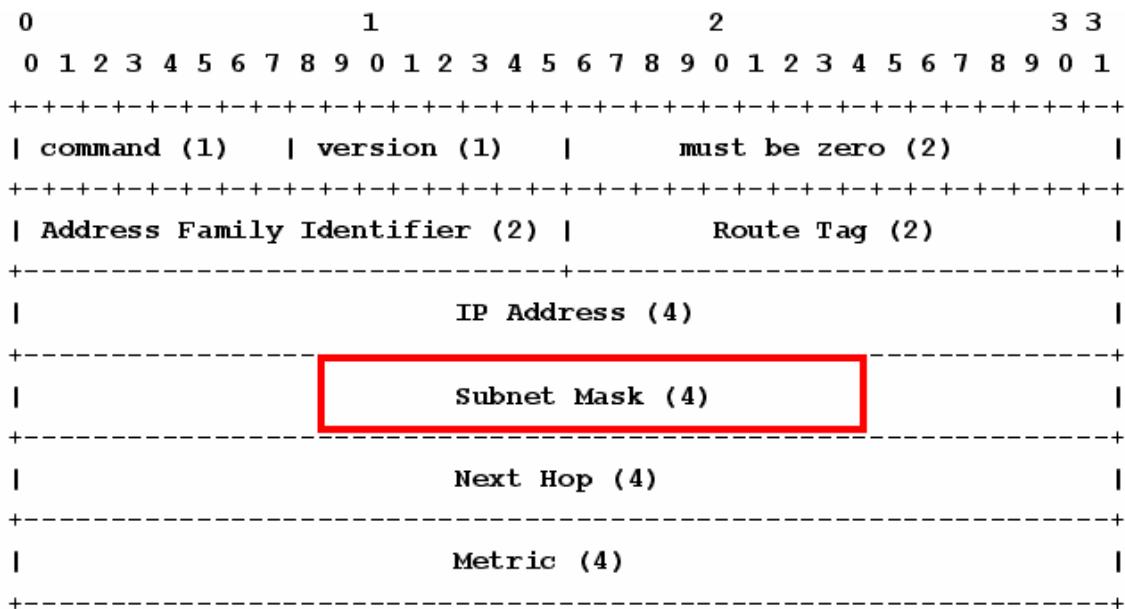
IGRP' de tipki Rip gibi updatelerini 255.255.255.255 broadcast adresinden yapar.

```
C#
C#debug ip igrp events
IGRP event debugging is on
C#
00:36:08: IGRP: received update from 11.1.1.1 on Serial0/0
00:36:08: IGRP: Update contains 0 interior, 2 system, and 0 exterior routes.
00:36:08: IGRP: Total routes in update: 2
00:36:27: IGRP: received update from invalid source 172.16.1.1 on Ethernet0/0
00:36:36: IGRP: sending update to 255.255.255.255 via Ethernet0/0 (172.17.1.1)
00:36:36: IGRP: Update contains 0 interior, 3 system, and 0 exterior routes.
00:36:36: IGRP: Total routes in update: 3
00:36:36: IGRP: sending update to 255.255.255.255 via Serial0/0 (11.1.1.2)
00:36:36: IGRP: Update contains 0 interior, 1 system, and 0 exterior routes.
00:36:36: IGRP: Total routes in update: 1
00:37:21: IGRP: received update from 11.1.1.1 on Serial0/0
00:37:21: IGRP: Update contains 0 interior, 2 system, and 0 exterior routes.
00:37:21: IGRP: Total routes in update: 2
```

RIPv2

Rip protokolünün classfull olması ve uygulamada sorunlar çıkarması sebebiyle geliştirilmiş ve Classless olan versiyonu çıkarılmıştır: Ripv2. Classless olmasının yanında bir önemli farkta Ripv2'nin updatelarını broadcast adresinden değil 224.0.0.9 multicast adresinden göndermesidir.

UDP 520 nolu portu kullanan RIP version 2 calistiginda update paketleri su sekildedir.



Burada ki subnet mask bilgisi protokolün Classless çalışmasını sağlar.

```
ISP#debug ip rip
RIP protocol debugging is on
ISP#01:23:34: RIP: received v2 update from 192.168.4.22 on Serial1
01:23:34:     172.30.100.0/24 -> 0.0.0.0 in 1 hops
01:23:34:     172.30.110.0/24 -> 0.0.0.0 in 1 hops
ISP#                                         Subnet Mask Bilgisi
01:23:38: RIP: received v2 update from 192.168.4.26 on Serial0
01:23:38:     172.30.2.0/24 -> 0.0.0.0 in 1 hops
01:23:38:     172.30.1.0/24 -> 0.0.0.0 in 1 hops
ISP#                                         multicast
01:24:31: RIP: sending v2 update to 224.0.0.9 via Ethernet0 (10.0.0.1)
01:24:31:     172.30.2.0/24 -> 0.0.0.0, metric 2, tag 0
01:24:31:     172.30.1.0/24 -> 0.0.0.0, metric 2, tag 0
01:24:31:     172.30.100.0/24 -> 0.0.0.0, metric 2, tag 0
01:24:31:     172.30.110.0/24 -> 0.0.0.0, metric 2, tag 0
01:24:31:     192.168.4.24/30 -> 0.0.0.0, metric 1, tag 0
01:24:31:     192.168.4.20/30 -> 0.0.0.0, metric 1, tag 0
```

Konfigürasyonda ise tek küçük fark "version 2" komutunun verilecek olmasıdır. Bu komut router konfigürasyon alt moduna qəcidiyəndə verilmelidir.

Rip protokolü sırasında üzerinde çalıştığımız senaryoyu burada da uyguladığımızda arada ki farklıları daha iyi anlayacağız.



Bu topoloji de ip adreslerimizi ilgili interface' lere atadıktan sonra konfigürasyon RIPv2 için şu şekilde olacaktır.

```

RouterA#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
RouterA(config)#router rip
RouterA(config-router)#ver
RouterA(config-router)#version 2
RouterA(config-router)#net
RouterA(config-router)#network 10.1.1.0
RouterA(config-router)#net
RouterA(config-router)#network 192.168.3.0
RouterA(config-router)#exit
RouterA(config)#exit
RouterA#
(Router A için Konfigürasyon)
  
```

```

RouterB#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
RouterB(config)#router rip
RouterB(config-router)#ver
RouterB(config-router)#version 2
RouterB(config-router)#net
RouterB(config-router)#network 10.1.1.0
RouterB(config-router)#net
RouterB(config-router)#network 192.168.4.0
RouterB(config-router)#exit
RouterB(config)#exit
(Router B için Konfigürasyon)
  
```

Her iki router için konfigürasyonlar tamamlandığında networkler arasında iletişim sağlanmış olacaktır. Bu iletişim tabi ki Router'ların Routing Table'larında bulunan bilgilere dayanarak olacaktır.

Routing Table'lar artık çok iyi bildiğiniz gibi "show ip route" komutu ile görüntülenebiliyor.

```

RouterA#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

R    192.168.4.0/24 [120/1] via 10.1.1.2, 00:00:18, Serial0/1
      10.0.0.0/30 is subnetted, 1 subnets
C      10.1.1.0 is directly connected, Serial0/1
C      192.168.3.0/24 is directly connected, Ethernet0/0
RouterA#

```

00:20:15 bağlandı OtoAlgıla 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı
(RouterA için Routing Table)

```

RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.4.0/24 is directly connected, Ethernet0/0
      10.0.0.0/30 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, Serial0/1
R    192.168.3.0/24 [120/1] via 10.1.1.1, 00:00:12, Serial0/1
RouterB#

```

00:24:43 bağlandı OtoAlgıla 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı
(RouterB için Routing Table)

Routign Table' lar dikkatle incelendiğinde uzak networklere giderken kullanılacak yollar metric ifadeleriyle birlikte görüntülenebiliyor. RIPv2' de tipki RIPv1 gibi metric hesabında hop sayısını kullandığı için buradaki metricler aynı zaman da hop sayısına eşittir.

```

RouterA#debug ip rip
RIP protocol debugging is on
RouterA#
00:28:32: RIP: sending v2 update to 224.0.0.9 via Ethernet0/0 (192.168.3.1)
00:28:32: RIP: build update entries
00:28:32:      10.0.0.0/8 via 0.0.0.0, metric 1, tag 0
00:28:32:      192.168.4.0/24 via 0.0.0.0, metric 2, tag 0
00:28:32: RIP: sending v2 update to 224.0.0.9 via Serial0/1 (10.1.1.1)
00:28:32: RIP: build update entries
00:28:32:      192.168.3.0/24 via 0.0.0.0, metric 1, tag 0
00:28:37: RIP: ignored v2 update from bad source 192.168.4.1 on Ethernet0/0
00:28:37: RIP: received v2 update from 10.1.1.2 on Serial0/1
00:28:37:      192.168.4.0/24 via 0.0.0.0 in 1 hops
00:28:59: RIP: sending v2 update to 224.0.0.9 via Ethernet0/0 (192.168.3.1)
00:28:59: RIP: build update entries
00:28:59:      10.0.0.0/8 via 0.0.0.0, metric 1, tag 0
00:28:59:      192.168.4.0/24 via 0.0.0.0, metric 2, tag 0
00:28:59: RIP: sending v2 update to 224.0.0.9 via Serial0/1 (10.1.1.1)
00:28:59: RIP: build update entries
00:28:59:      192.168.3.0/24 via 0.0.0.0, metric 1, tag 0_

```

00:23:57 bağlandı OtoAlgıla 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı
(RouterA için debug)

Updateler multicast 224.0.0.9 adresinden gönderiliyor ve alınıyor. Oysa Rip version 1' de updateler broadcast 255.255.255.255 adresinden yapılmıştı.

İsterseniz şimdi tekrar Rip version 1' e geçip, bir de oradaki updateleri inceleyelim. Geçiş her iki Router'da da "version 2" ifadesini kaldırarak yapılabilir. Her zaman olduğu gibi kaldırmak istediğimiz bir komut olduğunda başına "no" yazmamız yeterli olacaktır. Örneğin A Router'ı Rip version 1' e şu şekilde geçer:

```
RouterA(config)#router rip
RouterA(config-router)#no ver
RouterA(config-router)#no version 2
RouterA(config-router)#_
```

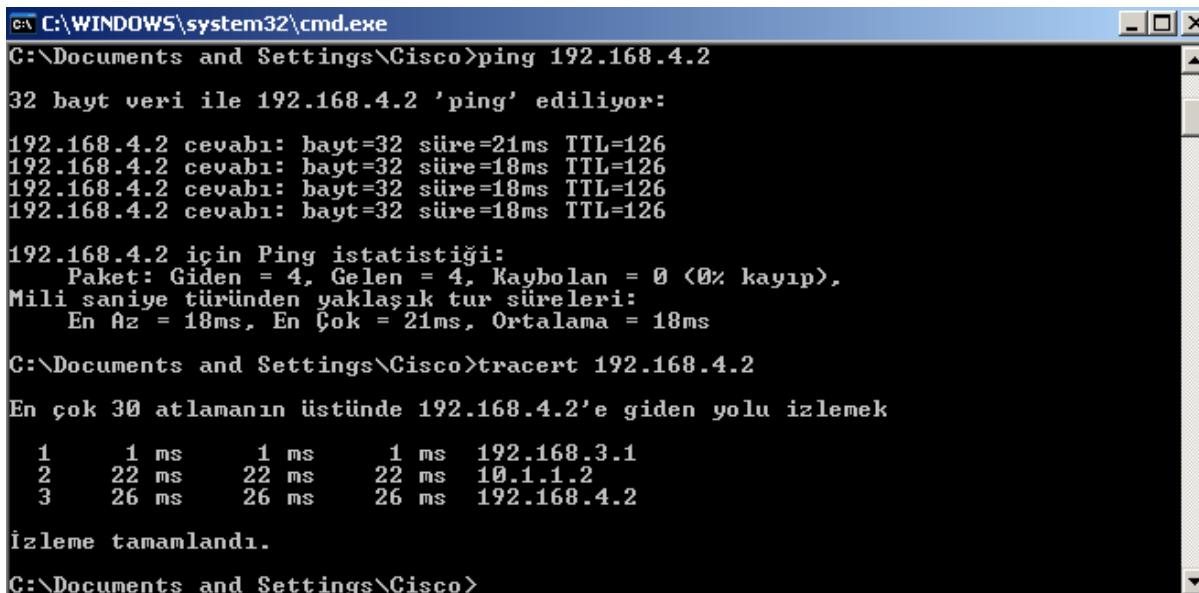
00:25:05 bağlandı OtoAlqila 9600 8-N-1 Kaydır büyh SAYI Yak

```
RouterA#debug ip rip
RIP protocol debugging is on
RouterA#
00:32:39: RIP: sending v1 update to 255.255.255.255 via Ethernet0/0 (192.168.3.1)
)
00:32:39: RIP: build update entries
00:32:39:      network 10.0.0.0 metric 1
00:32:39:      network 192.168.4.0 metric 2
00:32:39: RIP: sending v1 update to 255.255.255.255 via Serial0/1 (10.1.1.1)
00:32:39: RIP: build update entries
00:32:39:      network 192.168.3.0 metric 1
```

00:27:40 bağlandı OtoAlqila 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma vankısı
(RouterA için debug)

Her iki versiyonun update'leri arasında ki fark artık daha iyi anlaşılmıştır.

Networkler arasından ki iletişim komut satırında kullanabileceğimiz "tracert" komutu ile inceleyebiliriz.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Cisco>ping 192.168.4.2
32 bayt veri ile 192.168.4.2 'ping' ediliyor:
192.168.4.2 cevabı: bayt=32 süre=21ms TTL=126
192.168.4.2 cevabı: bayt=32 süre=18ms TTL=126
192.168.4.2 cevabı: bayt=32 süre=18ms TTL=126
192.168.4.2 cevabı: bayt=32 süre=18ms TTL=126

192.168.4.2 için Ping istatistiği:
  Paket: Giden = 4, Gelen = 4, Kaybolan = 0 (0% kayıp),
Mili saniye türünden yaklaşık tur süreleri:
  En Az = 18ms, En Çok = 21ms, Ortalama = 18ms

C:\Documents and Settings\Cisco>tracert 192.168.4.2
En çok 30 atlamanın üstünde 192.168.4.2'e giden yolu izlemek
  1      1 ms      1 ms      1 ms  192.168.3.1
  2     22 ms     22 ms     22 ms  10.1.1.2
  3     26 ms     26 ms     26 ms  192.168.4.2

İzleme tamamlandı.

C:\Documents and Settings\Cisco>
```

Ben bu komutu kullanırken 192.168.3.2 ip adresine sahip bilgisayarı kullandım. 1 adımda ping paketim varsayılan ağ geçidi olarak tanımladığım RouterA'ının Ethernet interface'ine, ikinci adım da bir sonraki Router'ın Serial interface'ine ve üçüncü adımda da hedefe ulaştı.

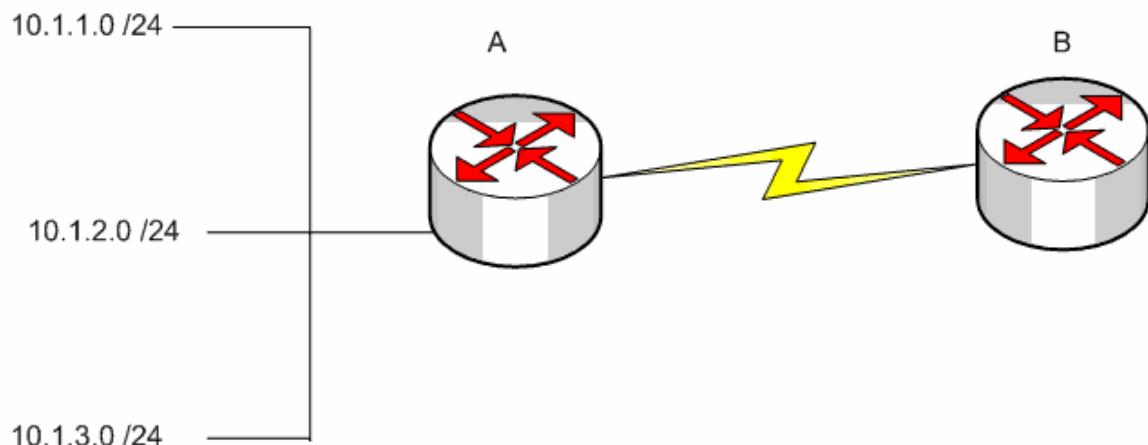
Her iki Router için Running-config dosyasının incelenmesi fayda sağlayacaktır.

```
RouterA#show running-config
Building configuration...
Current configuration : 567 bytes
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterA
!
!
!
memory-size iomem 10
ip subnet-zero
!
!
!
interface Ethernet0/0
 ip address 192.168.3.1 255.255.255.0
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
!
interface Serial0/1
 ip address 10.1.1.1 255.255.255.252
!
router rip
 version 2
 network 10.0.0.0
 network 192.168.3.0
!
ip classless
ip http server
!
!
line con 0
line aux 0
line vty 0 4
!
no scheduler allocate
end
```

```
RouterB#show running-config
Building configuration...
Current configuration : 578 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterB
!
memory-size iomem 10
ip subnet-zero
!
interface Ethernet0/0
 ip address 192.168.4.1 255.255.255.0
 half-duplex
!
interface Serial0/0
 no ip address
 shutdown
!
interface Serial0/1
 ip address 10.1.1.2 255.255.255.252
 clockrate 64000
!
!
version 2
network 10.0.0.0
network 192.168.4.0
!
ip classless
ip http server
!
!
gatekeeper
 shutdown
!
!
line con 0
line aux 0
line vty 0 4
!
end
```

RipV2 Auto Summary

Ripv2' nin Auto Summarization ozelligi vardır ve default olarak açık durumdadır.



Ornegin sekildeki yapı içerisinde Ripv2 ile konfigüre edilmiş A routeri yine Ripv2 ile konfigüre edilmiş B routerına 10.0.0.0 networkunu update edecektir.

Bu çalışma mantığı içerisinde default olarak açık olan auto summarization ozelligi kapatılmadığı takdir de Ripv2 ninde sanki Classfull mus gibi çalıstığı söylenilir.

Auto Summarization ozelligi "no auto-summary" komutu ile kaldırılabilir.

```
A(config)#router rip
A(config-router)#network 10.1.1.0
A(config-router)#network 10.1.2.0
A(config-router)#network 10.1.3.0
A(config-router)#version 2
A(config-router)#no auto summary
```

Konfigurasyonun bu hali ile artık A routeri summary update yerine butun networkleri update edecektir ve B Routeri Routing Table' inde butun networkler yer alacaktır.

Extra

Split Horizon kuralının enable olmadığı durumlarda ripv2 ile konfigüre edilmiş ve update edilecek networklerin üzerine yazacak ve interface' e uygulanacak "ip summary-address" komutu kullanılabilir.

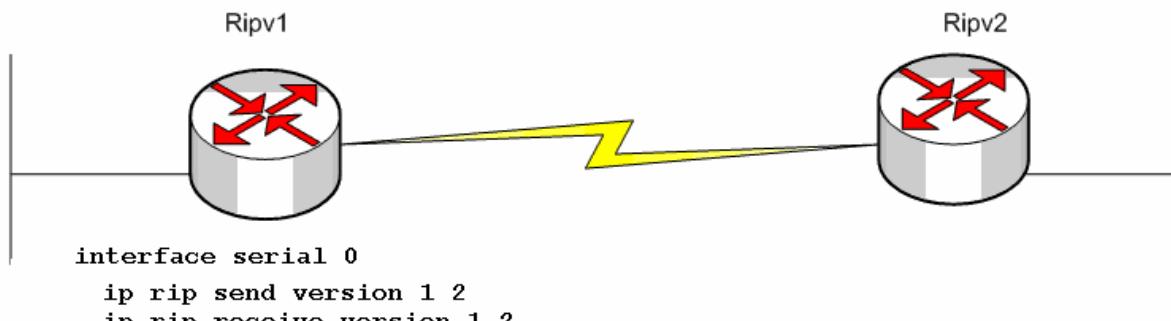
```
int s1
ip address 10.1.1.1 255.255.255.0
ip summary-address rip 10.2.0.0 255.255.0.0
no ip split-horizon
```

```
router rip
network 10.0.0.0
```

Ornegin bu uygulamada 10.2.0.0 update' i rip tarafından ozetlenen 10.0.0.0 update' inin üzerine yazacaktır.

Ripv1 ve Ripv2 Haberlesmesi

Ripv1 ve Ripv2 konfigürasyonlar sistemde bulunan routerlar arasında sadece Ripv1, Ripv2 paketlerini alıp göndermek yada her ikisinde alıp göndermek üzere konfigür edilebilir.

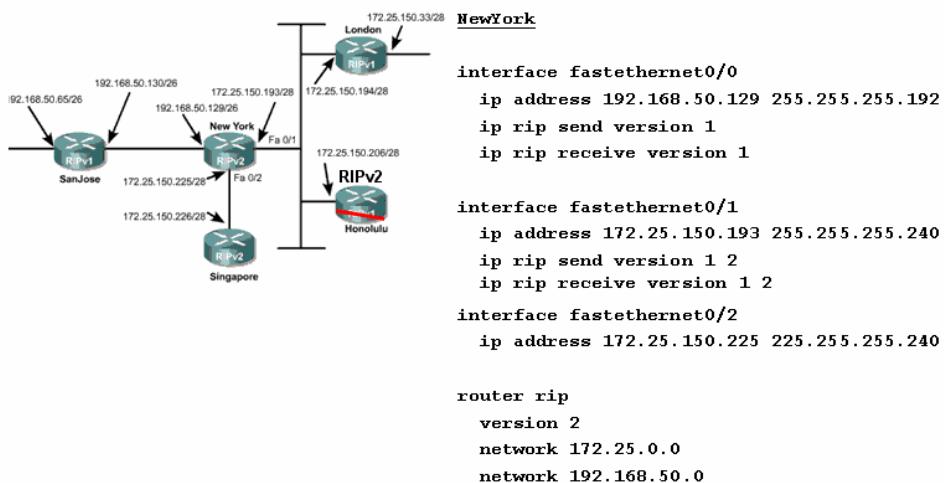


Burada istenirse farklı uygulamalar yapılabilir.

Ornegin;

`ip rip receive version 1`

komut satırı ile söz konusu Routerin sadece Version 1 updatelerini alması sağlanabilir.



CNAP Slaytlarından alınan bu şekilde durum daha iyi anlaşılmaktır. Burada Newyork Routeri RIPv2 ile konfigür edilmiş. Ve interfacelerine sırasıyla su şekilde konfigür edilmiş diğer routerlar bağlı;

Fa0/0: RIPv1

Fa0/2: RIPv2

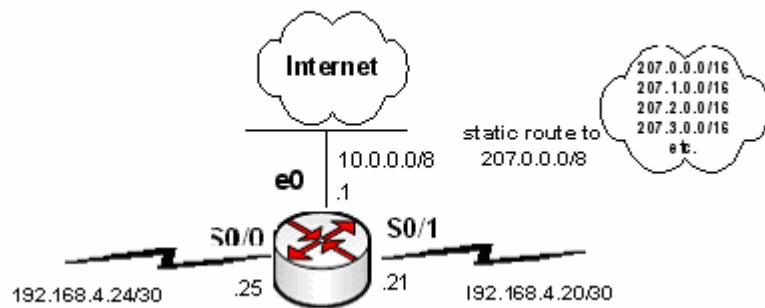
Fa0/1: RIPv1 ve RIPv2

Bu durumda Fa0/0 interface ine hem London Routerinden aldığı v1 updatelerini gondermeli hemde ondan v1 updatelerini almalıdır.

Fa0/1 interface inde ise konfigurasyona bakildiginda hem v1 hem de v2 updateelerini gondermek üzere hem de almak üzere konigure edildigi anlasilmistir. Cunki bu interface' e hem RIPv1 ile hem de RIPv2 ile konfigure edilmiş Routerlar baglanmistir. (Multiaccess)

Fa0/2 icin zaten ozel bir konfigurasyona gerek yoktur.

Ripv2 ve Default Routing



ISP Routeri üzerinde Default Route tanimlasınması aşağıdaki gibi olacaktır.

```
ISP
router rip
redistribute static
network 10.0.0.0
network 192.168.4.0
version 2
no auto-summary
default-information originate

ip route 207.0.0.0 255.0.0.0 null0
ip route 0.0.0.0 0.0.0.0 10.0.0.2
etherenet0
```

Ripv2 Authentication

Ripv2 konusan Routerların update'leri sırasında authentication sağlanabilir. Bunun için Global Konfigurasyon modunda "key" komutu kullanılmalıdır.

```
Router(config)#key chain Ozcan  
Router(config-keychain)#key 1  
Router(config-keychain-key)#key-string Yildiz
```

Authentication sağlanacak Routerlar için password aynı olmalıdır ancak key adı değiştirilebilir. Key oluşturulduktan sonra interface'ye uygulanmalıdır.

```
Router(config)#interface fastethernet 0/0  
Router(config-if)#ip rip authentication key-chain Ozcan  
Router(config-if)#ip rip authentication mode md5
```

Burada ki "ip rip authentication mode md5" komutunun kullanımı opsyoneldir. Authentication bilgilerinin encrypted halde gönderilmesini sağlayan bu komut kullanılmadığında da interface default olarak text halinde authentication bilgilerini gönderecektir.

Access-Lists (Erişim Listeleri)

Access list'ler sistem yöneticilerine, ağdaki trafik üzerinde geniş bir kontrol imkanı sunar. Ayrıca access list'ler router üzerinden geçen paketlere izin vermek veya reddetmek içinde kullanılır. Bunun haricinde telnet erişimleri de access list'ler kullanılarak düzenlenebilir. Oluşturulan access list'ler router'daki interface'lerin herhangi birisine giren veya çıkış trafiği kontrol edecek şekilde uygulanabilir. Eğer herhangi bir interface'e bir access list atanmışsa router bu interface'den gelen her paketi alıp inceleyecektir ve access list'te belirtilen işlevi yerine getirecektir. Yani ya o paketi uygun yöne ileticek ya da paketi yönlendirmeden yok edecektir.

- Access List'lerde kriterler satır satır belirtilmiştir. Gelen isteklerin kriterlere uyup uymadıkları sırayla belirlenir.
- İlk eşleşen kriterin bulunduğu satırda o satırda ki aksiyon (deny yada permit) gerçekleştirilir.
- Paket bütün satırları geçmiş ve herhangi bir kriterle eşleşme olmamışsa "bütün paketleri yoket" (implicit deny all) kuralı uygulanır.

Access List'ler 3 Başlık altında incelenirler:

1. Standart ACL
2. Extended ACL
3. Named Acl

3 başlık dememize rağmen aslında iki başlık gibi düşünülmelidir. Çünkü Named Accces Listler hem standart hem de Extended olarak kullanılabilirler.

Access Listler arasından ki bu ayırım Acces List Numaraları ile yapılır. Access Listler şu numaraları alabilirler;

Access List Numarası	Açıklama
1-99 arası	IP standart access list
100-199 arası	IP extended access list
1000-1099 arası	IPX SAP access list
1100-1199 arası	Extended 48-bit MAC address access list
1200-1299 arası	IPX summary address access list
200-299 arası	Protocol type-code access list
300-399 arası	DECnet access list
400-499 arası	XNS standart access list
500-599 arası	XNS extended access list
600-699 arası	Appletalk access list
700-799 arası	48-bit MAC address access list
800-899 arası	IPX standart access list
900-999 arası	IPX extended access list

Access List'ler oluşturulurken dikkat edilmesi gerekenler şunlardır;

- Oluşturulduktan sonra mutlaka bir interface ile ilişkilendirilmelidir aksi takdir de aktif olmayacağındır.
- Kriterler satır satır uygulanacağı için listeler oluşturulurken en belirgin kriterden en genel kriterde doğru yukarıdan başlayarak organize edilmelidir.
- Listededen satır silmek ve satır eklemek sadece Named ACL'lerde mümkündür. Diğer listelerde silme işleminde satır değil listenin tamamı silinir. Bu durumda araya satır eklenmek isteniyorsa liste bir yazı editörüne aktarılıp değişiklik orada yapılmalıdır.

- Standart Access Listler mümkün olduğu kadar hedefe, Extended Access Listler mümkün olduğu kadar kaynağa yakın olmalıdır.
- Access Listlerin en sonudan görünmeyen bir satır olduğunu ve bu satırında diğer satırlardaki herhangi bir kritere uymayan istekleri yok ettiğini söylemişтик. Dolayısıyla mutlaka ve mutlaka bir Access List grubunda "permit" aksiyonu olmalıdır.
- Access Listler sadece Router üzerinden giden veya gelen trafiği düzünləmek için kullanılır. Router'ın sebep olduğu trafik için kullanılamazlar.
- Access Listler' den satır çıkaramasınız ve satır eklediğinizde de o satır en son satır olarak yerini alır. Dolayısıyla kriterlerini yeniden düzenlemek bu şekilde imkansızdır. (Named Access List'ler hariç) Bu durumda yapılması gereken Access List' i bir text editörüne kopyalayıp gerekli değişiklikleri yaptıktan sonra ger kopyalamaktır.

Access Listler oluşturulurken Subnet Mask yerine Wild Card Mask denilen ve Subnet Maskın 255'e tamamlanmasıyla elde edilen bir maske kullanılır. Örneğin 255.255.128.0 subnet maskının wild-card maskı 0.0.127.255 olacaktır.

Tek bir host belirtmek için kullanılacak;

Ip adresi: 192.168.1.2

Wild-Card Mask: 0.0.0.0

Standart Access-Listler

Bu tür access list'te IP paketlerinin sadece kaynak (source) adreslerine bakılarak filtreleme yapılır. Izin verme ya da yasaklama bütün protokol kümlesi için geçerlidir.

```
Router(config)#access-list {Access list numarası} {permit / deny} {kaynak} {mask}
```

Şeklinde kullanılır. Burada ki "permit" izin vermek için, "deny" yasaklamak için kullanılır.

Daha sonra uygulanacak olan interface' gidilerek

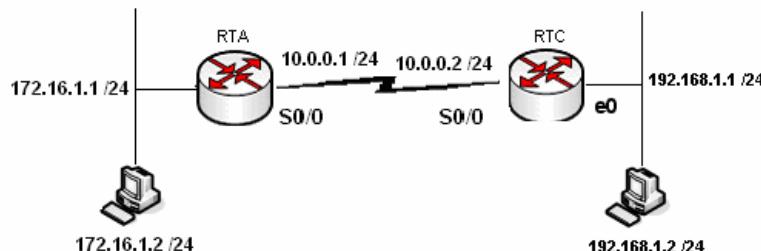
"ip Access-group {numarası} in/out" komutuyla interface ile ilişkilendirilir. Burada ki in ve out komutlara isteğe göre içерiden dışarıya (in) ve dışarıdan içeriye (out) olan trafiği kısıtlamak için kullanılır.

Örneğin networkümüz de bulunan 192.168.1.100 ip adresine sahip bilgisayarın dışarıya çıkışını önlemek istersek komut satırında;

```
Router(config)#access-list 1 deny 192.168.1.100 0.0.0.0
Router(config)#access-list 1 permit any
Router(config)#interface Ethernet 0/0
Router(config-if)#ip Access-group 1 in
```

Yazmalıyız. Burada 1. satırda ilgili hosta "deny" uygulandı, 2.satırda diğer hostların "implicit deny all" kuralı ile yok edilmemeleri için kalan hostlara "permit uygulandı, 3 ve 4.satırlarda ise oluşturulan Access list Ethernet interface' ile ilişkilendirildi. "

Access listlerde "{ip adresi wild-card mask}" yerine "host {ip adresi}" kullanılabilir. Fakat networklere bir aksiyon uygulanacaksa Wild-Card Mask kullanılmalıdır.



Örnek senaryomuz da A Router'ının Ethernet interface'ine bağlı 172.16.1.0 networkünde yer alan 172.16.1.2 bilgisayarının Ethernet interface2inden dışarı çıkışmasını engelleyelim fakat diğer bilgisayarlar bundan hiçbir şekilde etkilenmesin.

Bu durumda A Router'ın şu konfigürasyon yapılmalıdır;

```
A(config)#access-list 1 deny 172.16.1.2 0.0.0.0
A(config)#access-list 1 permit any
A(config)#int
A(config)#interface ethernet 0/0
A(config-if)#ip access-group 1 in
A(config-if)#
```

1:05:10 başlandı | OtoAçılı | 9600 8-N-1 | Kaydır | büyüh | SAYI | Yakala | Yazdırma ve

Bu konfigürasyon yapıldığı andan itibaran 192.168.1.2 bilgisayarı sadece kendi LAN'ı ile haberleşebilecek, Router üzerinden kesinlikle dışarıya çıkamayacaktır. İkinci satırda yer alan "Access-list 1 permit any" satırı ile diğer bilgisayarların bu kısıtlamadan etkilenmesi engellenmiş oldu. Konfigürasyon sırasında 2. satır yazılmamış olsaydı gelen paketler / istekler tamamen yok edilecekti.

Extended Access Listler

Bu tür access listler de kaynak ile birlikte kullanılan protokol, hedef ip adresi ve hedef port numarası da kısıtlanabilir.

Örneğin 192.168.1.100 bilgisayarının 212.1.1.8 bilgisayarına 80. porttan erişememini, aynı bilgisayara 25. porttan erişebilmesini, diğer bilgisayarlar için herhangi bir kısıtlama olmamasını istiyoruz. (Söz konusu portlar TCP çalışır) Bu durumda komut satırına;

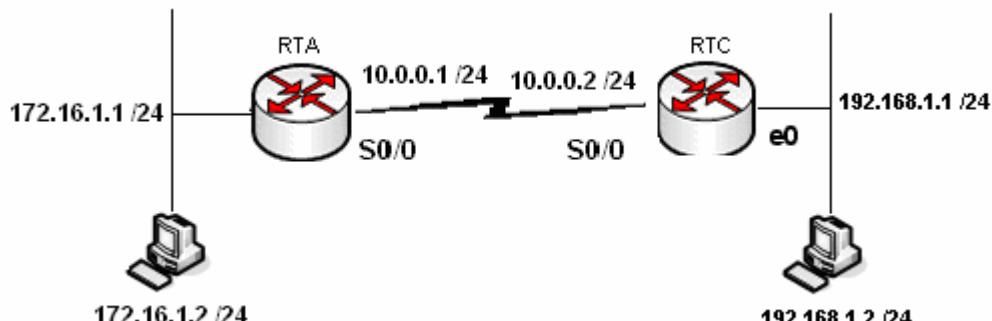
```
Router(config)#access-list 101 deny tcp host 192.168.1.100 host 212.1.1.8 eq 80
Router(config)#access-list 101 permit tcp host 192.168.1.100 host 212.1.1.8 eq 25
Router(config)#access-list 101 permit ip any any
```

Yazmak ve gerekli interface' e uygulamak yeterli olacaktır. Burada 1 ve ikinci satırlarda 192.168.1.100 ip adresine sahip bilgisayarın 212.1.1.8 ip adresine sahip uzak bilgisayara, 80. porttan erişememini fakat 25. porttan erişmesini sağlamış oluyoruz. 3. satır ile de diğer bilgisayarların "implicit deny all" kuralı ile yok edilmelerini önlemış olduk.

Yine burada host 192.168.1.100 yerine Wild-Card Mask kullanarak "192.168.1.100 0.0.0.0" yazabilirdik.

Bir senarayo üzerinde çalışmak gereklidir;

Elimizde şekildeki gibi birbirlerine bağlanmış 2 farklı network var. A Router' imizin Ethernet interface' ine bağlı networkte bulunan 172.16.1.2 bilgisayarı üzerinde bazı kısıtlamalar yapmak istiyoruz;



1. 172.16.1.2 bilgisayarı 172.17.1.2 bilgisayarına 3389. porttan erişemesin.
2. 172.16.1.2 bilgisayarı 172.17.1.2 bilgisayarına 80. porttan erişebilsin.
3. 172.16.1.2 bilgisayarı 172.17.1.2 bilgisayarına 80. porttan erişebilsin.
4. 172.16.1.0 networkündeki diğer bilgisayarlar uzak networkteki diğer bilgisayarlara istedikleri porttan erişebilsinler.

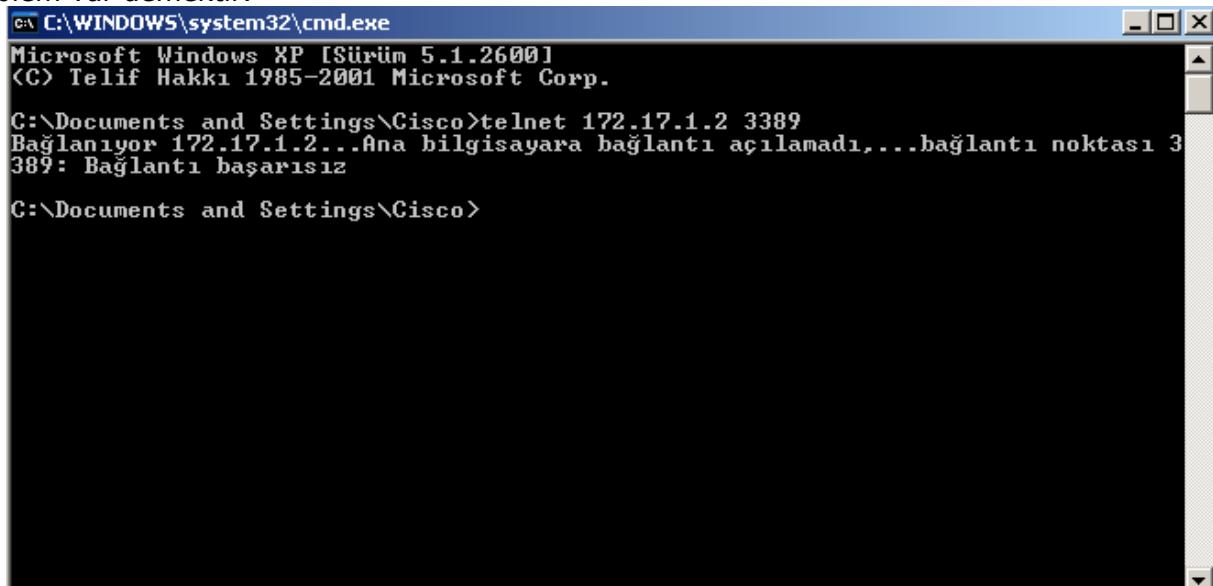
Böyle bir durumda A Router' i üzerinde yapılacak konfigürasyon şu şekilde yapılmalıdır:

```
A(config)#
A(config)#access-list 101 deny tcp host 172.16.1.2 host 172.17.1.2 eq 3389
A(config)#access-list 101 permit tcp host 172.16.1.2 host 172.17.1.2 eq 80
A(config)#access-list 101 permit tcp host 172.16.1.2 host 172.17.1.2 eq 25
A(config)#access-list 101 permit ip any any
A(config)#interface et
A(config)#interface ethernet 0/0
A(config-if)#ip access-group 101 in
A(config-if)#

```

Senaryo için belirlediğimiz istekleri satır satır konfigüre ettik. Access List' in 4. satırındaki komut ile kalan bilgisayarların çıkışmasına izin verilirken 1,2 ve 3. satırlarda 172.16.1.2 bilgisayarının uzak networkte ki 172.17.1.2 bilgisayarına doğru olan trafiğinde çeşitli kısıtlamalar ve izinler uygulandı. Devam eden satırlar da ise oluşturduğumuz Acces List ilgili interface' imizle eşleştirildi.

Yaptığımız düzenlemelerin düzgün çalışıp çalışmadığını test etmek isteyebiliriz. Bu durumda bize Telnet yardımcı olacaktır. Telnet ile uzak bilgisayara yasaklanan bir port üzerinde erişmek istediğiniz de Bağlantının başarısız olduğuna dair bir satır karşımıza gelecek, izin verilen bir porttan erişmek istediğimizde tamamen boş bir sayfa anında açılacaktır. Eğer anlatıldığı gibi durumlar ile karşılaşılmamışsa Access List' lerin oluşturulması ya da uygulanmasıyla ilgili bir problem var demektir.

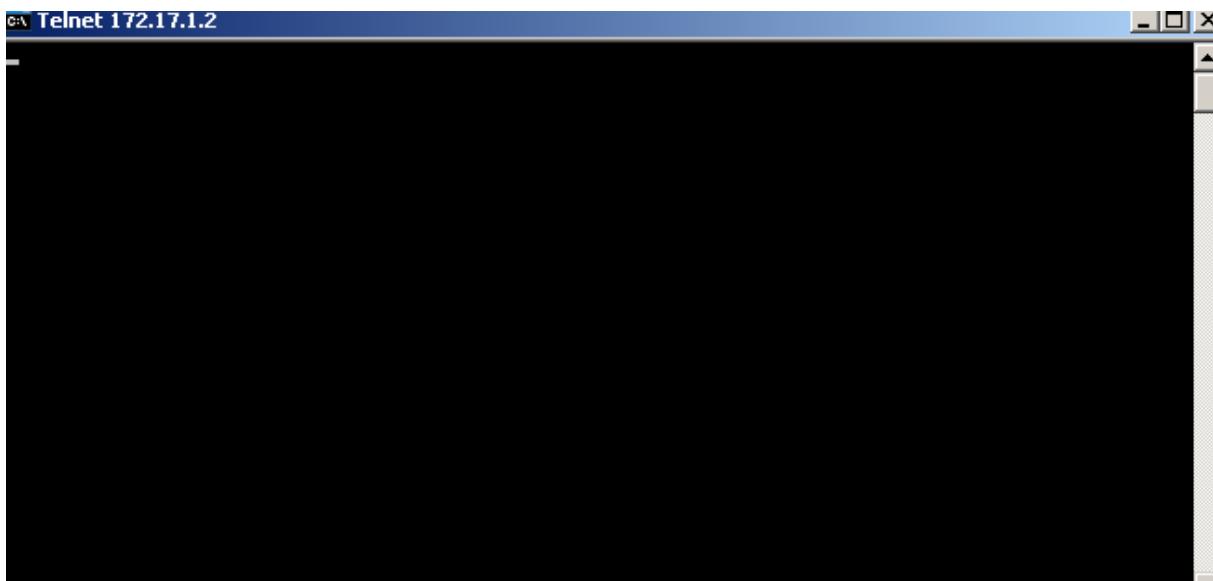


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Sürüm 5.1.2600]
(C) Telif Hakkı 1985-2001 Microsoft Corp.

C:\Documents and Settings\Cisco>telnet 172.17.1.2 3389
Bağlanıyor 172.17.1.2...Ana bilgisayara bağlantı açılamadı,...bağlantı noktası 3
389: Bağlantı başarısız

C:\Documents and Settings\Cisco>
```

(172.17.1.2 uzak bilgisayara 3389. porttan bağlanılamıyor.)



```
c:\ Telnet 172.17.1.2
-
```

(172.17.1.2 bilgisayara 80. porttan Telnet ile bağlanılması)

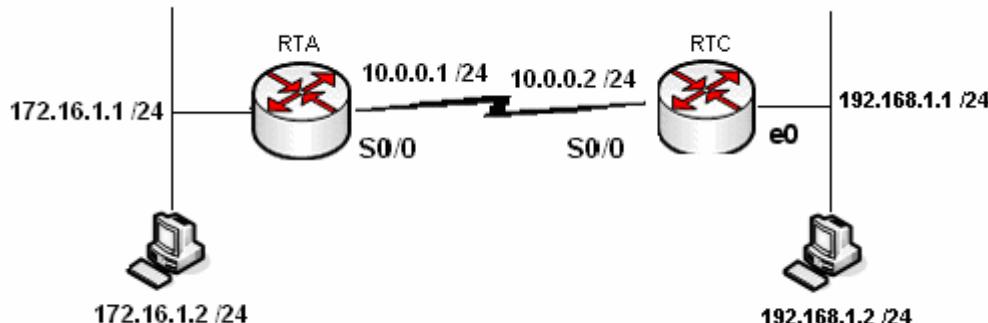
Named Access Listler

Diğer Access Listlerden sadece konfigürasyon sırasında farklılık gösterir. Named Access listler acces-list numarası vermek yerine akılda kalması da kolay olacak, isimler kullanılabilir. Named Access List'lerde satırlar tek tek silinebilir veya yeni satır eklenebilir. Çünkü listenin Standart ve Extended olmasına göre uygun modlar oluşturulur ve konfigürasyon bu modlar altında yapılır.

Extended Access List' te üzerinde çalıştığımız aynı senaryoyu Named Access List ile konfigüre etmek istersek komut satırına;

```
Router(config)# ip access-list extended AcademyTech
Router(config-ext-nacl)# deny tcp host 192.168.1.100 host 212.1.1.8 eq 80
Router(config-ext-nacl)# permit tcp host 192.168.1.100 host 212.1.1.8 eq 25
Router(config-ext-nacl)# permit tcp any any
```

Yazmamız yeterli olacaktır. Burada 1.satırda belirtilen AcademyTech bizim belirleyeceğimiz bir ismidir ve Access list'lerin standart numaraları yerine kullanılır. Bu konfigürasyonda hatalı bir satır yazıldığında başına "no" yazılarak satır iptal edilebilir.



Böyle bir senaryo da 172.16.1.2 bilgisayarının uzak networkteki 172.17.1.2 bilgisayarının 80 ve 25. portlardan erişmemesini, 3389. porttan erişebilmesini, diğer bilgisayarlar için herhangi bir kısıtlama olmamasını Named Access List ile yapmak istediğimizde konfigürasyon şu şekilde tanımlanmalıdır;

```
A(config)#ip access-list extended AcademyTech
A(config-ext-nacl)#deny tcp host 172.16.1.2 host 172.17.1.2 eq 80
A(config-ext-nacl)#deny tcp host 172.16.1.2 host 172.17.1.2 eq 25
A(config-ext-nacl)#permit tcp host 172.16.1.2 host 172.17.1.2 eq 3389
A(config-ext-nacl)#permit ip any any
A(config-ext-nacl)#exit
A(config)#interface ethernet 0/0
A(config-if)#ip access-group AcademyTech in
A(config-if)#
A(config-if)#

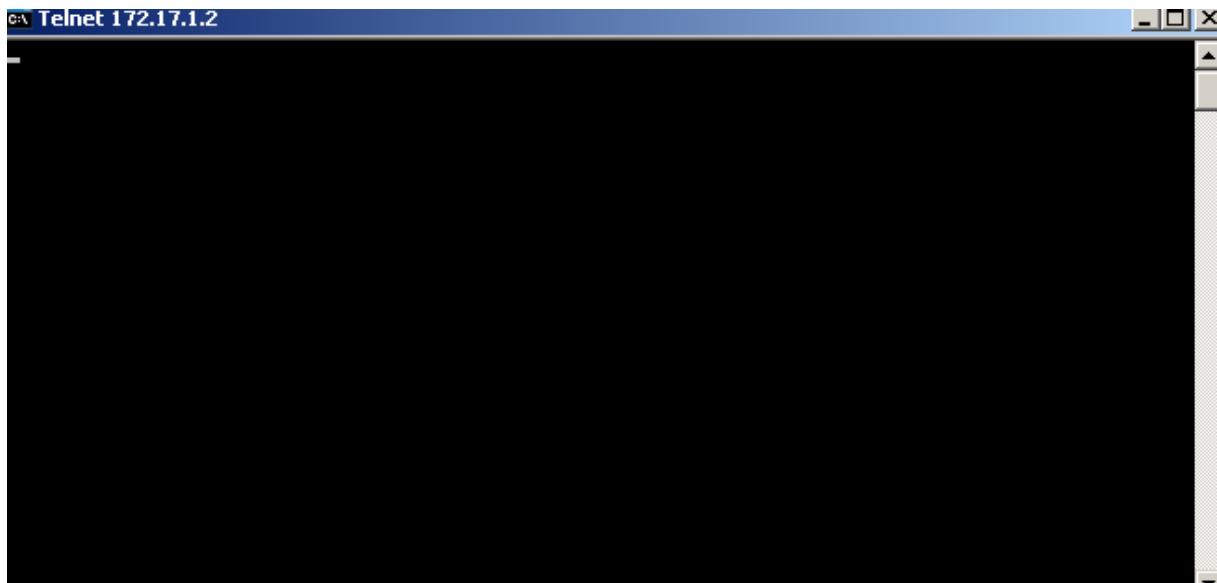
```

Named Access List

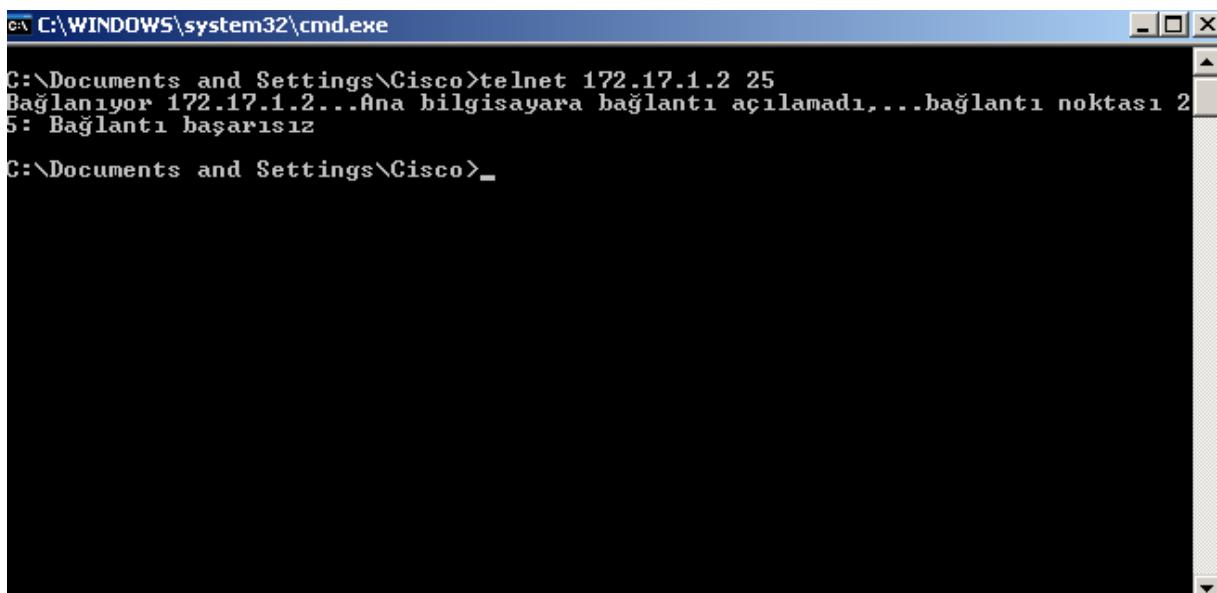
Burada Access List' in 1. satırında extenden Acces List kullanılacağı ve bu Access List' in isminin AcademyTech olacağı belirtildi, 2. ve 3. satırları ile uzak networkteki 172.17.1.2 bilgisayarına 80 ve 25. portlardan erişilmesi, 172.16.1.2 bilgisayıarı için yasaklanmış oldu.

4 ve 5. satırlarda ise gerekli izinler verildi. 4. satırdaki komut örnek olması için komut satırına yerleştirildi. Normal şartlarda bu satır kullanılmayabilir, çünkü 5. satırda ki ifade ile zaten 172.16.1.2 bilgisayıarı da diğer izinleri elde ediyor.

Telnet ile Acces List'leri test edersek;



(172.16.1.2'den 172.17.1.2'ye Telnet ile 3389. porttan bağlanma)



(25. porttan bağlanma denemesi başarısız.)

ACL Uygulamaları -1

Host B' den cikan paketlerin 192.168.3.0 networkune erismesini engellemek.

CCNA Routerinda;

```

CCNA(config)#access-list 1 deny 192.168.5.2 0.0.0.0
CCNA(config)#access-list 1 permit any
CCNA(config)#inter serial 0
CCNA(config-if)#ip access-group 1 in
  
```

Veya;

CCNP Routerinda;

```

CCNP(config)#access-list 1 deny 192.168.5.2 0.0.0.0
CCNP(config)#access-list 1 permit any
CCNP(config)#inter ethernet 0
CCNP(config-if)#ip access-group 1 in
  
```

ACL Uygulamaları -2

192.168.5.0 networkunun tamaminin 192.168.3.0 networkune erismesini engellemek.

CCNA Routerinda;

```

CCNA(config)#access-list 1 deny 192.168.5.2 0.0.0.255
CCNA(config)#access-list 1 permit any
CCNA(config)#inter serial 0
CCNA(config-if)#ip access-group 1 in

```

Veya;

CCNP Routerinda;

```

CCNP(config)#access-list 1 deny 192.168.5.2 0.0.0.255
CCNP(config)#access-list 1 permit any
CCNP(config)#inter ethernet 0
CCNP(config-if)#ip access-group 1 in

```

ACL Uygulamaları -3

HostA da bulunan FTP Server ve Web Server'a 192.168.5.2 bilgisayarının erişmesini engellemek. (Kalan trafik akışı normal devam etmeli)

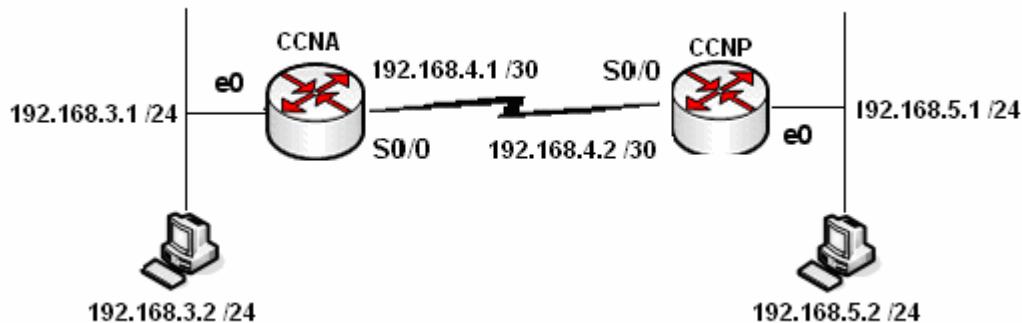
```

CCNA(config)#access-list 101 deny tcp host 192.168.5.2 host 192.168.3.2 eq 80
CCNA(config)#access-list 101 deny tcp host 192.168.5.2 host 192.168.3.2 eq 21
CCNA(config)#access-list 101 permit ip any any
CCNA(config)#
CCNA(config)#inter serial 0
CCNA(config-if)#ip access-group 101 in
CCNA(config-if)#
  
```

Veya;

```

CCNP(config)#access-list 101 deny tcp host 192.168.5.2 host 192.168.3.2 eq 80
CCNP(config)#access-list 101 deny tcp host 192.168.5.2 host 192.168.3.2 eq 21
CCNP(config)#access-list 101 permit ip any any
CCNP(config)#
CCNP(config)#inter ethernet 0
CCNP(config-if)#ip access-group 101 in
CCNP(config-if)#
  
```

ACL Uygulamalari -4

**192.168.5.0 networkunun 192.168.3.0 networkune ping atmasini yasaklamak,
192.168.3.0 networkunde 192.168.3.1 disindaki telnet isteklerini yasaklamak. (Kalan
trafig akisi devam etmeli)**

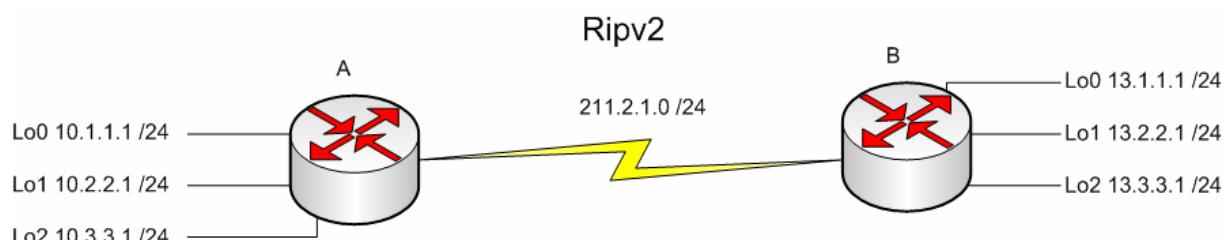
```
CCNA(config)#ip access-list extended Yildiz
CCNA(config-ext-nacl)#deny icmp 192.168.5.0 0.0.0.255 any echo
CCNA(config-ext-nacl)#permit tcp 192.168.5.0 0.0.0.255 host 192.168.3.1 eq telnet
CCNA(config-ext-nacl)#deny tcp 192.168.5.0 0.0.0.255 any eq telnet
CCNA(config-ext-nacl)#permit ip any any
CCNA(config)#inter serial 0
CCNA(config-if)#ip access-group Yildiz in
CCNA(config-if)#

```

Access Lists ve Distribute List

Routing protokoller ile calisirken bazi networklerin update edilmemesini isteyebiliriz. Bunun icin passive interface komutu bir cozumdur ancak burada o interface' den hic bir update yapilmayacaktir. Oysa Access Listler ile birlikte olusturulacak Distribute List' ler ile hangi networklerin update edilecegine hatta hangi networklerin update' inin alinacagini karar verebiliriz.

Durumu ornek calisma ile ozetleyecegim.



Ornek topolojide her router icin 3'er adet loopback interface olusturdum ve konfigurasyon icinde bu loopbacklarde RIPv2 icerisinde tanittim. Baslangicta Routing Table' lar A ve B icin sirasiyla su sekilde olustu.

```

Router#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/24 is subnetted, 3 subnets
C       10.3.3.0 is directly connected, Loopback1
C       10.2.2.0 is directly connected, Loopback0
C       10.1.1.0 is directly connected, Ethernet0/0
  13.0.0.0/24 is subnetted, 3 subnets
R       13.3.3.0 [120/1] via 211.2.1.2, 00:00:01, Serial0/0
R       13.2.2.0 [120/1] via 211.2.1.2, 00:00:01, Serial0/0
R       13.1.1.0 [120/1] via 211.2.1.2, 00:00:01, Serial0/0
      211.2.1.0/30 is subnetted, 1 subnets
C       211.2.1.0 is directly connected, Serial0/0
Router#

```

```

Router#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/24 is subnetted, 3 subnets
R       10.3.3.0 [120/1] via 211.2.1.1, 00:00:20, Serial0/0
R       10.2.2.0 [120/1] via 211.2.1.1, 00:00:20, Serial0/0
R       10.1.1.0 [120/1] via 211.2.1.1, 00:00:20, Serial0/0
  13.0.0.0/24 is subnetted, 3 subnets
C       13.3.3.0 is directly connected, Loopback1
C       13.2.2.0 is directly connected, Loopback0
C       13.1.1.0 is directly connected, Ethernet0/0
      211.2.1.0/30 is subnetted, 1 subnets
C       211.2.1.0 is directly connected, Serial0/0
Router#

```

Her iki Router da Loopback adresler Directly Connected ve RIPv2 ile update edilmiş olarak görülmekteydi.

A routerında iki adet acces list yazdım ve bunlar RIPv2 konfigurasyonuna Distribute list komutu ile bağladım.

```

version 2
network 13.0.0.0
network 211.2.1.0
distribute-list 10 out
distribute-list 20 in
no auto-summary
!
ip http server
ip classless
!
access-list 10 deny 13.1.1.0 0.0.0.255
access-list 10 permit any
access-list 20 deny 10.2.2.0 0.0.0.255
access-list 20 permit any
!
line con 0
line aux 0
line vty 0 4
!
end
Router#

```

10 numaralı access list ile 13.1.1.0 networkunun, 20 numaralı access list ile 10.2.2.0 networkunu yasaklamak için gereken satırları yazdıktan sonra "in" ve "out" olarak Rip'e uyguladım.

Yazilan satirlarin tam turkcesi su sekildedir: 10.2.2.0 networkunu iceriden disariya gonderme, 13.1.1.0 netwrokune ait update' i disaridan iceriye alma.

Bu durumda Routing Table'lar su sekillerde degisti.

```
Router#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/24 is subnetted, 3 subnets
C    10.3.3.0 is directly connected, Loopback1
C    10.2.2.0 is directly connected, Loopback0
C    10.1.1.0 is directly connected, Ethernet0/0
  13.0.0.0/24 is subnetted, 2 subnets
R    13.3.3.0 [120/1] via 211.2.1.2, 00:00:15, Serial0/0
R    13.2.2.0 [120/1] via 211.2.1.2, 00:00:15, Serial0/0
      211.2.1.0/30 is subnetted, 1 subnets
C    211.2.1.0 is directly connected, Serial0/0
Router#
```

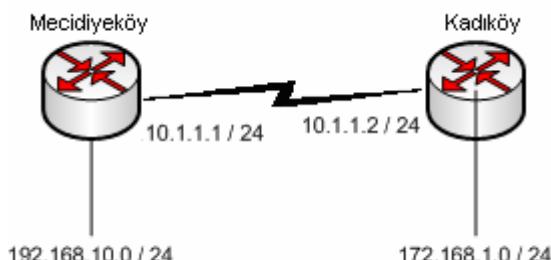
```
Router#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/24 is subnetted, 2 subnets
R    10.3.3.0 [120/1] via 211.2.1.1, 00:00:24, Serial0/0
R    10.1.1.0 [120/1] via 211.2.1.1, 00:00:24, Serial0/0
  13.0.0.0/24 is subnetted, 3 subnets
C    13.3.3.0 is directly connected, Loopback1
C    13.2.2.0 is directly connected, Loopback0
C    13.1.1.0 is directly connected, Ethernet0/0
      211.2.1.0/30 is subnetted, 1 subnets
C    211.2.1.0 is directly connected, Serial0/0
Router#_
```

CDP (Cisco Discovery Protocol)

CDP (Cisco Discovery Protocol) Cisco tarafından geliştirilmiş bir protokoldür. CDP ortamındaki Cisco cihazları hakkında bilgi almak için kullanılmaktadır. IOS 10.3 ve üzerinde CDP default olarak çalışmaktadır. Aşağıdaki örnekte routing işlemi gerçekleştirilmiş, çalışan bir sistem bulunmaktadır.



```
Mecidiyekoy(config)#cdp ?
  holdtime  Specify the holdtime (in sec) to be sent in packets
  timer      Specify the rate at which CDP packets are sent (in sec)
  run
```

Mecidiyeköy router'ında cdp yazdıktan sonra soru işaretini kullandık. Eğer IOS versiyonumuz 10.3 den düşük ise "cdp run" komutunu kullanarak CDP protokolünü çalıştırabiliriz.

```
Mecidiyekoy(config)#cdp run
Mecidiyekoy(config)#
```

CDP ile ilgili özellikleri görüntülemek için "sh cdp" komutu kullanılır.

```
Mecidiyekoy#sh cdp
Global CDP information:
  Sending CDP packets every 60 seconds
  Sending a holdtime value of 180 seconds
Mecidiyekoy#
```

Göründüğü gibi 60 saniyede bir CDP paketleri gönderilir. Bu paketler sayesinde ortamındaki Cisco cihazları hakkında bilgi toplanır. Eğer 180 saniye haber alınamaz ise cihaz database' den silinir. Bunlar default değerlerdir. Değiştirmek için aşağıdaki komutlar kullanılır.

```
Mecidiyekoy(config)#
Mecidiyekoy(config)#cdp timer 100
Mecidiyekoy(config)#cdp holdtime 100
```

Yukardaki örnekte hem CDP paketlerinin süresi hemde holdtime süresi 100 saniye olarak değiştirilmiştir.

CDP kullanılarak öğrenilen bilgiler "show cdp neighbors" komutu ile görülebilir.

```
Mecidiyekoy#
Mecidiyekoy#sh cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP
                  Device ID        Local Intrfce     Holdtme   Capability Platform Port ]
                  kadikoy          Ser 0             120        R           2610      Ser 0
Mecidiyekoy#
```

CDP ile öğrenilen bazı önemli bilgiler :

- **Device ID** : Bilgisi alınan aygıtın adını göstermektedir.
- **Port ID** : Komşu aygıtın bağlı olduğu port bilgisini göstermektedir.
- **Platform** : Komşu aygıtın donanım modelini göstermektedir.

Daha detaylı bilgi alabilmek için “sh cdp neighbor detail” komutu kullanılır.

```
Mecidiyekoy#sh cdp neighbors detail
-----
Device ID: kadikoy
Entry address(es):
  IP address: 10.1.1.2
Platform: cisco 2610, Capabilities: Router
Interface: Serial0, Port ID (outgoing port): Serial0/0
Holdtime : 122 sec

Version :
Cisco Internetwork Operating System Software
IOS (tm) C2600 Software (C2600-IS-M), Version 12.1(9), RELEASE SOFTWARE (f
Copyright (c) 1986-2001 by cisco Systems, Inc.
Compiled Wed 13-Jun-01 20:49 by kellythw
```

CDP protokolünün Router üzerinde çalışmasını engellemek için “no cdp run” komutu kullanılır.

```
Mecidiyeköy(config)#no cdp run
Mecidiyeköy(config)#
```

Ayrıca CDP protokolü interface bazında çalıştırılabilir. Interface içerisinde “cdp enable” komutu kullanılır.

```
Mecidiyeköy(config)#int ser 0
Mecidiyeköy(config-if)#cdp enable
Mecidiyeköy(config-if)#_
```

“no cdp enable” komutu ile interface içerisinde CDP kullanımı kaldırılır.

```
Mecidiyeköy(config)#interface serial 0
Mecidiyeköy(config-if)#no cdp enable
Mecidiyeköy(config-if)#|
```

EIGRP(Enhanced Interior Gateway Routing Protocol)

Cisco daha önce geliştirdiği IGRP' nin yetersiz kalması ve RIP'in RIPv2'ye yükseltilmesiyle boş durmamış, EIGRP' yi geliştirmiş ve bu protokolü sınıflandırmada da, hem Distance Vektör hem de Link State protokollerin özelliklerini taşıdığı için Hybrid başlığı altına yerleştirmiştir.

Bütün Routing protokolleri gibi EIGRP' de Routing update mantığı ile çalışır fakat Rip ve IGRP' den farklı olarak belirli zaman aralıklarında tüm networklerin bilgisini göndermektense küçük hello paketleri yollayarak komşu routerlarının up olup olmadıklarını kontrol eder. Komşu routernardan gelen Acknowledgement paketleriyle o routerın hala up olduğu kabul eder.

Hello ve Acknowledgement mesajları dikkate alındığında burada TCP gibi bir protokolün kullanılması gerekliliği ortaya çıkar. Fakat bu işlemler sırasında EIGRP yine Cisco'nun geliştirdiği ve RTP (Reliable Transport Protocol) protokolünü kullanır. Çalışma mantığı TCP ile aynıdır.

Gerektiği zamanlarda, sözgelimi yeni bir router eklendiğinde veya bir router down olduğunda, "ADD" ya da "DELETE" bilgilerini yollar.

Bir router ortama dahil olduğunda öncelikle bir Query paketi yollar ve bu paketlerden gelen Reply'lar ile komşu routerları hakkında bilgi edinir ve topoloji tablosunu oluşturur.

Buraya kadar anlattıklarımızla EIGRP' nin 5 farklı paket ile çalıştığını söyleyebiliriz.

EIGRP Paketleri

- Hello
- Acknowledgement
- Update
- Query
- Reply

EIGRP Hello paketlerini 224.0.0.10 multicast ip adresi üzerinden gönderir. T1 ve üzeri bant genişliklerinde 5 saniye de bir gönderilen bu paketler T1 den daha düşük bant genişliklerinde 60 saniyede bir gönderili. (Hold Time=3 X hello interval)

Acknowledgement paketleri data içermeyen paketlerdir ve güvenli iletişim sağlar. Hello paketlerinin multicast olmasına karşın Acknowledgement paketleri unicast çalışırlar.

Update paketleri sistemdeki bir router yeni bir network bulduğunda ya da kaybettiginde, metric hesabında bir değişiklik olduğunda ve Successor değişiginde gönderilir. Bu aksiyonlardan biri gerçekleştiğinde EIGRP konusun bir Router bütün komsularını multic平update gönderir.

Query paketleri bir router herhangi bir şekilde yeni, özel bir bilgiye ihtiyaç duyuldugunda gönderilir. Sözelimi Successor' i down olan ve Feasible Successor' i bulunmayan bir router Query paketleri gönderir ve cevaplar Reply paketleri ile doner. Query paketleri multicast iken Reply paketleri unic平tir.

EIGRP Metric Hesabi

EIGRP metric hesabında K1 ve K3 değerlerini kullanır. (Bandwidth ve Delay)

```
Router> show interface s0/0
Serial0/0 is up, line protocol is up
  Hardware is QUICC Serial
  Description: Out to VERIO
  Internet address is 207.21.113.186/30
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
    rely 255/255, load 246/255
  Encapsulation PPP, loopback not set
  Keepalive set (10 sec)
<output omitted>
```

EIGRP ve IGRP bant genişliklerini aynı formül ile hesaplarlar.

$$\text{metric} = [\text{K1} * \text{bandwidth} + (\text{K2} * \text{bandwidth}) / (256 - \text{load}) + (\text{K3} * \text{delay})] * [\text{K5} / (\text{reliability} + \text{K4})]$$

Fakat EIGRP için K2, K4 ve K5 default olarak 0 sayılır.

EIGRP Table'ları

EIGRP çalışma mantığı içerisinde bütün komşularını Neighbor Table'da ve hedef networke olan bütün yolları da Topology Table'da tutar. Bu bilgiler işığında en iyi yol seçimini yapar.

RouterC#show ip eigrp neighbors								
IP-EIGRP neighbors for process 44								
H	Address	Interface	Hold	Uptime	SRTT	RTO	Q	Seq
			(sec)		(ms)		Cnt	Num
0	192.168.0.1	Se0	11	00:03:09	1138	5000	0	6
1	192.168.1.2	Eto	12	00:34:46	4	200	0	4

Neighbor Table'da komsu routerların network katmanı adresleri (ip adresleri), Q ile gösterilen ve sırası gonderilmeyen bekleyen paket sayısını ifade eden bir değer (ki bu değer 0 dan büyük ise router da olası bir problemden bahsedilebilir), SRTT ile gösterilen ve komsu routerlara gonderilen ve alınan paketler için geçen ortalama sureyi gösteren bir değer ve Hold Time değeri bulunur.

```
RouterB#show ip eigrp topology
IP-EIGRP Topology Table for process 44
Codes: P - Passive, A - Active, U - Update, Q - Query, R -
      Reply, r - Reply status
P 206.202.17.0/24, 1 successors, FD is 2195456
      via 206.202.16.1 (2195456/2169856), Ethernet0
P 206.202.18.0/24, 2 successors, FD is 2198016
      via 192.168.0.2 (2198016/284160), Serial0
      via 206.202.16.1 (2198016/2172416), Ethernet0
```

EIGRP hedef networklere gitmek için kullanacağı yolların bilgisini ise Topology Table'ında saklar. Bu table'da bulunan bilgilere dayanarak Successor ve Feasible Successor'u seçer.

Routing Table ise Successor (best route) olarak seçilen yolu bulundugu yerdir.

```
RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
      BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
Gateway of last resort is not set
C    10.1.1.0 is directly connected, Serial0
D    172.16.0.0 [90/2681856] via 10.1.1.0, Serial0
D EX  192.168.1.0 [170/2681856] via 10.1.1.1, 00:00:04, Serial0
```

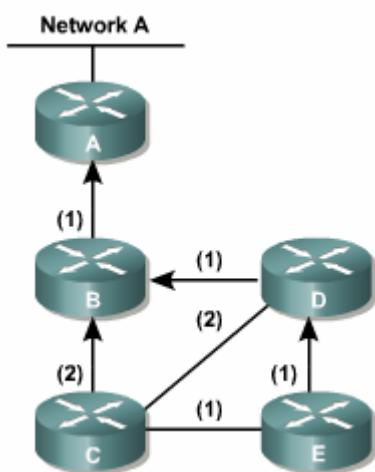
EIGRP harici bir protokolden gelen update bilgileri Routing Table'ında EX (external) olarak işaretler.

EIGRP topolojini oluştururken Dual Algoritmasını kullanır. Bu algoritma ile kendisine bir en iyi yol (Successor) bir de yedek söylelecek en iyi ikinci yol (Feasible Successor) seçer.

Successor seçerken tek dayanağı mümkün olan yollara ait metrik toplamlarının (Her biri Feasible Distance olarak adlandırılır.) en küçüğünü kullanır. Feasible Distance'ları eşit olan birden fazla yol var ise en düşük Reported Distance' a sahip olan yolu seçer. Burada Reported Distance' dan kasıt adından anlaşılabileceği gibi bir sonraki router için geçerli olan Feasible Distance' dır.

Burada bir önemli kuralda, Feasible Successor seçilen yola ait Reported Distance değeri, Successor seçilen yolu Feasible Distance'ından küçük olmalıdır, aksi takdirde loop başlar.

Örnek üzerinde açıklamak gerekirse;



(Parantez İçindeki değerler metrik değerleridir.)

C Routerından Net A ya gidilme istendiğinde topoloji şöyle olacak;

Next Hop	FD	RD	Topoloji
B	3		Successor
D	4	2	FS
E	4	3	

En iyi yol B routerı üzerinden gidilen yoldur, çünkü metrik değerleri toplandığında en küçük değere (Feasible Distance) sahiptir.

Feasible Distance'ları eşit olan D ve E routerları üzerinde gidilen yollar için Reported Distance'ı küçük olan (D) Feasible Successor seçilir. (Burada D routerı için RD değerinin B routerı FD değerinden küçük olduğunu dikkat edin)

D Router'ından Net A ya gidilme istendiğinde topoloji şöyle olacak;

Next Hop	FD	RD	Topoloji
B	2		Successor
E	5	4	
C	5	3	

Burada görüldüğü gibi Feasible Successor seçilemiyor çünkü Reported Distance değerleri hem E hem de C routerı için B routerının Feasible Distance'ından büyük.

(Feasible Successor' a default route' da denmektedir.)

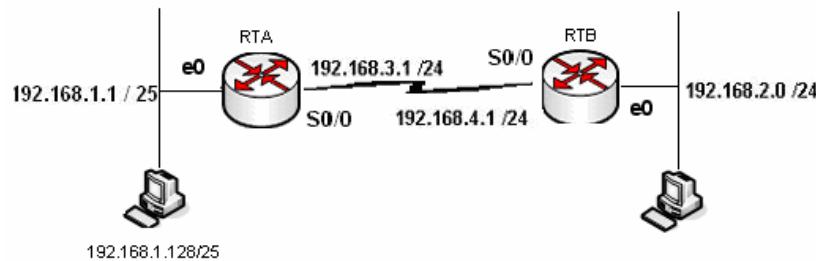
Not: EIGRP IPX ve AppleTalk networklerini de destekler ve bu networklere ait Neigbor, Topology ve Routing table' lari ayri ayri tutar.

Auto Summarization

Auto Summarization ve Load Balancing özellikleri detaylı olarak incelenmelidir. (Auto Summarization özelliği Ripv2'de de vardır.)

Sözelimi elimizde, interfacelerinde sırasıyla s0=192.168.1.1, s1=10.1.1.0 / 25 ve s2=10.1.1.128 / 25 networkleri olan bir router (Router A) var ve s0 interface'inden başka bir routera (Router B) bağlı.

Routerlar EIGRP ile konfigure edildiği zaman A routera B routera Auto Summarization yapacak ve 10.1.1.0 / 24 networkü bilgisini update edecktir. Bu istenmeyen bir durum ise "no auto-Summarization" komutu ile özellik kaldırılabilir.



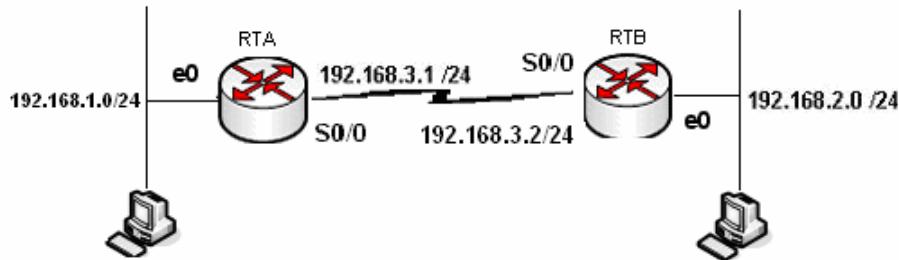
Auto Summarization özelliği "no auto-summary" komutu ile kaldırılabilir.

```

Router(config)#router eigrp 34
Router(config-router)#no auto-summary
  
```

EIGRP Konfigurasyonu

EIGRP de tipki IGRP gibi konfigure edilir.



Router A

```
RouterA(config)#router eigrp 34
RouterA(config-router)#network 192.168.1.0
RouterA(config-router)#network 192.168.3.0
RouterA(config-router)#no auto-summary
```

Router B

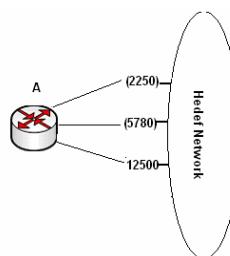
```
RouterB(config)#router eigrp 34
RouterB(config-router)#network 192.168.2.0
RouterB(config-router)#network 192.168.3.0
RouterB(config-router)#no auto-summary
```

Load Balancing

Rip söz konusu olduğunda, Metric hesabı tamamen hop sayısına bağlı olduğundan aynı metriğe sahip birden fazla yol olması ve bu yollar arasından router in load Balancing yapması ihtimaller arasındadır. Fakat EIGRP' yi de içene alan diğer bütün protokoller de Metric hesabı birçok değerle birlikte yapıldığı için, aynı metriğe sahip birden fala yolun olması çok çok zor bir ihtimaldir.

Bu durumda load Balancing imkânsızdır. Fakat EIGRP "variance n" komutu ile load balancing yapılmasına izin verir. (Bu özellik IGRP' de de vardır.)

Bu komutta n ile belirtilen bölüm, bizim belirleyeceğimiz bir sayıdır. Ve komut işletilmeye başladığında EIGRP en düşük Metric değerini alır, n ile çarpar ve çıkan sonucun altında yer alan bütün Metric değerlerine sahip yollar arasında load Balancing yapmaya başlar.

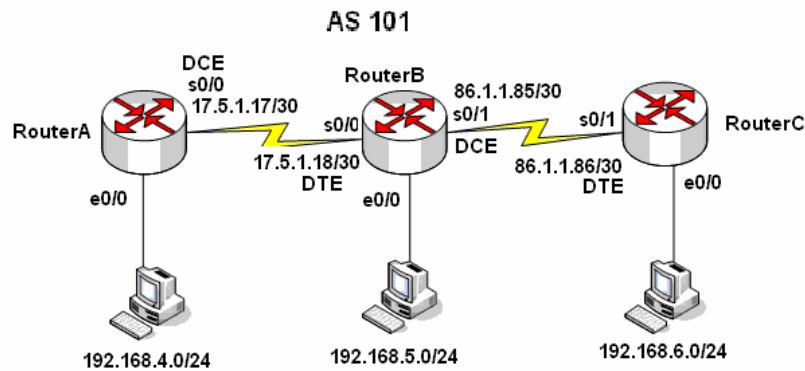


Burada variance 3 gibi bir komut kullanırsak,, bu komut en düşük metric değeri olan 2250' yi 3 ile çarpacak ve çıkan sonucun (6750) altında metric değerlerine sahip yollar (2250 ve 5780 metricli yollar) arasında load balancing yapacaktır.

Ornek Konfigurasyon;

```
Router(config)#router eigrp 14
Router(config-router)#network 10.1.1.0
Router(config-router)#network 10.2.1.0
Router(config-router)#network 10.3.1.0
Router(config-router)#variance 2
```

EIGRP Laboratuar Çalışması



Burada yapılan çalışmada AS olarak 101 seçilmiştir.

Laboratuar ortamında clock üretimini saglayacak DCE kabloların takıldığı interfacelere uygulama içerisinde clock rate komutu verilmistir.

Auto Summarization özelliği kapatılmıştır.

Her bir Router, dan Runnin-config dosyaları, Routing Table'ları, Neighbor Table'ları ve Topology Table'ları alınmıştır.

```
RouterA#show running-config
Building configuration...

Current configuration : 642 bytes
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterA
!
!
memory-size iomem 10
ip subnet-zero
!
!
interface Ethernet0/0
 ip address 192.168.4.1 255.255.255.0
!
interface Serial0/0
 ip address 17.5.1.17 255.255.255.252
 clockrate 64000
!
interface BRI0/0
 no ip address
 shutdown
 isdn x25 static-tei 0
!
router eigrp 101
 network 17.5.1.16 0.0.0.3
 network 192.168.4.0
 no auto-summary
 no eigrp log-neighbor-changes
!
ip classless
!
!
line con 0
line aux 0
line vty 0 4
!
no scheduler allocate
end
```

RouterA#

```
RouterB#show running-config
Building configuration...

Current configuration : 640 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterB
!
memory-size iomem 10
ip subnet-zero
!
interface Ethernet0/0
 ip address 192.168.5.1 255.255.255.0
 half-duplex
!
interface Serial0/0
 ip address 17.5.1.18 255.255.255.252
!
interface Serial0/1
 ip address 86.1.1.85 255.255.255.252
 clockrate 64000
!
router eigrp 101
 network 17.5.1.16 0.0.0.3
 network 86.1.1.84 0.0.0.3
 network 192.168.5.0
 no auto-summary
!
ip classless
!
!
dial-peer cor custom
!
gatekeeper
 shutdown
!
line con 0
line aux 0
line vty 0 4
!
end
```

RouterB#

```
RouterC#sh running-config
```

```
Building configuration...
```

```
00:29:43: IP-EIGRP: Neighbor 192.168.4.1 not on common subnet for Ethernet0/0 (192.168.6.1
255.255.255.0)
Current configuration : 620 bytes
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterC
!
!
memory-size iomem 10
ip subnet-zero
!
interface Ethernet0/0
 ip address 192.168.6.1 255.255.255.0
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
!
interface Serial0/1
 ip address 86.1.1.86 255.255.255.252
!
router eigrp 101
 network 86.1.1.84 0.0.0.3
 network 192.168.6.0
 no auto-summary
 no eigrp log-neighbor-changes
!
ip classless
!
line con 0
line aux 0
line vty 0 4
!
no scheduler allocate
end
```

```
RouterC#
```

```

RouterA#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
C    17.5.1.16 is directly connected, Serial0/0
  86.0.0.0/30 is subnetted, 1 subnets
D    86.1.1.84 [90/2681856] via 17.5.1.18, 00:14:16, Serial0/0
C    192.168.4.0/24 is directly connected, Ethernet0/0
D    192.168.5.0/24 [90/2195456] via 17.5.1.18, 00:01:44, Serial0/0
D    192.168.6.0/24 [90/2707456] via 17.5.1.18, 00:13:29, Serial0/0
RouterA#

```

39:45 bağlanıldı OtoAlgıla 9600 8-N-1 Kaydır büyült SAYI Yakala Yazdırma yankısı

```

RouterA#sh ip eigrp neighbors
IP-EIGRP neighbors for process 101
H  Address           Interface   Hold Uptime   SRTT     RTT Q Seq Type
   (sec)          (ms)          Cnt Num
0  17.5.1.18         Se0/0        13 00:10:14  607    3642  0  23
RouterA#sh ip eigrp top
RouterA#sh ip eigrp topology
IP-EIGRP Topology Table for AS(101)/ID(192.168.4.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 86.1.1.84/30, 1 successors, FD is 2681856
via 17.5.1.18 (2681856/2169856), Serial0/0
P 17.5.1.16/30, 1 successors, FD is 2169856
via Connected, Serial0/0
P 192.168.4.0/24, 1 successors, FD is 281600
via Connected, Ethernet0/0
P 192.168.6.0/24, 1 successors, FD is 2707456
via 17.5.1.18 (2707456/2195456), Serial0/0
RouterA#

```

39:45 bağlanıldı OtoAlgıla 9600 8-N-1 Kaydır büyült SAYI Yakala Yazdırma yankısı

```

RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
C    17.5.1.16 is directly connected, Serial0/0
  86.0.0.0/30 is subnetted, 1 subnets
C    86.1.1.84 is directly connected, Serial0/1
D    192.168.4.0/24 [90/2195456] via 17.5.1.17, 00:12:37, Serial0/0
C    192.168.5.0/24 is directly connected, Ethernet0/0
D    192.168.6.0/24 [90/2195456] via 86.1.1.86, 00:11:48, Serial0/1
RouterB#

```

```

IP-EIGRP neighbors for process 101
H Address Interface Hold Uptime SRTT RT0 Q Seq Type
  1 86.1.1.86     Se0/1      14 00:12:44  672 4032 0 14
  0 17.5.1.17     Se0/0      12 00:13:36   21 200 0 14
RouterB#show ip eigrp topology
IP-EIGRP Topology Table for AS(101)/ID(192.168.5.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 86.1.1.84/30, 1 successors, FD is 2169856
  via Connected, Serial0/1
P 17.5.1.16/30, 1 successors, FD is 2169856
  via Connected, Serial0/0
P 192.168.4.0/24, 1 successors, FD is 2195456
  via 17.5.1.17 (2195456/281600), Serial0/0
P 192.168.5.0/24, 1 successors, FD is 281600
  via Connected, Ethernet0/0
P 192.168.6.0/24, 1 successors, FD is 2195456
  via 86.1.1.86 (2195456/281600), Serial0/1
RouterB#

```

```

RouterC#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
D    17.5.1.16 [90/2681856] via 86.1.1.85, 00:00:46, Serial0/1
  86.0.0.0/30 is subnetted, 1 subnets
C    86.1.1.84 is directly connected, Serial0/1
D    192.168.4.0/24 [90/2707456] via 86.1.1.85, 00:00:46, Serial0/1
D    192.168.5.0/24 [90/2195456] via 86.1.1.85, 00:00:46, Serial0/1
C    192.168.6.0/24 is directly connected, Ethernet0/0
RouterC#

```

```

RouterC#show ip eigrp neighbors
IP-EIGRP neighbors for process 101
H Address Interface Hold Uptime SRTT RT0 Q Seq Type
  0 86.1.1.85     Se0/1      14 00:05:50   28 200 0 21
RouterC#

```

```

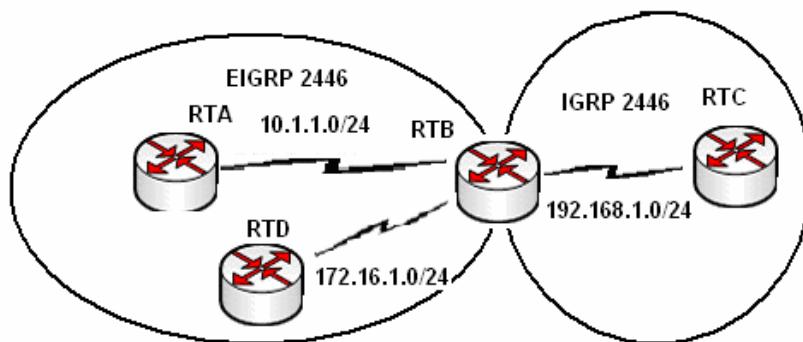
RouterC#show ip eigrp topology
IP-EIGRP Topology Table for AS(101)/ID(192.168.6.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 86.1.1.84/30, 1 successors, FD is 2169856
  via Connected, Serial0/1
P 17.5.1.16/30, 1 successors, FD is 2681856
  via 86.1.1.85 (2681856/2169856), Serial0/1
P 192.168.4.0/24, 1 successors, FD is 2707456
  via 86.1.1.85 (2707456/2195456), Serial0/1
P 192.168.5.0/24, 1 successors, FD is 2195456
  via 86.1.1.85 (2195456/281600), Serial0/1
P 192.168.6.0/24, 1 successors, FD is 281600
  via Connected, Ethernet0/0
RouterC#

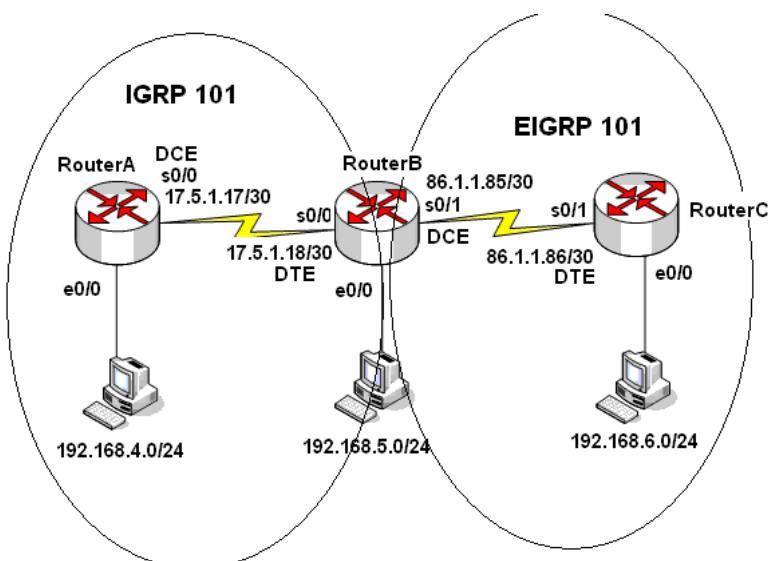
```

EIGRP ve IGRP Birlikte Çalışması



```
RTB(config)#router igrp 2446
RTB(config-router)#network 192.168.1.0
RTB(config)#router eigrp 2446
RTB(config-router)#network 10.1.1.0
RTB(config-router)#network 172.16.1.0
```

IGRP ve EIGRP ayno AS içerisinde birbirleriyle haberleşirler. Burada özel olarak dikkat edilecek tek nokta EIGRP konusun Routerların Routing Tablalarında IGRP konusun Routerlara giden yolları External olarak etiketlemis olmasidir.



Hem IGRP hem de EIGRP için AS numarası 101 seçilmiştir.

Router B üzerinde hem IGRP hem EIGRP konfigürasyonları yapılmıştır.

Bütün Routerların Routing Table'ları ve B routerinin running-config dosyası incelenmek üzere alınmıştır.

```
RouterB#show run
Building configuration...
Current configuration : 670 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterB
!
memory-size iomem 10
ip subnet-zero
!
interface Ethernet0/0
 ip address 192.168.5.1 255.255.255.0
 half-duplex
!
interface Serial0/0
 ip address 17.5.1.18 255.255.255.252
!
interface Serial0/1
 ip address 86.1.1.85 255.255.255.252
 clockrate 64000
!
router eigrp 101
 network 86.1.1.84 0.0.0.3
 network 192.168.5.0
 no auto-summary
!
router igrp 101
 network 17.0.0.0
 network 192.168.5.0
!
ip classless
!
dial-peer cor custom
!
gatekeeper
 shutdown
line con 0
line aux 0
line vty 0 4
!
end
```

RouterB#

RouterA Routing Table'i

```
RouterA#sh ip route
00:38:31: %SYS-5-CONFIG_I: Configured from console by console
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
C    17.5.1.16 is directly connected, Serial0/0
I  86.0.0.0/8 [100/10476] via 17.5.1.18, 00:00:03, Serial0/0
C  192.168.4.0/24 is directly connected, Ethernet0/0
I  192.168.5.0/24 [100/8576] via 17.5.1.18, 00:00:03, Serial0/0
I  192.168.6.0/24 [100/10576] via 17.5.1.18, 00:00:03, Serial0/0
RouterA#_
```

RouterB Routing Table'i

```
RouterB#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
C    17.5.1.16 is directly connected, Serial0/0
  86.0.0.0/30 is subnetted, 1 subnets
C    86.1.1.84 is directly connected, Serial0/1
I  192.168.4.0/24 [100/8576] via 17.5.1.17, 00:01:14, Serial0/0
C  192.168.5.0/24 is directly connected, Ethernet0/0
D  192.168.6.0/24 [90/2195456] via 86.1.1.86, 00:03:45, Serial0/1
RouterB#_
```

RouterC Routing Table'i

```
RouterC#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
D EX  17.5.1.16 [170/2681856] via 86.1.1.85, 00:02:43, Serial0/1
  86.0.0.0/30 is subnetted, 1 subnets
C    86.1.1.84 is directly connected, Serial0/1
D EX  192.168.4.0/24 [170/2707456] via 86.1.1.85, 00:01:26, Serial0/1
D  192.168.5.0/24 [90/2195456] via 86.1.1.85, 00:02:43, Serial0/1
C  192.168.6.0/24 is directly connected, Ethernet0/0
RouterC#_
```

OSPF (Open Shortest Path First)

OSPF Link State Protocol olup, ulaşılmak istenen networke giden en kısa yolu Dijkstra algoritması kullanarak tespit etmektedir.

"Hello" protokolü ile OSPF çalışan routerlar komşularını keşfederler. Hello paketleri her 10 saniye de bir gönderilir ve bu paketlerden alınan sonuçlara göre OSPF database oluşturulur.

OSPF metrik için Cost adı verilen değeri kullanırlar. Standart bir tanımı yapılamamakla birlikte Cisco Routerlar da ön görülen OSPF metriği bant genişliği ile ters orantılıdır.

(cost= 10.000.000 / bantgenisligi)

Bu protokolde, networkteki yönlendirme bilgilerini kendisinde toplayıp, diğerlerine dağıtacak bir router vardır. Bu routera Designated Router denir ve DR olarak kısaltılır.

DR aktif olmadığı durumlarda Backup Designated Router devreye direr. (BDR)

Hello Paket İçeriği (Type 1)

Router ID: Router da konfigüre edilen en yüksek IP adresidir.

Network Mask: Router ID' yi belirleyen interface'in ağ maskesidir.

Area ID: Hello paketi gönderen routerın interface'inin alan kimliğidir. Hello paketindeki bilgilerin geçerli olabilmesi için bu paketi alan routerın interface'i ile aynı olmalıdır.

Router Priority: Routerın DR veya BDR seçimini belirlemektedir.

Hello Aralığı: Hello paketleri arasındaki süredir ve 10 saniyedir.

Router(config-if)#ip ospf hello-interval *n* komutuyla degistirilebilir. *n* bizim belirleyeceğimiz birimim saniye olan bir degerdir. Burada dikkat edilmesi gereken bir konu ise birbirine bagla olan iki interface ' inde hello zaman araliginin esit olması gerektigidir. Aksi takdirde komsuluk ilişkisi kurulamaz.

Ölüm Aralığı (Dead Interval): Komşu router ile bağlantının koptuğunu belirten süredir. (Hello Aralığının 4 katıdır.)

DR IP adresi: Mevcut DR ip adresidir. Bu adresi öğrenen Routerlar, OSPF mesajlarını bu ip adresine gönderirler.

BDR IP Adresi: Mevcut BDR ip adresidir. DR aktif olmadığı zaman OSPF mesajları bu ip adresine gönderilir.

Komşu Router ID'leri: Komşuluk tablosunda bulunan routerların ip adresleridir. Router kendi ip adresini bu alanda görürse database paylaşımı gerçekleştirilir.

Authentication Information: Kimlik doğruluma tipi ve bilgisini içerir.

Stub Area Flag: Hangi tip LSA (Link State Advertisement) mesajlarının gönderileceği ve alınacağı bilgisini içerir.

Hello paketleri disinda OSPF konusan Routerların birbirlerine gonderdikler 4 ayrı paket sekli daha vardır. Bunlar;

Type2: DBD yani Database Descriptiin paketleri olarak bilinir ve Routerların Link durumları hakkında ozet bilgiler icerir.

Type3: LSR yani Link State Request paketleri olarak bilinir. Routerlar DBD paketleri ile öğrenmek istedikleri bilgilerin detayı için diğer Routerlara LSR paketleri gönderebilir.

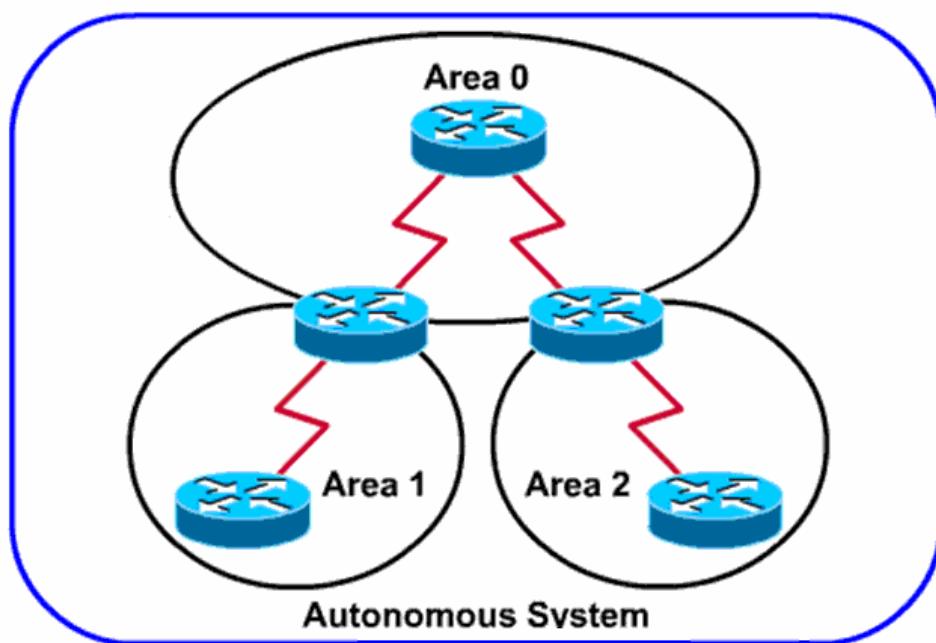
Type4: LSU yani Link State Update paketleri olarak bilinir. LSR ile istenen Link State Advertisements (LSAs) paketlerini taşır.

Type5: LSA yani Link State Acknowledgement paketleridir ve routerlar arasında paketlerin alındığı onay bilgisini taşır.

OSPF Area

Ospf çalışma mantığı arealar üzerinde kurulmuştur ve bu sayede bir dizayn hiyerarşisi sağlanabilmektedir. Bu hiyerarsik yapının convergence'si hızlandırdığı da söylenebilir.

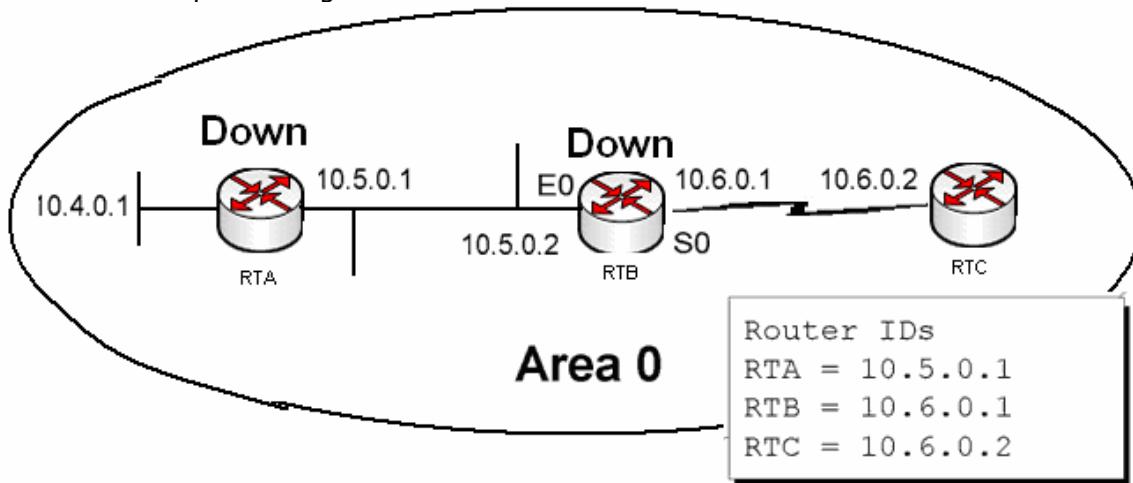
Ospf'in merkezi area 0'dır. Area 0 backbone area olarak adlandırılır ve farklı arealar olduğunda o arealar içinde area 0 ile konuşan interface' e sahip routerlar olmalıdır.



OSPF Komşuluğu

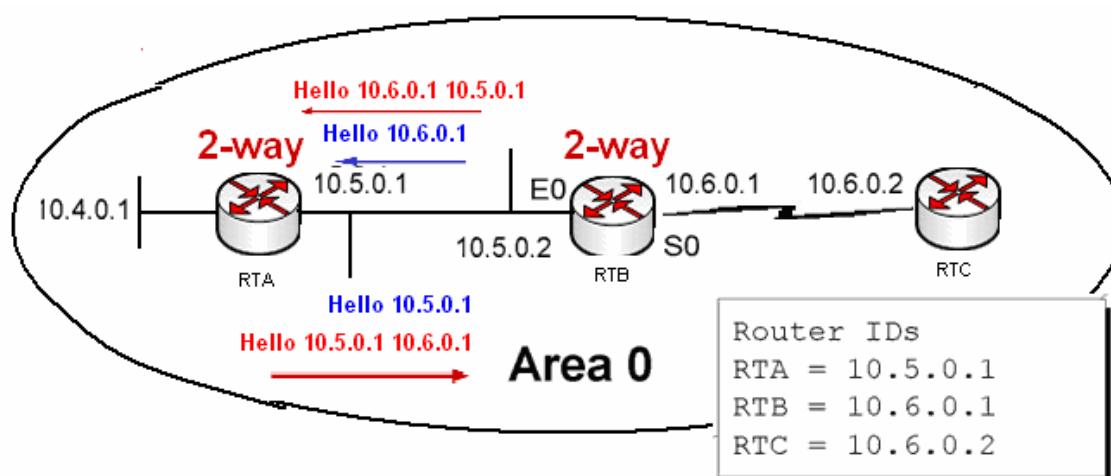
OSPF ile konfigüre edilmiş routerlar 7 adım ile diğer routerlar ile komşuluk kurarlar. Bu adımlar şunlardır;

Down, Hello paketinin alınamadığı durumdur. Yeni bir router networke katıldığından down durumdadır. Routerlar networkteki varlıklarını duyurmak için 224.0.0.5 multicast adresini kullanarak Hello paketleri gönderir.



Init, Diğer routerlardan cevap beklemeye adımdır.

Two-Way, Diğer routerların gönderdikleri Hello mesajlarının Komşu Router ID alanında kendi IP adreslerini gördükleri durumdur. Artık iki router komşuluk bağı kurmuştur.



Exstart, Karşılıklı iki router arasında paket alış verişinin yapıldığı andır. Bu adımda iki Router dan biri master diğeri slave rolü üstlenir. Burada seçim sadece iletişimini başlatacak routerı belirlemek için kullanılır, bu seçim herhangi birine bir üstünlük sağlamaz.

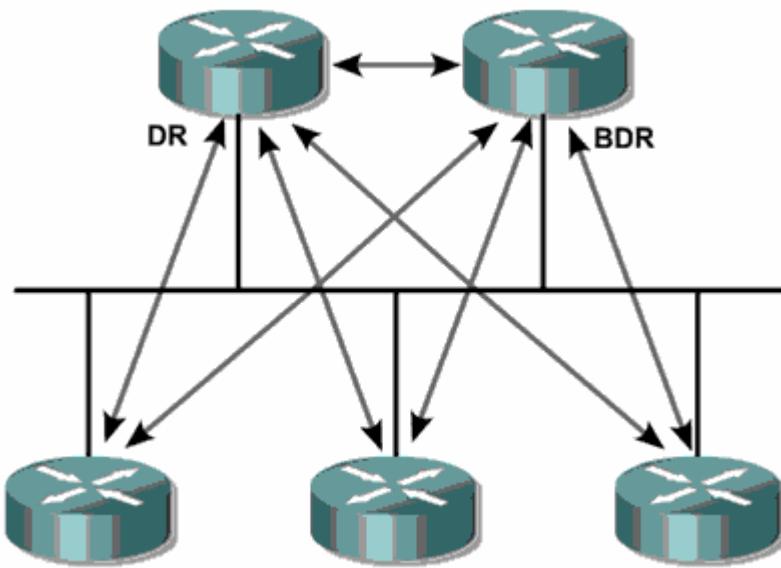
Exchange, Routerların bilgi alışverişini yaptıkları adımdır.

Loading, Exchange adımı ile elde edilen yeni yollar / networkler hakkındaki bilgileri ilgili routerlardan alma adımdır.

Full, Yönlendirme bilgilerinin senkron hale getirilmesi durumudur.

DR ve BDR Secimi

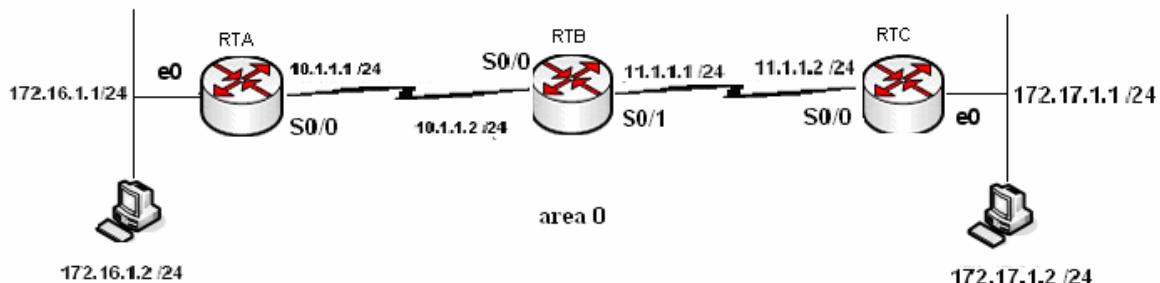
Multi-acces networklerde işler biraz daha farklı yürürlükte. Bu networklerde Two Way halindeyken ortamda bütün trafiği yönetecek bir router seçilir ki bu da Designated Router (DR) denir. Ve yine Backup Designated Router (BDR) denen ve DR'ın yedekisi olan bir router daha seçilir.



DR ve BDR seçimleri Router ID'ler ile yapılır. En yüksek Router ID'ye sahip router DR ve ikinci en yüksek ID'ye sahip router BDR seçilir.

Router ID bir routerin aktif olan interfacelerindeki en yüksek IP adresidir. Burada loopback adreslerin bir ayrıcalığı vardır. Eğer bir Routerda loopback adresi tanımlanırsa o routerın ID'si loopback IP'sidir. IP adresinin küçük veya büyük olması durumu değiştirmez.

Single Area OSPF Konfigürasyonu



Router A Konfigürasyonu;

```
A(config)#router ospf 1
A(config-router)#network 172.16.1.0 0.0.0.255 area 0
A(config-router)#network 10.1.1.0 0.0.0.255 area 0
A(config-router)#exit
A(config)#
```

Router B Konfigürasyonu;

```
B(config)#router ospf 1
B(config-router)#network 10.1.1.2 0.0.0.255 area 0
B(config-router)#network 11.1.1.2 0.0.0.255 area 0
B(config-router)#exit
B(config)#_
```

1:50:48 saatlerdir | Oturumda | Açılan A-N-1 | Kavşak | İňih | SAVT | Yakala | Yazdırma

Router C Konfigürasyonu;

```
C(config)#router ospf 1
C(config-router)#network 11.1.1.0 0.0.0.255 area 0
C(config-router)#network 172.17.1.0 0.0.0.255 area 0
C(config-router)#exit
C(config)#_
```

(A Router'ının Routing Table'sı)

```
A#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  172.17.0.0/24 is subnetted, 1 subnets
O    172.17.1.0 [110/943] via 10.1.1.2, 00:45:03, Serial0/0
  172.16.0.0/24 is subnetted, 1 subnets
C    172.16.1.0 is directly connected, Ethernet0/0
  10.0.0.0/24 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, Serial0/0
  11.0.0.0/24 is subnetted, 1 subnets
O    11.1.1.0 [110/933] via 10.1.1.2, 00:45:03, Serial0/0
A#_
```

(B Router'ının Routing Table'sı)

```
B#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route, o - ODR

Gateway of last resort is not set

  172.17.0.0/24 is subnetted, 1 subnets
O    172.17.1.0 [110/879] via 11.1.1.2, 00:43:08, Serial3
  172.16.0.0/24 is subnetted, 1 subnets
O    172.16.1.0 [110/74] via 10.1.1.1, 00:43:08, Serial1
  10.0.0.0/24 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, Serial1
  11.0.0.0/24 is subnetted, 1 subnets
C    11.1.1.0 is directly connected, Serial3
B#
```

(C Router'ının Routing Table'sı)

```
C#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

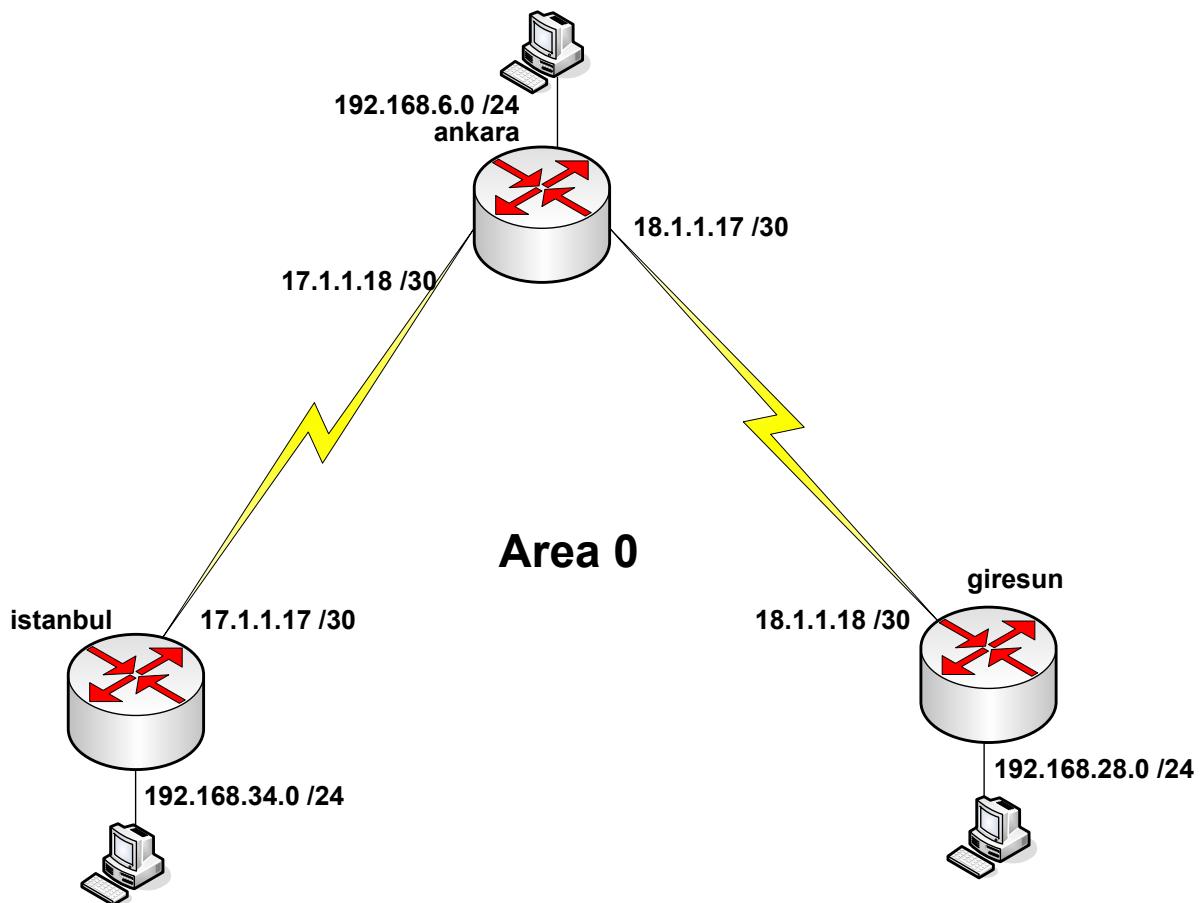
  172.17.0.0/24 is subnetted, 1 subnets
C    172.17.1.0 is directly connected, Ethernet0/0
  172.16.0.0/24 is subnetted, 1 subnets
O      172.16.1.0 [110/138] via 11.1.1.1, 00:43:54, Serial0/0
  10.0.0.0/24 is subnetted, 1 subnets
O      10.1.1.0 [110/128] via 11.1.1.1, 00:43:54, Serial0/0
  11.0.0.0/24 is subnetted, 1 subnets
C    11.1.1.0 is directly connected, Serial0/0
C#_
```

10:55:06 bağlandı | OtoAloala | 9600 8-N-1 | Kaydır | büyüh | SAYI | Yakala | Yazdırma yanısı |

(Hello Paketleri)

```
C#debug ip ospf events
OSPF events debugging is on
C#
01:15:44: OSPF: Rcv pkt from 172.16.1.1, Ethernet0/0, area 0.0.0.0 : src not on
the same network
01:15:48: OSPF: Rcv hello from 11.1.1.1 area 0 from Serial0/0 11.1.1.1
01:15:48: OSPF: End of hello processing
01:15:54: OSPF: Rcv pkt from 172.16.1.1, Ethernet0/0, area 0.0.0.0 : src not on
the same network
01:15:58: OSPF: Rcv hello from 11.1.1.1 area 0 from Serial0/0 11.1.1.1
01:15:58: OSPF: End of hello processing
01:16:04: OSPF: Rcv pkt from 172.16.1.1, Ethernet0/0, area 0.0.0.0 : src not on
the same network
01:16:08: OSPF: Rcv hello from 11.1.1.1 area 0 from Serial0/0 11.1.1.1
01:16:08: OSPF: End of hello processing
01:16:14: OSPF: Rcv pkt from 172.16.1.1, Ethernet0/0, area 0.0.0.0 : src not on
the same network
01:16:18: OSPF: Rcv hello from 11.1.1.1 area 0 from Serial0/0 11.1.1.1
01:16:18: OSPF: End of hello processing
01:16:24: OSPF: Rcv pkt from 172.16.1.1, Ethernet0/0, area 0.0.0.0 : src not on
the same network
```

OSPF Laboratuar Çalışmaları



```

!
router ospf 101
network 17.1.1.16 0.0.0.3 area 0
network 192.168.34.0 0.0.0.255 area 0
!
ip classless
no ip http server
!
!
!
line con 0
 transport input none
line aux 0
line vty 0 4
!
no scheduler allocate
end

istanbul# sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route

Gateway of last resort is not set

C  192.168.34.0 is directly connected, Ethernet0
    17.0.0.0/30 is subnetted, 1 subnets
C    17.1.1.16 is directly connected, Serial1
O    192.168.6.0 [110/64] via 17.1.1.18, 00:03:10, Serial1
    18.0.0.0/30 is subnetted, 1 subnets
O    18.1.1.16 [110/64] via 18.1.1.17, 00:03:00, Serial1
O    192.168.28.0 [110/192] via 17.1.1.18, 00:03:31, Serial1

```

istanbul#

```

!
router ospf 101
network 192.168.6.0 0.0.0.255 area 0
network 17.1.1.16 0.0.0.3 area 0
network 18.1.1.16 0.0.0.3 area 0
!
ip classless
no ip http server
!
!
line con 0
 transport input none
line aux 0
line vty 0 4
!
no scheduler allocate
end

ankara# sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, 0 - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route

Gateway of last resort is not set

 17.0.0.0/30 is subnetted, 1 subnets
C    17.1.1.16 is directly connected, Serial1
C    192.168.6.0 is directly connected, Ethernet0
 18.0.0.0/30 is subnetted, 1 subnets
C    18.1.1.16 is directly connected, Serial0
0    192.168.34.0 [110/64] via 17.1.1.17, 00:06:50, Serial1
0    192.168.28.0 [110/64] via 18.1.1.18, 00:05:11, Serial0

```

ankara#

```

!
router ospf 101
  network 192.168.28.0 0.0.0.255 area 0
  network 18.1.1.16 0.0.0.3 area 0
!

ip classless
no ip http server
!
!
!
line con 0
  transport input none
line aux 0
line vty 0 4
!
no scheduler allocate
end

giresun# sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route

Gateway of last resort is not set

  10.0.0.0/30 is subnetted, 1 subnets
C        18.1.1.16 is directly connected, Serial1
C        192.168.28.0 is directly connected, Ethernet0
  17.0.0.0/30 is subnetted, 1 subnets
O          17.1.1.16 [110/128] via 18.1.1.17, 00:06:00, Serial1
O          192.168.6.0 [110/64] via 18.1.1.17, 00:06:00, Serial1
O          192.168.34.0 [110/192] via 18.1.1.17, 00:04:00, Serial1

giresun#

```

```

istanbul#ping 192.168.6.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.6.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
istanbul#ping 192.168.28.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.28.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
istanbul#

```

```
istanbul#sh ip ospf neighbor
Neighbor ID      Pri   State            Dead Time    Address          Interface
192.168.6.1       1     FULL/           00:16:30     17.1.1.18        Serial1

istanbul#
```

```
ankara#sh ip ospf neighbor
Neighbor ID      Pri   State            Dead Time    Address          Interface
192.168.34.1      1     FULL/           00:16:03     17.1.1.17        Serial1
192.168.28.1      1     FULL/           00:16:03     18.1.1.18        Serial0

ankara#
```

```
giresun#
giresun#sh ip ospf neighbor
Neighbor ID      Pri   State            Dead Time    Address          Interface
192.168.6.1       1     FULL/           00:17:01     18.1.1.17        Serial1

giresun#
```

```
giresun#show ip ospf interface
Serial1 is up, line protocol is up
  Internet Address 18.1.1.18/30 , Area 0
  Process ID 101, Router ID 192.168.28.1, Network Type , Cost: 64
  Transmit Delay is 1 sec, State , Priority 1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:02
  Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 18.1.1.17
  Suppress hello for 0 neighbor(s)
Ethernet0 is up, line protocol is up
  Internet Address 192.168.28.1/24 , Area 0
  Process ID 101, Router ID 192.168.28.1, Network Type , Cost: 10
  Transmit Delay is 1 sec, State , Priority 1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:02
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)

giresun#
```

OSPF Özeti

- Link State bir protokoldür.

- Hizli yayılma ozelligine sahiptir.
- VLSM (Variable Length Subnet Mask) ve CIDR (Classless Inter Domain Routing) destegi vardır.
- Metric hesabi tamamen bant genisligi genisligi uzerine kuruludur.
- Distance Vector protokollerin aksine periyodik update'ler yapmaz, gerektiginde yani networkte degisiklik oldugu zaman update yapar.
- Area 0 Backbone area olarak adlandirili ve diger butun arealar ancak area 0 uzerinde birbirleriyle konusabilirler.
- Komsu Routerlarina 10 saniye araliklar ile gonderdigi Hello paketleri ile komsuluk iliskilerini baslatir devam ettirir. Non-Broadcast Multi Access (NBMA) networklerde 30 saniyedir.
- Dead Interval Hello Interval, in 4 katidir. Routerların komsuluk ilişkisi kurabilmeleri için Hello ve Dead Intervallarının aynı olması gereklidir. Hello ve Dead Interval aralıkları değiştirilebilir.

```
Rtr(config-if)# ip ospf hello-interval seconds
```

```
Rtr(config-if)# ip ospf dead-interval seconds
```

- Broadcast Multi Access ve Non-Broadcast Multi Access networklerde butun trafiği DR denen router yönetir, BDR ile yedeklenmistir. Bu networklerde Routerlar sadece birbirlerine Hello paketleri gönderirken diger butun paketler DR uzerinden gerçekleşir.
- Konfigurasyonu oldukça basittir.

```
Rtr(config)# router ospf process-id
```

```
Rtr(config-router)#network address wildcard-mask area area-id
```

- Asagidaki show komutlari ile olaylar goruntulenebilir.

```

Router# show ip route
Router# show ip ospf
Router# show ip ospf interface
Router# show ip ospf neighbor
Router# show ip ospf database
Router# debug ip ospf adj
Router# debug ip ospf events

```

Router# debug ip ospf adj

04:19:46: OSPF: Rcv hello from 201.0.0.1 area 0 from FastEthernet0 192.168.20.1
 04:19:46: OSPF: 2 Way Communication to 201.0.0.1 on FastEthernet0, **state 2WAY**
 04:19:46: OSPF: End of hello processing

04:20:22: OSPF: end of Wait on interface FastEthernet0
 04:20:22: OSPF: DR/BDR election on FastEthernet0
 04:20:22: OSPF: Elect BDR 200.0.0.1
 04:20:22: OSPF: Elect DR 200.0.0.1
 04:20:22: OSPF: Elect BDR 201.0.0.1
 04:20:22: OSPF: Elect DR 200.0.0.1
 04:20:22: DR: 201.0.0.1 (Id) BDR: 200.0.0.1 (Id)
 04:20:23: OSPF: Rcv DBD from 201.0.0.1 on FastEthernet0 seq 0x2657 opt 0x2 flag
 0x7 len 32 mtu 1500 **state EXSTART**
 04:20:23: OSPF: NBR Negotiation Done. We are the SLAVE
 04:20:23: OSPF: Send DBD to 201.0.0.1 on FastEthernet0 seq 0x2657 opt 0x2 flag 0 x2 len 92
 04:20:23: OSPF: Rcv DBD from 201.0.0.1 on FastEthernet0 seq 0x2658 opt 0x2 flag
 0x3 len 72 mtu 1500 **state EXCHANGE**
 <text omitted>
 04:20:23: OSPF: Synchronized with 201.0.0.1 on FastEthernet0, **state FULL**

Extralar

Authentication Konfigurasyonu yapılabilir.

(Basit)



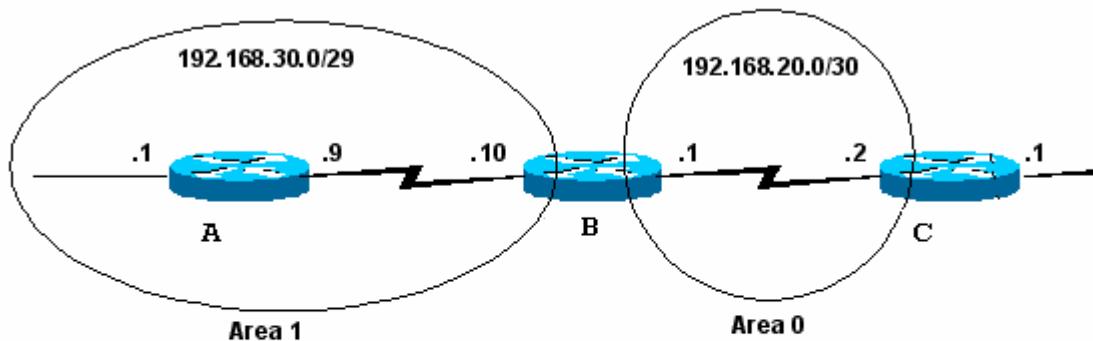
RouterA	RouterB
<pre> interface Serial1 ip address 192.16.64.1 255.255.255.0 ip ospf authentication-key secret ! router ospf 10 network 192.16.64.0 0.0.0.255 area 0 network 70.0.0.0 0.255.255.255 area 0 area 0 authentication </pre>	<pre> interface Serial2 ip address 192.16.64.2 255.255.255.0 ip ospf authentication-key secret ! router ospf 10 network 172.16.0.0 0.0.255.255 area 0 network 192.16.64.0 0.0.0.255 area 0 area 0 authentication </pre>

(MD5)



RouterA	RouterB
<pre> interface Serial1 ip address 192.16.64.1 255.255.255.0 ip ospf message-digest-key 1 md5 secret ! router ospf 10 network 192.16.64.0 0.0.0.255 area 0 network 70.0.0.0 0.255.255.255 area 0 area 0 authentication message-digest </pre>	<pre> interface Serial2 ip address 192.16.64.2 255.255.255.0 ip ospf message-digest-key 1 md5 secret ! router ospf 10 network 172.16.0.0 0.0.255.255 area 0 network 192.16.64.0 0.0.0.255 area 0 area 0 authentication message-digest </pre>

Iki farklı Area Area Borde Router (ABR) denen Routerlar ile haberleşebilirler.



B

```
router ospf 20
  network 192.168.30.0 0.0.0.255 area 1
  network 192.168.20.0 0.0.0.255 area 0
```

Burada B Routeri Area Border Router dir ve interfacelerinden biri area 0' da bir digeri area 1' dendir.

Default Route yapılabılır.

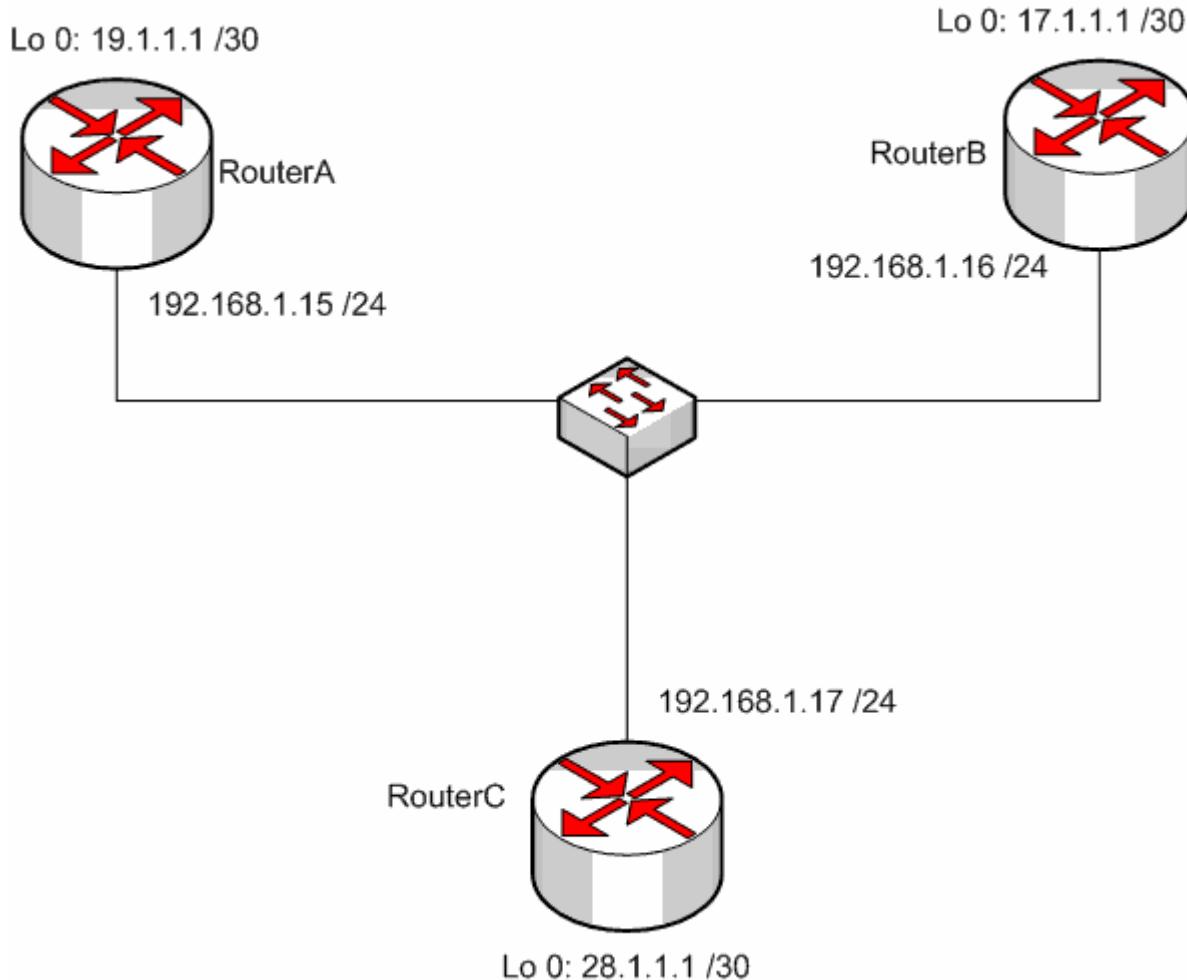
Bunun icin static default route OSPF konfigurasyonu icine gomulmelidir.

```
Router(config)# ip route 0.0.0.0 0.0.0.0 serial0
```

```
Router(config)# router ospf 1
```

```
Router(config-router)# default-information originate
```

OSPF DR-BDR Seçimi Lab. Çalışması



Bu çalışma içerisinde DR ve BDR seçimlerinin anlaşılması amaçlanmıştır. Laboratuar imkanlarının elverdiği ölçüde tasarlanan senaryo dağlı router aynı ethernet networke bağlanmış ve her Router üzerinde Loopback adresleri tanımlanmıştır.

Routerlarda OSPF konfigürasyonu yapılrken Loopback betworklerde tanıtılmıştır.

Konfigürasyon ve convergence tamamlandıktan sonra Routing Table' lar aşağıdaki gibi oluşturulmuştur.

```

RouterA#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route
  
```

Gateway of last resort is not set

```

  17.0.0.0/32 is subnetted, 1 subnets
0      17.1.1.1 [110/11] via 192.168.1.16, 00:00:11, Ethernet0/0
  19.0.0.0/30 is subnetted, 1 subnets
C      19.1.1.0 is directly connected, Loopback0
  192.168.1.0/24 is subnetted, 1 subnets
  28.0.0.0/32 is subnetted, 1 subnets
0      28.1.1.1 [110/11] via 192.168.1.17, 00:00:11, Ethernet0/0
  
```

RouterA#

70:29:40 saatlerdir | İstihdamla | QADİG R-N-1 | Kavdır | İhiwh | SAVT | Yakala | Yazdırma seçenekleri

```

RouterB#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/30 is subnetted, 1 subnets
C    17.1.1.0 is directly connected, Loopback0
  19.0.0.0/32 is subnetted, 1 subnets
O      19.1.1.1 [110/11] via 192.168.1.15, 00:02:05, Ethernet0/0
C    192.168.1.0/24 is directly connected, Ethernet0/0
  28.0.0.0/32 is subnetted, 1 subnets
O      28.1.1.1 [110/11] via 192.168.1.17, 00:02:05, Ethernet0/0
RouterB#

```

00:30:34 bağlanıldı | OtoAlgıla | 9600 8-N-1 | Kaydır | büy+ | SAYI | Yakala | Yazdırma yankısı

```

RouterC#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  17.0.0.0/32 is subnetted, 1 subnets
O      17.1.1.1 [110/11] via 192.168.1.16, 00:03:01, Ethernet0/0
  19.0.0.0/32 is subnetted, 1 subnets
O      19.1.1.1 [110/11] via 192.168.1.15, 00:03:01, Ethernet0/0
C    192.168.1.0/24 is directly connected, Ethernet0/0
  28.0.0.0/30 is subnetted, 1 subnets
C      28.1.1.0 is directly connected, Loopback0
RouterC#

```

00:31:27 bağlanıldı | OtoAlgıla | 9600 8-N-1 | Kaydır | büy+ | SAYI | Yakala | Yazdırma yankısı

Routing Table' larin ardindan OSPF database'i ve Ospf komsulari incelenmistir. Bu incelemede DR ve BDR' lar detayli gorulebilmektedir.

```
RouterA#sh ip ospf database
```

OSPF Router with ID (19.1.1.1) (Process ID 123)

Router Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
17.1.1.1	17.1.1.1	321	0x80000002	0x0043A2	2
19.1.1.1	19.1.1.1	255	0x80000004	0x0007D7	2
28.1.1.1	28.1.1.1	321	0x80000003	0x00B50D	2

Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
192.168.1.17	28.1.1.1	257	0x80000002	0x003819

```
RouterA#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
17.1.1.1	1	FULL/BDR	00:00:34	192.168.1.16	Ethernet0/0
28.1.1.1	1	FULL/DR	00:00:33	192.168.1.17	Ethernet0/0

RouterA' dan alınan bu goruntude komsu routerlar ve bu routerlar ile olan iliski tespit edilebilmektedir. Ornegin 28.1.1.1 ID' sine sahip Router ile Full komsuluk iliskisi kurulmus ve DR olarak kabul edilmistir. (Bunun boyle olacagini zaten biliyorduk zira 28.1.1.1 ortamdaki en yüksek ID)

```
RouterB#show ip ospf database
```

OSPF Router with ID (17.1.1.1) (Process ID 123)

Router Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
17.1.1.1	17.1.1.1	383	0x80000002	0x0043A2	2
19.1.1.1	19.1.1.1	318	0x80000004	0x0007D7	2
28.1.1.1	28.1.1.1	383	0x80000003	0x00B50D	2

Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
192.168.1.17	28.1.1.1	319	0x80000002	0x003819

```
RouterB#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
19.1.1.1	1	FULL/DROTHER	00:00:36	192.168.1.15	Ethernet0/0
28.1.1.1	1	FULL/DR	00:00:37	192.168.1.17	Ethernet0/0

0:33:38 başlanıldı | OtoArama | 9600 8-N-1 | Kaydır | büyüh | SAYI | Yakala | Yazdırma yankısı

```
RouterC#show ip ospf database
OSPF Router with ID (28.1.1.1) (Process ID 123)
```

Router Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
17.1.1.1	17.1.1.1	435	0x80000002	0x43A2	2
19.1.1.1	19.1.1.1	370	0x80000004	0x7D7	2
28.1.1.1	28.1.1.1	435	0x80000003	0xB50D	2

Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
192.168.1.17	28.1.1.1	370	0x80000002	0x3819

RouterC#show ip ospf neighbor

Neighbor ID	Pri	State	Dead Time	Address	Interface
19.1.1.1	1	FULL/DROTHER	00:00:31	192.168.1.15	Ethernet0/0
17.1.1.1	1	FULL/BDR	00:00:32	192.168.1.16	Ethernet0/0

RouterC#

```
RouterA#sh ip ospf interface
Loopback0 is up, line protocol is up
  Internet Address 19.1.1.1/30, Area 0
    Process ID 123, Router ID 19.1.1.1, Network Type LOOPBACK, Cost: 1
    Loopback interface is treated as a stub Host
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.1.15/24, Area 0
    Process ID 123, Router ID 19.1.1.1, Network Type BROADCAST, Cost: 10
    Transmit Delay is 1 sec, State DROTHER, Priority 1
    Designated Router (ID) 28.1.1.1, Interface address 192.168.1.17
    Backup Designated router (ID) 17.1.1.1, Interface address 192.168.1.16
    Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
      oob-resync timeout 40
      Hello due in 00:00:05
    Index 1/1, flood queue length 0
    Next 0x0(0)/0x0(0)
    Last flood scan length is 0, maximum is 1
    Last flood scan time is 4 msec, maximum is 4 msec
    Neighbor Count is 2, Adjacent neighbor count is 2
      Adjacent with neighbor 17.1.1.1 (Backup Designated Router)
      Adjacent with neighbor 28.1.1.1 (Designated Router)
    Suppress hello for 0 neighbor(s)
RouterA#
```

RouterA#

```

RouterB#sh ip ospf interface
Loopback0 is up, line protocol is up
  Internet Address 17.1.1.1/30, Area 0
    Process ID 123, Router ID 17.1.1.1, Network Type LOOPBACK, Cost: 1
    Loopback interface is treated as a stub Host
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.1.16/24, Area 0
    Process ID 123, Router ID 17.1.1.1, Network Type BROADCAST, Cost: 10
    Transmit Delay is 1 sec, State BDR, Priority 1
    Designated Router (ID) 28.1.1.1, Interface address 192.168.1.17
    Backup Designated router (ID) 17.1.1.1, Interface address 192.168.1.16
    Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
      oob-resync timeout 40
      Hello due in 00:00:07
    Index 1/1, flood queue length 0
    Next 0x0(0)/0x0(0)
    Last flood scan length is 0, maximum is 1
    Last flood scan time is 0 msec, maximum is 0 msec
    Neighbor Count is 2, Adjacent neighbor count is 2
      Adjacent with neighbor 19.1.1.1
      Adjacent with neighbor 28.1.1.1 (Designated Router)
    Suppress hello for 0 neighbor(s)
RouterB#_

```

00:36:30 bağlandı | OtoAlqila | 9600 8-N-1 | Kaydır | büyh | SAYI | Yakala | Yazdırma yankısı

```

RouterC#show ip ospf interface
Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.1.17/24, Area 0
    Process ID 123, Router ID 28.1.1.1, Network Type BROADCAST, Cost: 10
    Transmit Delay is 1 sec, State DR, Priority 1
    Designated Router (ID) 28.1.1.1, Interface address 192.168.1.17
    Backup Designated router (ID) 17.1.1.1, Interface address 192.168.1.16
    Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
      Hello due in 00:00:05
    Index 1/1, flood queue length 0
    Next 0x0(0)/0x0(0)
    Last flood scan length is 1, maximum is 2
    Last flood scan time is 0 msec, maximum is 0 msec
    Neighbor Count is 2, Adjacent neighbor count is 2
      Adjacent with neighbor 19.1.1.1
      Adjacent with neighbor 17.1.1.1 (Backup Designated Router)
    Suppress hello for 0 neighbor(s)
Loopback0 is up, line protocol is up
  Internet Address 28.1.1.1/30, Area 0
    Process ID 123, Router ID 28.1.1.1, Network Type LOOPBACK, Cost: 1
    Loopback interface is treated as a stub Host
RouterC#

```

00:37:45 bağlandı | OtoAlqila | 9600 8-N-1 | Kaydır | büyh | SAYI | Yakala | Yazdırma yankısı

Show ip ospf interface komutu ile aldığımız görüntülere baktığımızda 17.1.1.1 ID'li routerin BDR seçildiğini olsa da yüksek ID'ye sahip 19.1.1.1 ID'li routerin DROther olarak kaldığını görüyoruz ki bu karmaşık bir durum.

OSPF konusun routerlar ortama daha yüksek ID'ye sahip bir Router katıldığında onu Drother olarak alırlar, yeniden bir DR – BDR seçime gitmezler. Bu Cisco'nun bir bug'udur. Anlaşılan o ki örneğimizde 17.1.1.1 ID'li router ortama daha önce katılmış ve 19.1.1.1 ID'li router up olmadan BDR seçmimi tamamlanmış.

Cisco'nun bu bug'ini aşmak için interfacelerden en azından birini down – up yapmamız gerekecek.

```
Router#debug ip ospf adj
00:50:38: OSPF: DR/BDR election on Ethernet0/0
00:50:38: OSPF: Elect DR 28.1.1.1
00:50:38: OSPF: Elect BDR 0.0.0.0
00:50:38:     DR: 28.1.1.1 (Id) BDR: none
00:50:38: OSPF: Remember old DR 17.1.1.1 (id)
00:50:39: OSPF: Reset old DR on Ethernet0/0
00:50:39: OSPF: Build router LSA for area 0, router ID 28.1.1.1, seq 0x80000010
00:50:39: OSPF: No full nbrs to build Net Lsa for interface Ethernet0/0
00:50:46: OSPF: 2 Way Communication to 19.1.1.1 on Ethernet0/0, state 2WAY
00:50:46: OSPF: Neighbor change Event on interface Ethernet0/0
00:50:46: OSPF: DR/BDR election on Ethernet0/0
00:50:46: OSPF: Elect BDR 19.1.1.1
00:50:46: OSPF: Elect DR 28.1.1.1
00:50:46:     DR: 28.1.1.1 (Id) BDR: 19.1.1.1 (Id)
00:50:46: OSPF: Send DBD to 19.1.1.1 on Ethernet0/0 seq 0x1692 opt 0x42 flag 0x7 len 32
00:50:46: OSPF: Rcv DBD from 19.1.1.1 on Ethernet0/0 seq 0x1692 opt 0x52 flag 0x2 len 112
mtu 1500 state EXSTART
00:50:46: OSPF: NBR Negotiation Done. We are the MASTER
00:50:46: OSPF: Send DBD to 19.1.1.1 on Ethernet0/0 seq 0x1693 opt 0x42 flag 0x3 len 112
00:50:46: OSPF: Database request to 19.1.1.1
00:50:46: OSPF: sent LS REQ packet to 192.168.1.15, length 12
00:50:46: OSPF: Rcv DBD from 19.1.1.1 on Ethernet0/0 seq 0x1693 opt 0x52 flag 0x0 len 32
mtu 1500 state EXCHANGE
00:50:46: OSPF: Send DBD to 19.1.1.1 on Ethernet0/0 seq 0x1694 opt 0x42 flag 0x1 len 32
00:50:46: OSPF: Rcv DBD from 19.1.1.1 on Ethernet0/0 seq 0x1694 opt 0x52 flag 0x0 len 32
mtu 1500 state EXCHANGE
00:50:46: OSPF: Exchange Done with 19.1.1.1 on Ethernet0/0
00:50:46: OSPF: Synchronized with 19.1.1.1 on Ethernet0/0, state FULL
00:50:46: %OSPF-5-ADJCHG: Process 123, Nbr 19.1.1.1 on Ethernet0/0 from LOADING to FULL,
Loading Done
00:50:46: OSPF: Build router LSA for area 0, router ID 28.1.1.1, seq 0x80000011

00:50:46: OSPF: Build network LSA for Ethernet0/0, router ID 28.1.1.1
00:50:48: OSPF: Rcv DBD from 17.1.1.1 on Ethernet0/0 seq 0x1C03 opt 0x52 flag 0x7 len 32
mtu 1500 state INIT
00:50:48: OSPF: 2 Way Communication to 17.1.1.1 on Ethernet0/0, state 2WAY
00:50:48: OSPF: Neighbor change Event on interface Ethernet0/0
00:50:48: OSPF: DR/BDR election on Ethernet0/0
00:50:48: OSPF: Elect BDR 19.1.1.1
00:50:48: OSPF: Elect DR 28.1.1.1
00:50:48:     DR: 28.1.1.1 (Id) BDR: 19.1.1.1 (Id)
00:50:48: OSPF: Send DBD to 17.1.1.1 on Ethernet0/0 seq 0xB19 opt 0x42 flag 0x7 len 32
00:50:48: OSPF: First DBD and we are not SLAVE
00:50:48: OSPF: Rcv DBD from 17.1.1.1 on Ethernet0/0 seq 0xB19 opt 0x52 flag 0x2 len 92
mtu 1500 state EXSTART
00:50:48: OSPF: NBR Negotiation Done. We are the MASTER
00:50:48: OSPF: Send DBD to 17.1.1.1 on Ethernet0/0 seq 0xB1A opt 0x42 flag 0x3 len 132
00:50:48: OSPF: Database request to 17.1.1.1
```

```

00:50:48: OSPF: sent LS REQ packet to 192.168.1.16, length 12
00:50:48: OSPF: Rcv DBD from 17.1.1.1 on Ethernet0/0 seq 0xB1A opt 0x52 flag 0x0 len 32
mtu 1500 state EXCHANGE
00:50:48: OSPF: Send DBD to 17.1.1.1 on Ethernet0/0 seq 0xB1B opt 0x42 flag 0x1 len 32
00:50:48: OSPF: Rcv DBD from 17.1.1.1 on Ethernet0/0 seq 0xB1B opt 0x52 flag 0x0 len 32
mtu 1500 state EXCHANGE
00:50:48: OSPF: Exchange Done with 17.1.1.1 on Ethernet0/0
00:50:48: OSPF: Synchronized with 17.1.1.1 on Ethernet0/0, state FULL
00:50:48: %OSPF-5-ADJCHG: Process 123, Nbr 17.1.1.1 on Ethernet0/0 from LOADING to FULL,
Loading Done
00:50:48: OSPF: Neighbor change Event on interface Ethernet0/0
00:50:48: OSPF: DR/BDR election on Ethernet0/0
00:50:48: OSPF: Elect BDR 19.1.1.1
00:50:48: OSPF: Elect DR 28.1.1.1
00:50:48: DR: 28.1.1.1 (Id) BDR: 19.1.1.1 (Id)
00:50:52: OSPF: Build network LSA for Ethernet0/0, router ID 28.1.1.1
RouterC(config)#

```

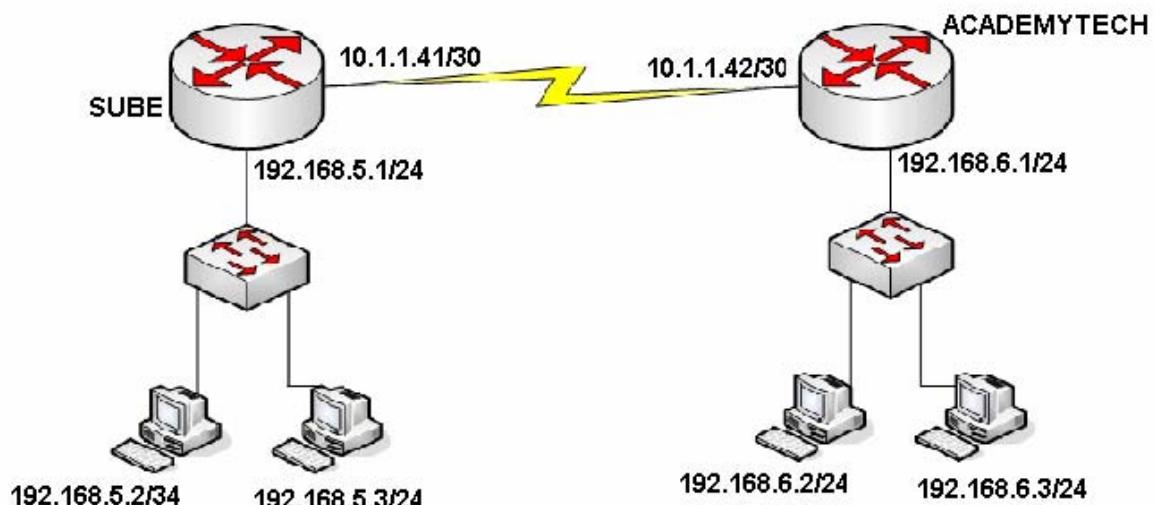
Routing Protokollere Genel Bakış

Bu bölümde Routing Protokollerini genel olarak inceleyeceğiz. Routing Protokollerini genel olarak üç grup halinde inceleyebiliriz.

Distance Vector Protokoller	Link State Protokoller	Hybrid Protokoller
RIP	OSPF	EIGRP
RIPv2		
IGRP		

Bütün Routing Protokollerin anlatımı sırasında hep söylediğimiz gibi, Routing Protokoller update mantığıyla daha açık bir ifadeyle sahip oldukları veritabanlarını (Routing Table) paylaşarak çalışırlar.

Biz sadece Routerlarımızın kendilerine direk bağlı olan networkleri protokoller vasıtasıyla tanıtırız. Protokol cinsine göre, belli zaman aralıklarından Routerlar arasında veritabanı paylaşımı gerçekleşir ve bir süre sonra bütün Routerlar sisteme deki bütün networkleri öğrenmiş olarak Routing Table'larını son haliyle oluştururlar.



Routing Protokollerin karşılaştırılması sırasında şekildeki topolojiden hareketle konfigürasyonlar yapılacak ve karşılaşmalar gerçekleştirilecektir.

Routerlar en iyi yol seçimi yaparken (Best Path Determination) referans olarak Routing Table'larında ki bilgileri alırlar. Dolayısıyla Routing Protokoller kullanarak oluşturulan Routing Table'ların sistem başladıkta belirli bir zaman sonra son halini alacak olması bir dezavantaj olarak görülebilir. Bunun yanında networklerin giderek büyüdükleri göz önüne alınırsa bir kez Routing Protokoller ile konfigüre ettiğimiz Routerlar ileride eklenecek networkleri biz müdahale etmeden öğrenebileceklerdir ki buda önemli avantajlarındandır.

Burada Update sürelerini baz alarak Routing Protokollerini karşılaştırabiliriz.

RIP	RIPv2	IGRP	EIGRP	OSPF
30 sn.	30 sn.	90 sn.	Gerektiğinde	Gerektiğinde

Rip, RIPv2 ve IGRP' de update'ler belirli zaman aralıklarında yapılırken, EIGRP ve OSPF için update gerektiğinde yani sistem üzerinde bir değişiklik olduğunda, yeni bir network eklendiğinde veya bir network down olduğunda yapılır. Ve burada yine aklımızda tutmamız gereken konu EIGRP ve OSPF gerektiğinde yaptığı update'lerde sadece değişen durum ile ilgili bilgi gönderirken diğerleri tüm Routing Table' larını her seferinde gönderirler. Bunun yanında EIGRP 5 ve OSPF 10 saniye aralıklarla komşu routerlarının up olup olmadıklarını kontrol etmek için küçük paketler gönderirler fakat bunlar hattı çok az meşgul ederler.

Burada Routing Protokollerin update yaparken kullandıkları broadcast ya da multicast adreslerde karşılaştırılabilir. Hatırlayacağınız gibi "debug" komutunu kullanarak protokollerin aldıkları veya gönderdikleri paketleri izleyebiliyoruz.

```
ACADEMYTECH#debug ip rip
RIP protocol debugging is on
ACADEMYTECH#
00:29:02: RIP: sending v1 update to 255.255.255.255 via Ethernet0/0 (192.168.6.1)
)
00:29:02:      network 192.168.5.0, metric 2
00:29:02:      network 10.0.0.0, metric 1
00:29:02: RIP: sending v1 update to 255.255.255.255 via Serial0/0 (10.1.1.42)
00:29:02:      network 192.168.6.0, metric 1
```

```
ACADEMYTECH#debug ip igrp events
IGRP event debugging is on
ACADEMYTECH#
00:33:35: IGRP: sending update to 255.255.255.255 via Ethernet0/0 (192.168.6.1)
00:33:35: IGRP: Update contains 0 interior, 2 system, and 0 exterior routes.
00:33:35: IGRP: Total routes in update: 2
00:33:35: IGRP: sending update to 255.255.255.255 via Serial0/0 (10.1.1.42)
00:33:35: IGRP: Update contains 0 interior, 1 system, and 0 exterior routes.
00:33:35: IGRP: Total routes in update: 1
```

0:18:39 bağlanıldı | OtoAlgıla | 9600 8-N-1 | Kaydır | büyhäuser | SAYI | Yakala | Yazdırma yankısı |

```
ACADEMYTECH#debug ip ospf events
OSPF events debugging is on
ACADEMYTECH#
00:43:30: OSPF: Rcv hello from 192.168.5.1 area 0 from Serial0/0 10.1.1.41
00:43:30: OSPF: End of hello processing
00:43:34: OSPF: Rcv pkt from 192.168.5.1, Ethernet0/0, area 0.0.0.0 : src not on
the same network
00:43:40: OSPF: Rcv hello from 192.168.5.1 area 0 from Serial0/0 10.1.1.41
00:43:40: OSPF: End of hello processing
```

0:28:41 bağlanıldı | OtoAlgıla | 9600 8-N-1 | Kaydır | büyhäuser | SAYI | Yakala | Yazdırma yankısı |
(OSPF Hello paketleri)

Routing Protokollerden bahsederken bahsettiğimiz konulardan biri de bazı protokollerin VLSM (Variable Length Subnet Mask) desteği verirken bazlarının vermemesi idi. Bundan kastettiğimiz şey protokollerin Classless veya Classfull olmalarıdır. Anladığınız gibi Classfull bir protokolde kullanacağımız network adreslerinde Subnet maski biz belirleyemeyiz, protokol o adresin ait olduğu sınıfa göre Subnet maskini kabul eder. Classless protokollerde ise Subnet mask tamamen bizim kontrolümüzdedir.

VLSM Desteği	Rip	Ripv2	IGRP	OSPF	EIGRP
Yok	Var	Yok		Var	Var

```

no ip directed-broadcast
no ip mroute-cache
no fair-queue
!
interface TokenRing0/0
no ip address
no ip directed-broadcast
shutdown
ring-speed 16
!
interface Serial0/1
no ip address
no ip directed-broadcast
shutdown
!
router ospf 101
network 10.1.1.40 0.0.0.3 area 0
network 192.168.6.0 0.0.0.255 area 0
!
ip classless
!
!
line con 0
--More--

```

(OSPF Runnin-Config, VLSM Desteği)

Burada Rip ve IGRP kullanırken örneğin 10.1.1.0 /30 gibi bir network tanımlamamız mümkün değildir. Bu protokoller söz konusu adres A sınıfı olduğu için Subnet maski 255.0.0.0 olarak kabul edeceklerdir.

Routing Protokoller metric hesaplarında farklı kritelere bakarlar. Rip tamamen hop sayısına bakarken IGRP bahsettiğimiz K1'den K5'e kadar olan değerlere büyük ölçüde bant genişliğini baz alarak bakar. Tipki değişik kriterlere göre metric hesabı yapıldığı gibi Routing Protokollerin çalışacakları maksimum hop sayıları da farklı farklıdır.

	Rip	Ripv2	IGRP	OSPF	EIGRP
Metric Hesabı	Hop	Hop	K1-K5	Bantwidth	K1, K2
Max. Hop Sayısı	15	15	255	Sınırsız	224

Routing protokoller Autonomous System numaralar kullanıp kullanmadıkları ve bir Area mantığı içine girerek hiyerarşik bir yapı oluşturup oluşturmadıklarına göre de incelenebilirler.

	Rip	Ripv2	IGRP	OSPF	EIGRP
Autonomous System	Yok	Yok	Var	Var	Var
Area	Yok	Yok	Yok	Var	Yok

Routing protokollerde Administrative Distance ve metric hesaplarında, Routing Table'larına bakıldığından detaylı bilgi sahibi olunabilir.

```
ACADEMYTECH#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route, o - ODR

Gateway of last resort is not set

R    192.168.5.0/24 [120/1] via 10.1.1.41, 00:00:05, Serial0/0
      10.0.0.0/30 is subnetted, 1 subnets
C      10.1.1.40 is directly connected, Serial0/0
C      192.168.6.0/24 is directly connected, Ethernet0/0
ACADEMYTECH#
```

Rip için Routing Table görüntülendiğinde Administrative Distance'ının 120 olduğu görülmektedir. Parantez içerisindeki bir diğer ifade ("1") metriği yani Rip için hop sayısını belirtmektedir. Routing Table'da Rip protokolüyle öğrenilen Networkler "R" harfi ile belirtilirler.

```
ACADEMYTECH#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route, o - ODR

Gateway of last resort is not set

I    192.168.5.0/24 [100/80225] via 10.1.1.41, 00:00:01, Serial0/0
      10.0.0.0/30 is subnetted, 1 subnets
C      10.1.1.40 is directly connected, Serial0/0
C      192.168.6.0/24 is directly connected, Ethernet0/0
ACADEMYTECH#
```

Igrp için Routing Table görüntülendiğinde Administrative Distance'ının 100 olduğu görülmektedir. Parantez içerisindeki bir diğer ifade ("80225") metriği belirtir. K1'den K5'e kadar olan kriterler baz alınarak hesaplanmıştır.

OSPF ve EIGRP Routing Table'ları aşağıdadır.

```
ACADEMYTECH#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route, o - ODR

Gateway of last resort is not set

D  192.168.5.0/24 [90/20537600] via 10.1.1.41, 00:01:57, Serial0/0
  10.0.0.0/30 is subnetted, 1 subnets
C    10.1.1.40 is directly connected, Serial0/0
C    192.168.6.0/24 is directly connected, Ethernet0/0
ACADEMYTECH#
```

0:25:55 bağlandı OtoAlgıla 9600 8-N-1 Kavdır büyüh SAYI Yakala Yazdırma yankısı
(D: EIGRP, Administrative Distance=90)

```
ACADEMYTECH#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route, o - ODR

Gateway of last resort is not set

O  192.168.5.0/24 [110/7911] via 10.1.1.41, 00:01:03, Serial0/0
  10.0.0.0/30 is subnetted, 1 subnets
C    10.1.1.40 is directly connected, Serial0/0
C    192.168.6.0/24 is directly connected, Ethernet0/0
ACADEMYTECH#
```

1:29:12 hizlanıldı OtoAlgıla 9600 8-N-1 Kavdır büyüh SAYI Yakala Yazdırma yankısı
(O: OSPF, Administrative Distance=110)

Cisco Özel Protokoller

IGRP ve EIGRP Cisco özel protokollerdir. Diğer bütün protokoller ise publicdir. Dolayısıyla sistemimizde Cisco dışında üreticilere ait Router' larda varsa IGRP ve EIGRP bizim için doğru seçim olmayacağındır.

IGRP ve EIGRP Cisco tarafından üretilenlerden aynı AS içinde birbirleriyle haberleşebilirler. Fakat bu durumda sistemin IGRP konusunda network bilgiler EIGRP konusunda Routerlarda External EIGRP olarak etiketlenir ve bu networkler için Administrative Distance 170'dir.

IPX- AppleTalk Desteği

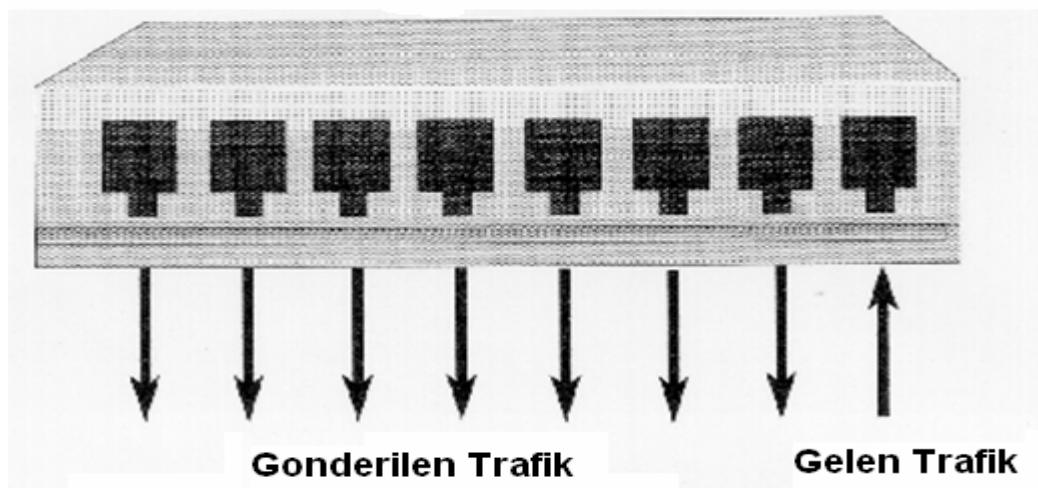
Cisco özel bir protokol olan EIGRP之外 Cisco'ya IPX ve AppleTalk networklerini de desteklemesiyle diğer protokollerden ayrılabılır.

Layer 2 Switching

Ben Switch gordum.



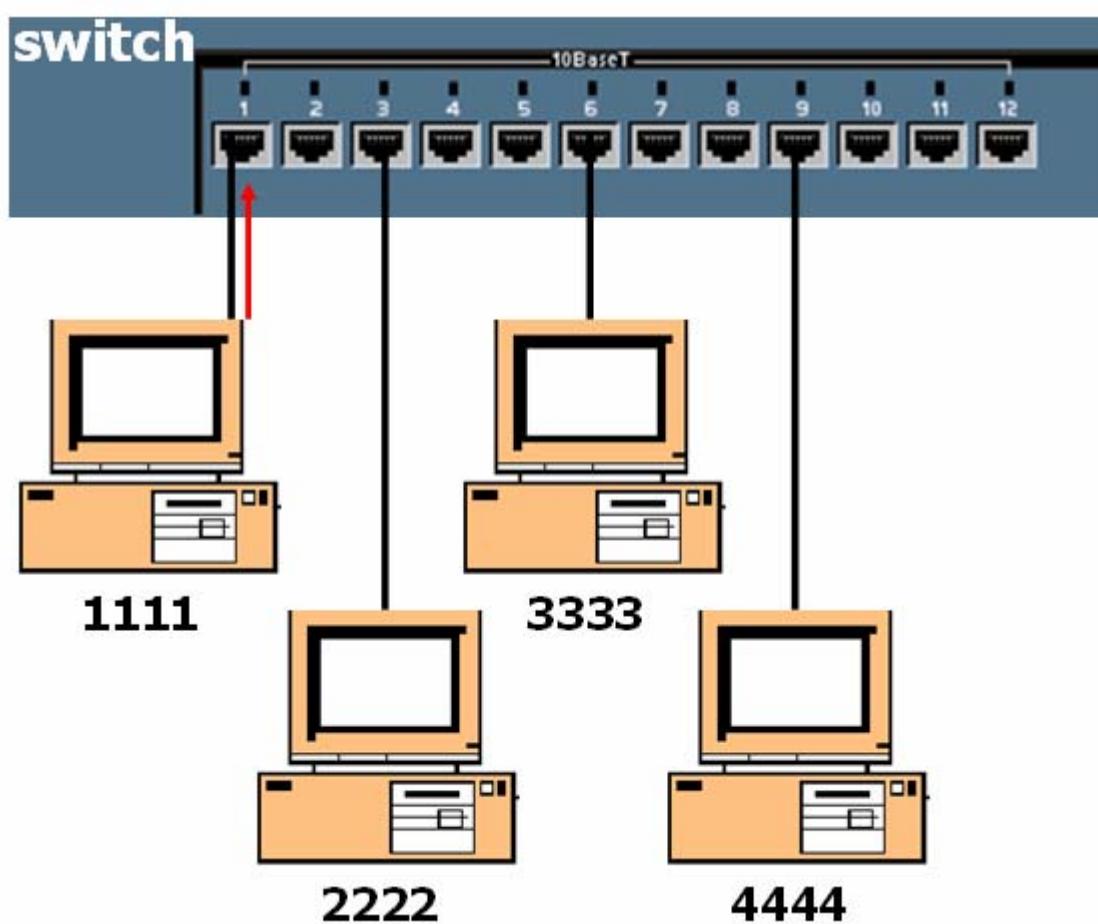
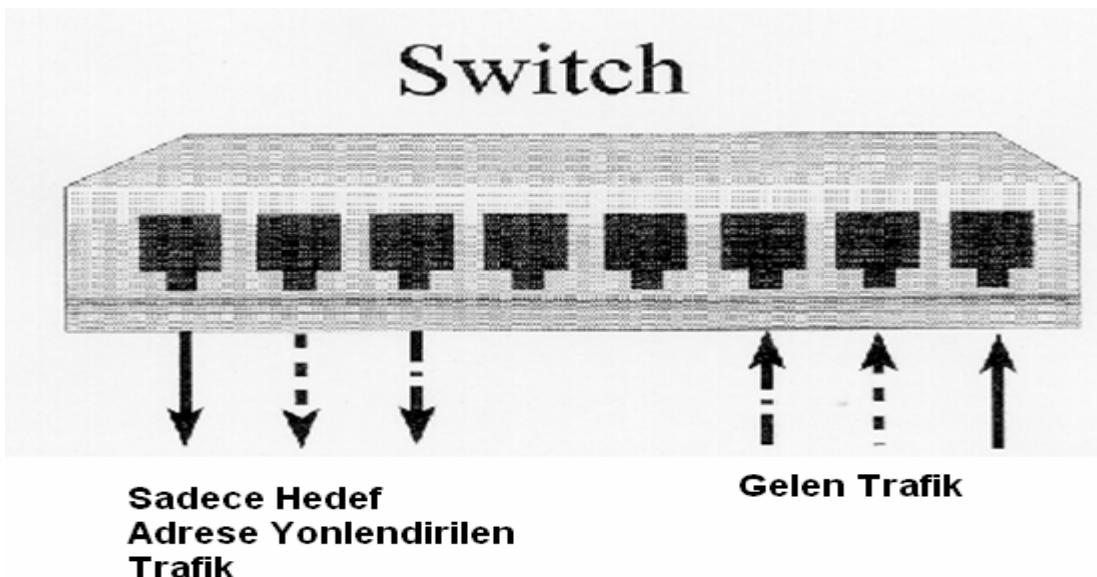
Hub'lar ile çalıştığımızda, hub aslında çok portlu bir repeater olduğu için ağdaki tüm bilgisayarlar aynı çakışma alanı içinde olacaklardır. Bu alana "collision domain" denire. Dolayısıyla network performansında düşme olacaktır.

Hub veya Repeater

Bu problemin çözümü olarak networklerde Switch adı verilen cihazlar kullanılmıştır.

Switch OSI 2. katmanda yani Data Link Layer Katmanında çalışır. Bir portuna bağlı bilgisayar veya bilgisayarları gönderdikleri帧elerden source MAC adreslerini okuyarak tanır. Bir portundan gelen veri paketini hub'lar gibi tüm portlara dağıtmak yerine sadece veri paketi üzerinde yazan "alıcı MAC adresine" sahip portuna yollar. Paketler direk hedefe gönderildiği için de network üzerinde çarpışmalar (collision) meydana gelmez.

Switch



Sekildeki yapı sistemin yeni baslatildigini varsayarsa. Switch 1111 MAC adresine sahip bilgisayarın gönderdiği frame' I alacak ve buradaki source ma adresi alanından okuduğu değeri ma adresi tablosuna yazacak.

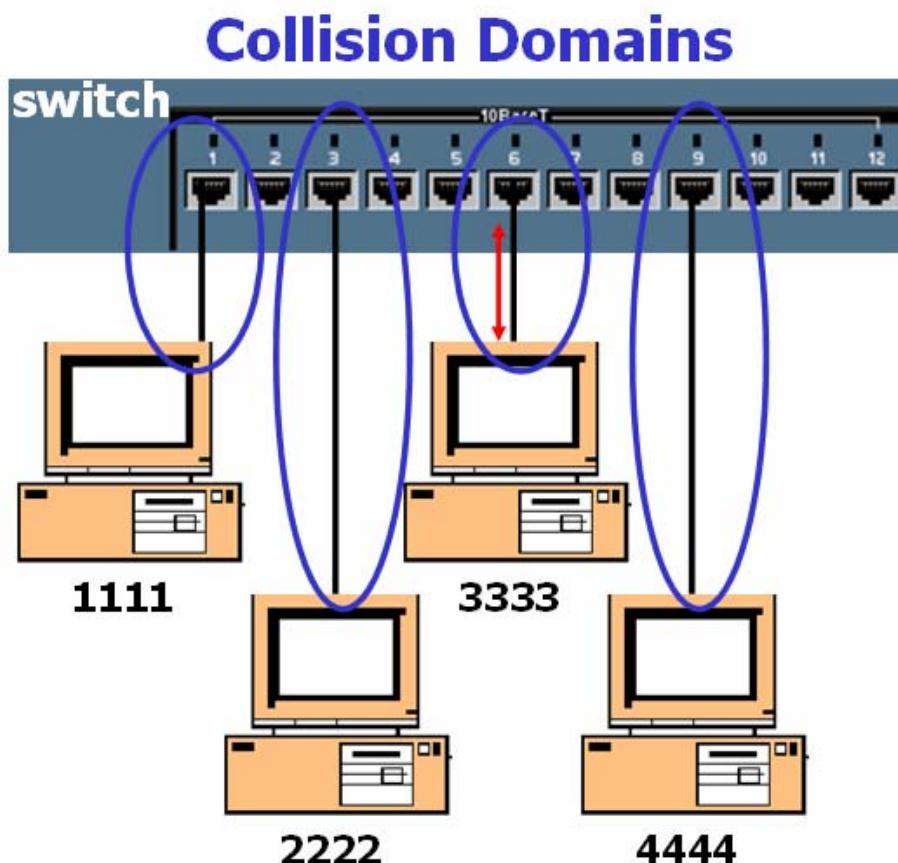
Sü an için MAC adresi tablosunda ki tek girdi 1111 MAC adresi olduğu için hedef mac adresinin hangi porta olduğunu bilmemelimizi söyleyebiliriz. Bu durumda frame butun portlardan flood edilecektir.

Bu şekilde zamanla switch butun portlarda ki bilgisayarların mac adreslerini onlari cache' in 300 saniye tutmak üzere mac adresi tablosuna yazacaktır. Artık hedef MAC adreslerine göre帧eleri yönlendirebilir.

Switch 300 saniye boyunca ilgili MAC adresinin bulunduğu porttan bir istek gelmez ise o adres tablodan silinecektir.

Simdi sözgelimi 3333 mac adresine sahip bilgisayardan 1111 mac adresine sahip bilgisayar bir istek gönderdiğini varsayıyalım. Bu durumda switch mac adresi tablosundan 1111 mac adresi bilgisayarın 1. portunda olduğunu group isteği sadece o porta fondaerecektir. Bu sayede olası collision'lar engellenmiştir.

Iste bu sebeple Switch'in her bir portu bir Collision Domain' dir denilebilir.



Switch Konfigurasyonu

Switch konfigurasyonu bir çok yonde Router ile aynıdır. Switch açıldığında user moddadır ve 'enable' yazılarla Enable moda geçilebilir.

```
Switch>enable
Switch#
```

Tipki Routerda olduğu gibi Switch de yaptığımız konfigurasyonları görüntüleyip sorun cozmemizde bize yardımcı olacak show komutları vardır.

```
Switch#show running-config
Building configuration...

Current configuration:
!
version 12.0
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Switch
!
!<OUTPUT OMITTED>
!
```

Switch# show vlan					
VLAN Name	Status Ports				
1 default	active	Fa0/1,	Fa0/2,	Fa0/3,	Fa0/4,
		Fa0/5,	Fa0/6,	Fa0/7,	Fa0/8,
		Fa0/9,	Fa0/10,	Fa0/11,	Fa0/12
1002 fddi-default	active				
1003 token-ring-default	active				
1004 fddinet-default	active				
1005 trnet-default	active				
VLAN	Type	SAID	MTU	Parent	RingNo BridgeNo

```

Switch>enable
Switch#conf t
Switch(config)#line con 0
Switch(config-line)#password ozcan
Switch(config-line)#login
Switch(config-line)#exit

```

```

Switch(config)#line vty 0 4
Switch(config-line)#password ozcan
Switch(config-line)#login
Switch(config-line)#exit

```

VLAN 1 yönetim VLAN'ıdır ve switch için verilecek IP adresi bu VLAN'da, default gateway adresi Global Configuration modda verilmelidir. 1900 serisi switchlerde ise durum biraz farklıdır.

```

Switch(config)#interface vlan 1
Switch(config-if)#ip address 192.168.1.10 255.255.255.0
Switch(config-if)#exit
Switch(config)#ip default-gateway 192.168.1.1

```

1900 serisi switchlerde ise durum biraz farklıdır. Bu switchlerde IP adresi ve default gateway adresi Global Configuration modda verilir.

```

Switch(config)#ip address 192.168.1.10 255.255.255.0
Switch(config)#ip default-gateway 192.168.1.1

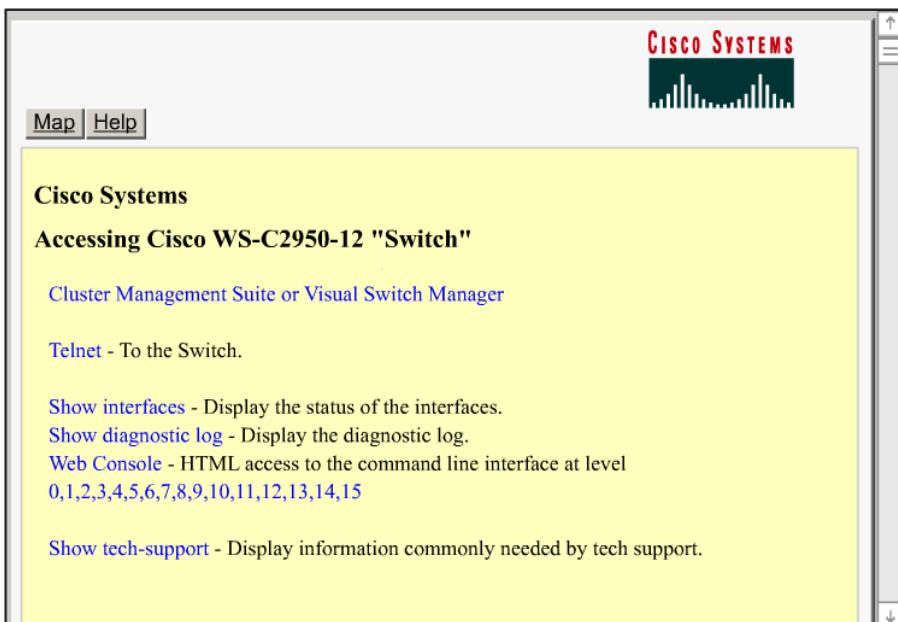
```

Switch'i web browser ile erişilebilir. Bunun için su konfigürasyon yapılmalıdır.

```

Switch#configure terminal
Enter configuration commands, one per line. End
with CNTL/Z.
Switch(config)#ip http ?
  access-class      Restrict access by access-class
  authentication    Set http authentication method
  path              Set base path for HTML
  port              HTTP port
  server            Enable HTTP server
Switch(config)#ip http server
Switch(config)#ip http port ?
  <0-65535>  HTTP port
Switch(config)#ip http port 80
Switch(config)#

```



MAC Address Table

Daha önce de belirttiğimiz gibi switchler alındıkları frameelerdeki source mac adres alanında ki bilgiler ile MAC adres tablolarını oluştururlar ve bu tablolara göre frameeleri filtrelerler.

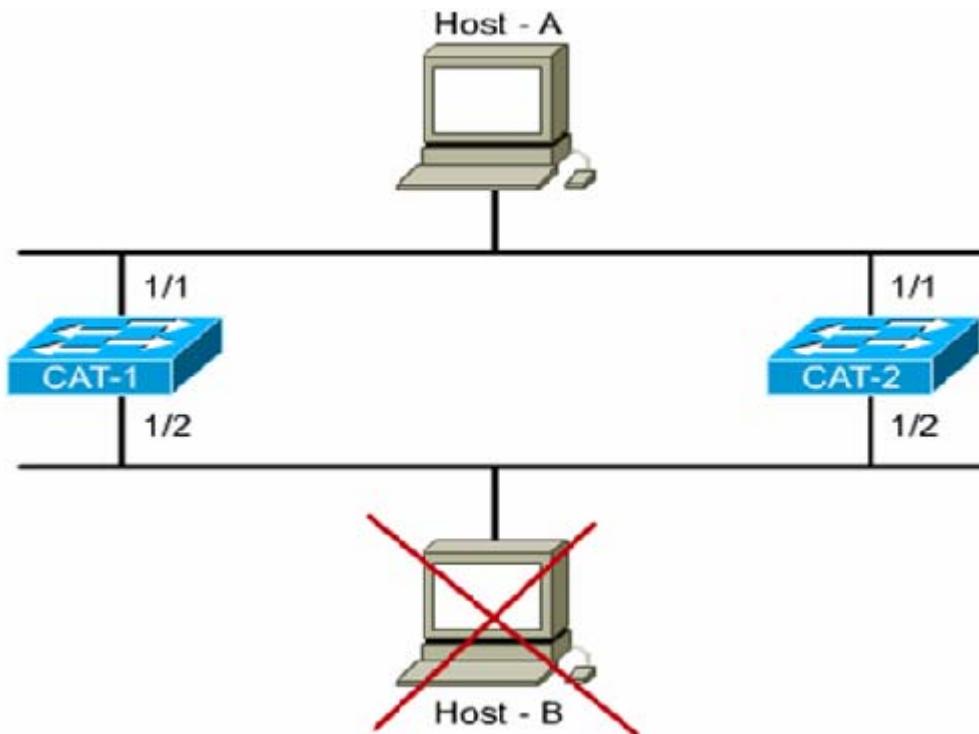
Bununla birlikte switchlere MAC adreslerinin static olarak atanması da mümkündür. Güvenliği artırmak için yapılabilecek bu uygulamayla aynı zamanda 300 saniyelik max. Age süresi de geçersiz olacaktır.

```
Switch(config)#mac-address-table static
0010.7a60.1884 interface FastEthernet0/5 VLAN1
Switch(config)#no mac-address-table static
0010.7a60.1884 interface FastEthernet0/5 VLAN1
```

Spanning Tree Protocol (STP)

STP Layer 2 cihazların haberleşmesi sırasında doğabilecek olası döngüler (loop) onleyen bir protokoldür. STP yapısı gereği kullandığı algoritma (Spanning Tree Algorithm) ile döngülere neden olmayacak bir topoloji oluşturur.

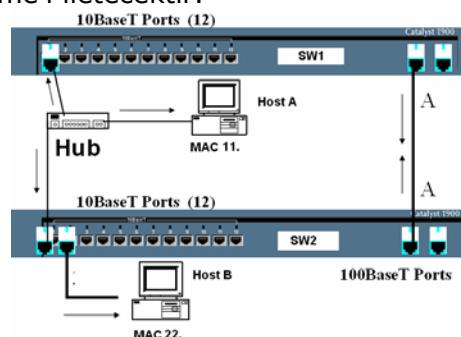
Ethernet Frame'leri TTL alanına sahip olmadıkları için STP ve loop yaratmayacak bir dizayn önemlidir. Aksi takdirde olusacak döngüler switch kapatılana kadar devam edecektir.



Sekildeki gibi bir yapıda, switchler kendilerine gelen ve hedefi bilinmeyen paketleri diğer bütün portlardan flood edeceklerine göre ciddi sorunlar yaşanacaktır. 1/1 portlarından frame'ı alan her iki switch de flood edecek ve hemen sonrasında yine her iki switch aynı frame'leri 1/2 portlarından alacak, devamında neler olacağını kestirebiliyorsunuzdur sanırım ☺

Hattları daha da zor durumda bırakacak paketler ise Broadcastlerdir. Bilindiği gibi switchler Broadcast geçirirler, dolayısıyla kendilerine gelen broadcastları bütün portlarına gönderirler.

Aşağıdaki şekilde hareketle Host A'nın söz gelimi bir ARP Request'ı bulunduğunu varsayıyalım. ARP Request帧leri broadcast olduğundan 1. portlarından bu帧leri alan her iki switchde diğer bütün portlarından bu frame'i ileticektir.



Her iki switch arasında ki A ile gösterilmis bağlantıdan da broadcastlar yayınlanacak dolayısıyla her iki switch de bu broadcast frame leri bu kez farklı portlardan olmak üzere, yeniden alacaklar ve yeniden flood edecekler. (Bu durum Broadcast Storm olarak bilinir.)

Bu vebunun gibi bir çok nedenle doğabilecek sorunlar için yardımımıza STP koşacaktır ☺

STP nin amacı genel olarak networklerdeki olası loop ları önlemek ve bunun için de her hedefe sadece bir yolun aktif olarak çalışmasını sağlamaktır. Bunun içinde STP Spanning Tree Algorithm'ı (STA) kullanır.

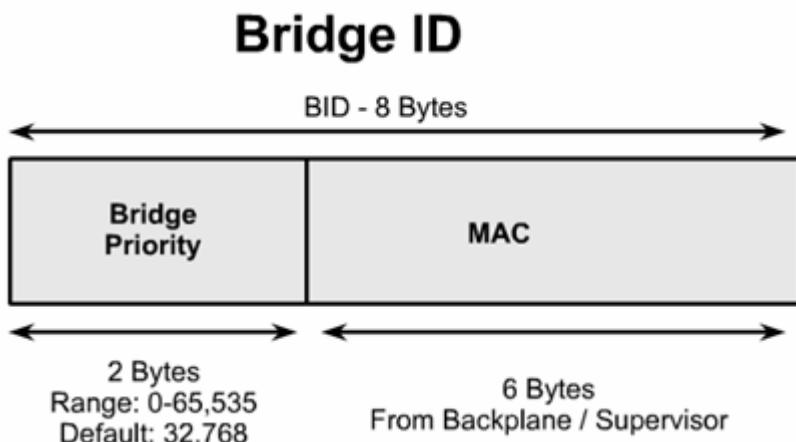
STA networkte bir referans noktası oluşturur ve bu referans noktasından hareketle, birden fazla alternatif yol varsa, en iyi yol seçimini yapar. Bu referans noktasına Root Bridge denir.

Peki Root Bridge nedir, nasıl seçilir, kim seçer...

Aslında ortamındaki bütün switchler Root Bridge'dir. Yani kendilerini böyle sanarlar ☺

Ortamda en **kucuk** Bridge ID' ye sahip bridge Root Bridge'dir.

Bridge ID Bridge Priority ve MAC adresinden oluşur, 8 Byte'tır. Bütün switchlerin Priority'si default olarak 32768' dir.



Switchlerin default priority'leri değiştirilmediginde hepsi eşit olacağınından MAC adreslerine bakılabilir, bu durum da en küçük MAC adresine sahip Bridge Root olacaktır. (Switchlerin efendisi ☺)

STP hesaplamaları sırasında en iyi yol seçiminin yapılmasını saglayacak kriter de Path Cost'tur. 1000/ Bandwidth ile hesaplansa da IEEE çok kullanılan bant genişlikleri için costları yayınlamıştır.

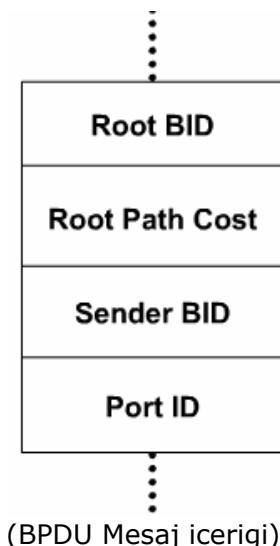
- 4 Mbps 250 (cost)
- 10 Mbps 100 (cost)
- 16 Mbps 62 (cost)
- 45 Mbps 39 (cost)
- 100 Mbps 19 (cost)
- 155 Mbps 14 (cost)
- 622 Mbps 6 (cost)
- 1 Gbps 4 (cost)
- 10 Gbps 2 (cost)

(Bir çok hesaptan kurtulduk sanırım ☺)

Butun bu ogrendiklerimizin isiginda switchlerin kriter olara aldiglari 4 adimi siralayabiliriz.

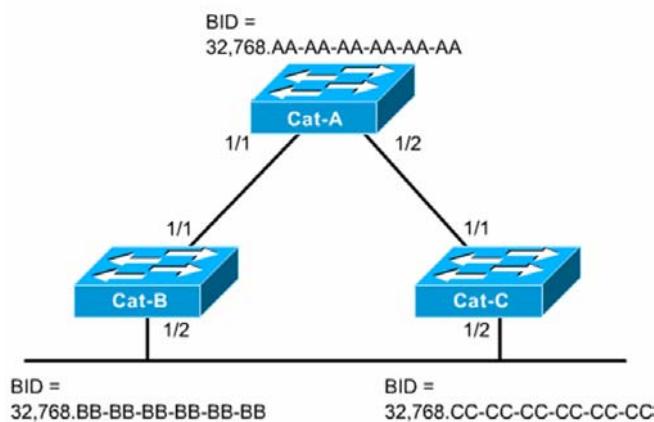
- Step 1 - Lowest BID
- Step 2 - Lowest Path Cost to Root Bridge
- Step 3 - Lowest Sender BID
- Step 4 - Lowest Port ID

Bridge butun bu haberlesmeler icin BPDU (Bridge Protocol Data Unit) mesajlarini kullanir, bu mesajlari bahsettigimi 4 adima gore degerlendirir. Bridge sadece kendisine gelen en iyi BPDU'yu tutar ve her yeni BPDU icin 4 adimi tekrarlar. Gelen BPDU'lar arasında daha iyisi varsa onu alip digerini silecektir.



STP 3 basamak ile yapisini olsturur.

- Step 1 Root Bridge Secilir
- Step 2 Root Portlar Secilir
- Step 3 Designated Portlar secilir.



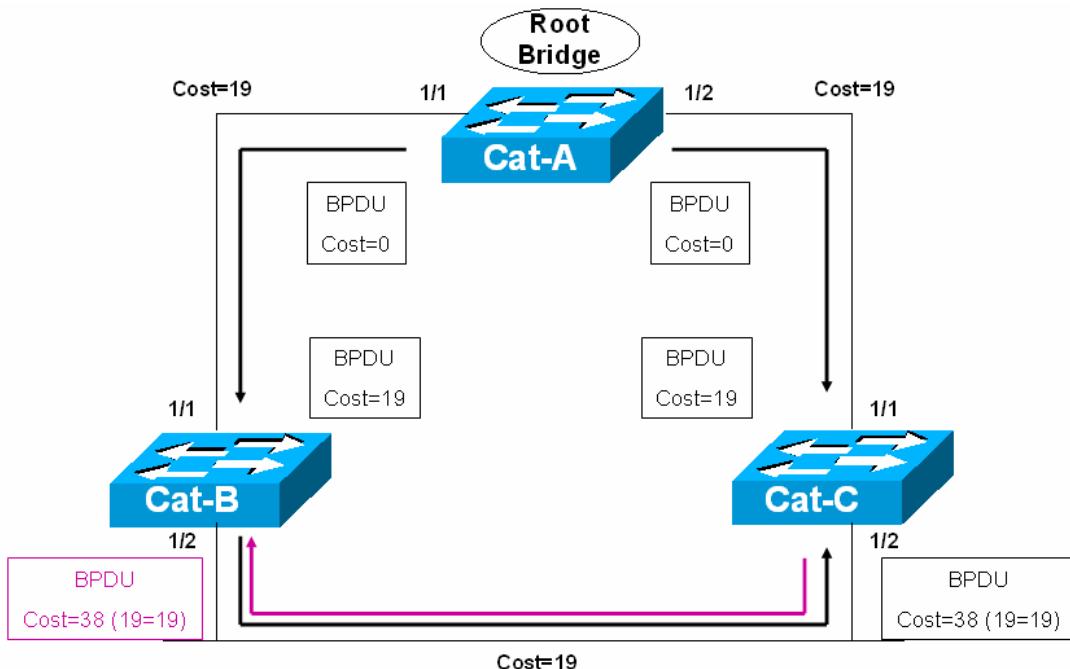
Sekilde ki gibi bir yapı üzerinden hareket ederek bu adimları açıklamaya çalışalım. İlk adımda Root Bridge seçilmesi gereklidir ve bunu için BPDU Mesajları gönderilir. Oncede söyledigmiz gibi her

Switch kendisini Root Bridge varsayıcağı için BPDU mesajlarındaki Root Bridge ID alanına kendi Idlerini yazar. (Root War başlıyor ☺)

Cok gecmeden durumun oyle olmadığını anılar asında ortamda ki CAT-A Switchinin gerçek Root olduğunu öğrenip BPDU larına bu Swirchi Root olarak eklerler. Root Bridge'in portları her zaman Designated porttur ve sürekli forward durumundadır.

Root Bridge seçildiğine göre ikinci adıma, diğer Switchler için Root Portlarının seçilmesine gelebiliriz.

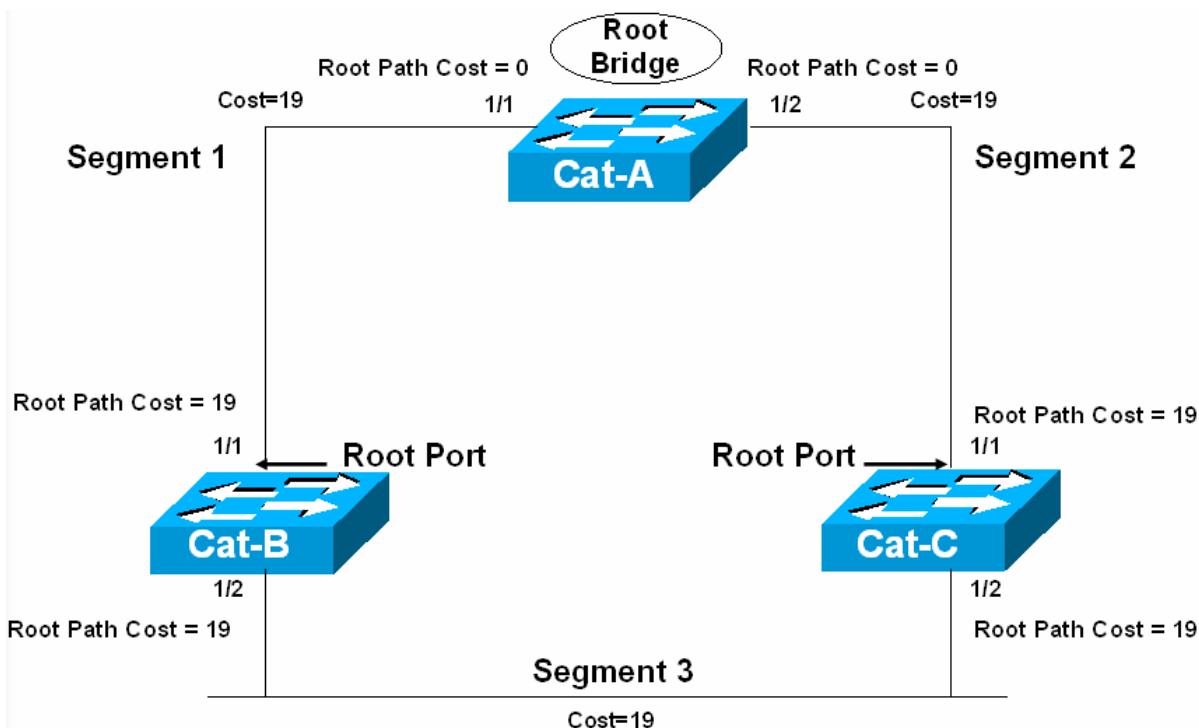
Switchlerin Root Bridge'ye en yakın portları Root Porttur. Burası yakınıktan kast ettigimiz şey asında Root Bridge olan costtur.



Şekilde costları incelediğimiz zaman rahatlıkla Cat-B ve Cat-C' nin 1/1 portlarının Root Port olduğunu söyleyebiliriz. Bu arada her Switch için sadece 1 tane Root Port , her segment içinde sadece 1 tane root port olmalı.

Root Portlar seçildiğine göre Designated portlara gelebiliriz.

Aşağıdaki şekilde de görüleceği gibi Segment 3' e dahil olan sadece 2 switch var ve bu switchlerin birer tane seçilebilen root portları bu segmentte degiller. Bu yuzden bu segment için bu iki Switch portlarından biri Designated Port olarak seçilmeli.



Yine sekilde goruldugu gibi Her iki Switchi birbirlerine baglayan portların Root Path Costlari esit. Root Bridge ID' lerde esit olduguna gore 3. adima gecebiliriz.

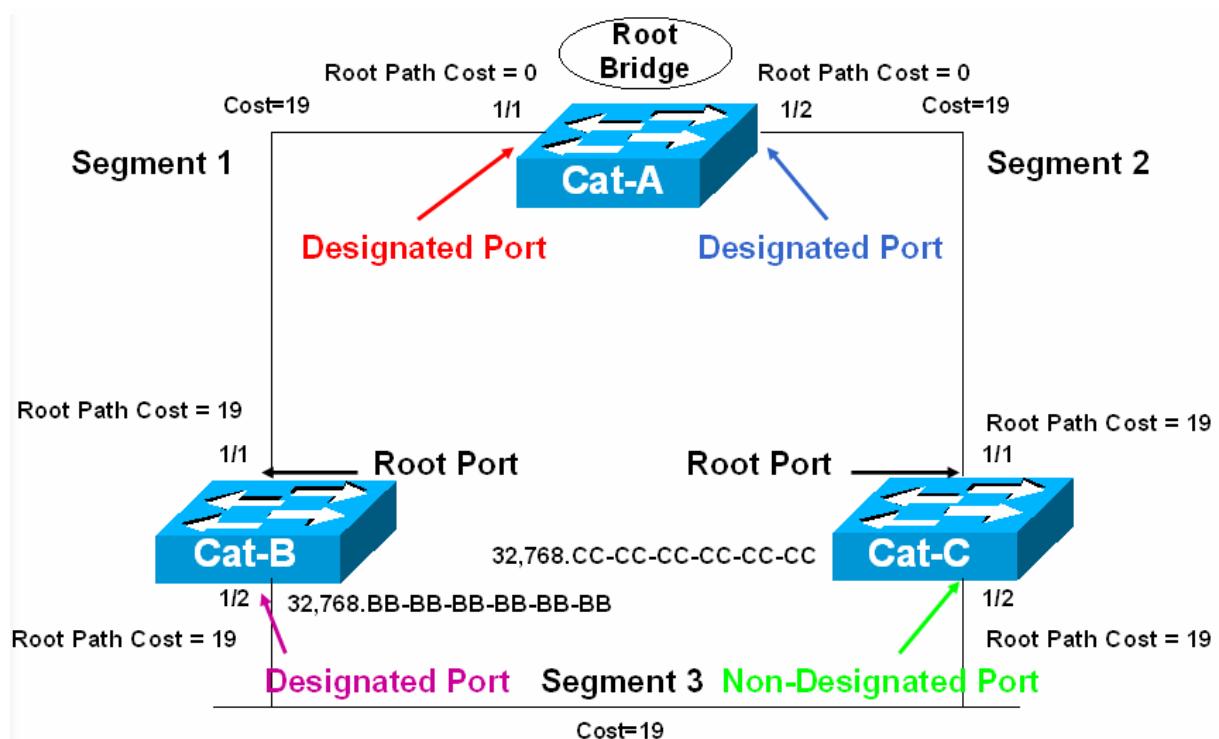
Hatirlamak icin Switchlerin BPDU paketleri iceriginde sirasiyla inceledikler 4 adimi tekrar siralamakta fayda var.

- Step 1 - Lowest BID
- Step 2 - Lowest Path Cost to Root Bridge
- Step 3 - Lowest Sender BID
- Step 4 - Lowest Port ID

Yani Lowest Sender BID... Bu durumda her iki Switch icin BID degerleri karsilastirilip kucuk olan Switche ait portun Designated Port oldugunu söyleyebiliriz.

Sonuc olarak Designated Port Forward duruma yani iletme gececek Non- Designated Port Block durumda kalacaktir.

Ornek olmasi acisindan MAC Adresleri ve Priority degerleri, bu bilgilerden hareketler Portlarin durumu asagidaki sekilde ozetlenmistir.



Spanning Tree Port Durumları

Spanning Tree yapısı içerisinde portlar 5 ayrı durumda bulunabilirler.

1. Forwarding : Datalar gönderilir ve alınır.
2. Learning : Bridge Table oluşturulur.
3. Listening : Aktif topology oluşturulur.
4. Blocking : Sadece BPDU'lar alınır.
5. Disabled : Yönetimsel olarak down durumdadır.

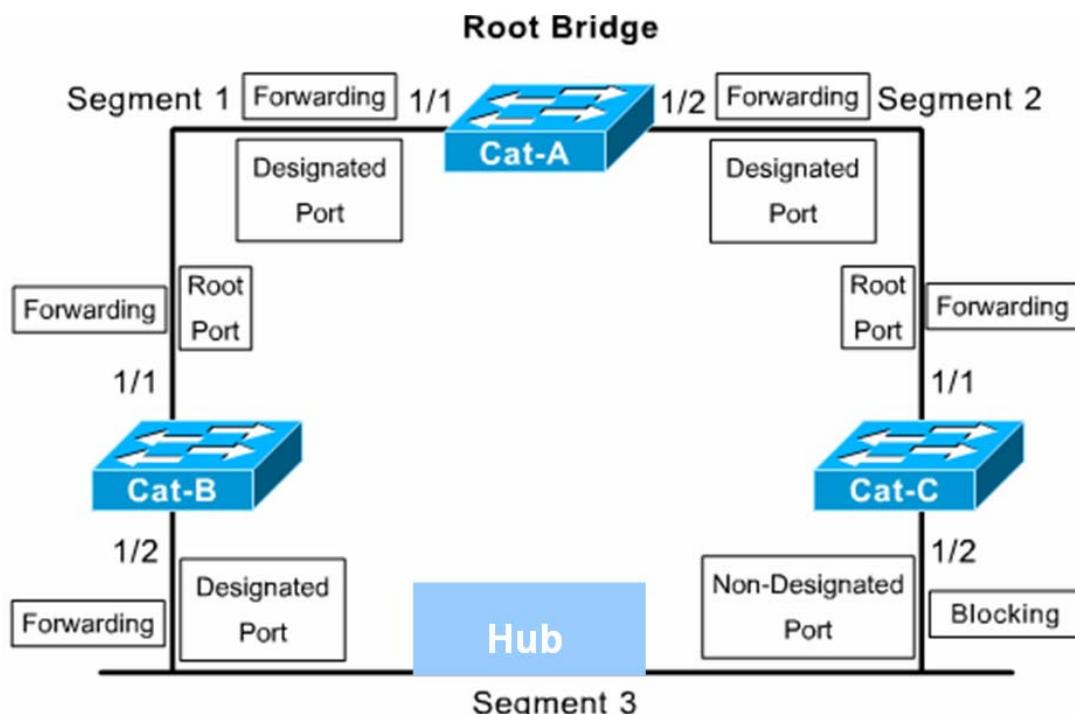
STP Timers

Hello Time: Root Bridge tarafından gönderilen BPDU mesajları zaman aralığıdır. Default olarak 2 saniyedir.

Forward Delay: Portların Forward duruma gecmeden önce Listening ve Learning adımlarında geçen suredir, 15 saniyedir.

Max. Age: Bir BPDU'nun saklanma süresidir, 20 saniyedir. 20 saniye boyunca daha önce aldığı en iyi BPDU mesajı tekrarlanmazsa Max. Age dolmuş olur ve port Listening Moda geçer.

Bir örnek ile STP zaman aralıklarını inceleyelim.



Şekildeki duruma göre başlangıçta Cat-C' nin ½ portu Blocking durumda ve yalnızca BPDU mesajlarını dinliyor.

Simdi Cat-B' nin ½ portunun down olduğunu varsayıyalım. Bu durumda Cat-C artık BPDU mesajlarını alamayacaktır. Cat-C 20 saniye boyunca Blocking durumda kalacak ve 20 saniyenin sonunda Max. Age' e ulaşıldığı için durumunu değiştirecek, 15 saniye surecek Listening mod ve yine 15 saniye surecek Learning Modun ardından Forwarding duruma geçecektir.

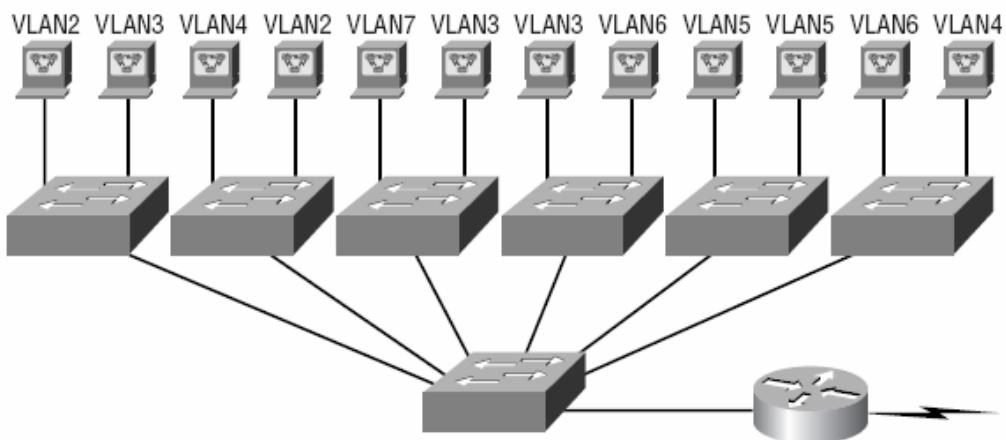
Yani 20 sn. max age + 15 sn. Listening + 15 sn. Learning modda kalacak dolayısıyla Cat-B ½ portu down olduktan 50 saniye sonra Cat-C ½ portu devreye girecekyir.

Fakat burada Cat-B' nin MAC Adres Table' inin silinmemesinden ve toplam 300 saniye botunca da Cache de kalacak olmasından dolayi bir sorun var gibi gorunuyor.

Bu sorunda Root Bridge tarafından gonderilecek TCN BPDU (Topology Change Notification BPDU) ile giderilecektir. Ortamdaki switch portlarinin durumunda bir degisiklik oldugunda gonderilen bu mesaj ile switchler MAC adres Table' larinin yasam surelerini 15 saniyeye cekerler.

VLANs (Virtual Local Area Networks)

Virtual Local Area Network switch üzerinde yapılan mantıksal biri graplama şekilden tanımlanabilir. VLAN oluştururken bilgisayarların fiziksel durumlarına, yerlerine bakmak yerine işlevine ya da departmanına göre düzenlemeler yapılır. Örneğin bir networkte Muhasebe bölümü bir VLAN' da İnsan Kaynakları başka bir VLAN' da bulundurulabilir ve bu sayede iki departman arasından ki iletişim engellenmiş olur.



Pazarlama	VLAN2	172.16.20.0/24
İnsan Kaynakları	VLAN3	172.16.30.0/24
Teknik Büro	VLAN4	172.16.40.0/24
Muhasebe	VLAN5	172.16.50.0/24
İYönetim	VLAN6	172.16.60.0/24
Satış	VLAN7	172.16.70.0/24

Her VLAN ayrı bir Broadcast domain olur ve dolayısıyla Broadcast' ler kontrol altına alınabilir. Network üzerinde kullanılan hemen hemen her protokol Broadcast oluşturur ve bu broadcast' lerin miktarı Network performansını olumsuz etkileyebilir. Bunu önlemenin iki yolu vardır:

- Router kullanımı
- Switch Kullanımı

Sistem içerisinde uzak networkler varsa Router kullanımı uygun bir çözüm olabilir ama Local Area Network düşünüldüğünde Switch kullanmak ve VLAN' lar oluşturmak daha ucuz dolayısıyla daha mantıklı bir çözüm olacaktır.

VLAN' lar Switch portlarının Network yöneticileri tarafından atanmasıyla oluşturulur ki buna Static VLAN denir. Sistem de bulunan cihazların bir veritabanına girilmesi ve switchler tarafından otomatik oalrak atanmasıyla oluşan VLAN' lara ise Dinamik VLAN denir.

Static VLAN' lar hem daha güvenlidir hem de yönetimi ve bakımını Dinamik VLAN' lara göre daha kolaydır.

Default olarak bir switch üzerindeki bütün portlar VLAN1' dendir.

VLAN konfigürasyonu Switch modeline göre farklılık gösterebilir. Önemli olan mantığını anlamaktır, komutlar kullanılan switch içerisinde yardım alınarak yapılabilir. (Biz hem Cisco1900 hem de Cisco 2950 serisi switchlerin konfigürasyon komutlarını vereceğiz fakat konfigürasyon çalışması yaparken Cisco1900 serisi Switchler üzerinde çalışacağız.)

VLAN oluşturmak komutlardan bağımsız olarak anlatmak gerekirse iki adımdan oluşur.

1. VLAN Oluşturulur
2. Portlar VLAN' lara üye edilirler.

1900 Switch İçin VLAN Oluşturma:

```
Switch#configure terminal
Switch(config)#vlan 2 name satis
Switch(config)#vlan 3 name muhasebe
Switch(config)#vlan 4 name yonetim
Switch(config)#exit
Switch#
```

2950 Switch İçin VLAN Oluşturma:

```
Switch#configure terminal
Switch(config)#vlan 2
Switch(config-vlan)#name satis
Switch(config)#vlan 3
Switch(config-vlan)#name muhasebe
Switch(config)#vlan 4
Switch(config-vlan)#name yonetim
```

2950 Seri switchlerde her VLAN kendi alt modunda konfigüre ediliyor.

NOT: VLAN1 silinemez, değiştirilemez veya yeniden adlandırılabilir.

VLAN'lar oluşturulduktan sonra artık ikinci adıma geçebiliriz. Bu adımda Switch portları VLAN'lar ile eşleştirilecek. Tabi burada VLAN üyeliğinin Static yada Dynamic olduğu da belirtiliyor.

1900 Seri Switchler İçin VLAN Üyeliği:

```
Switch#configure terminal
Switch(config)#interface Ethernet 0/2
Switch(config-if)#vlan-membership static 2
Switch(config-if)#exit
Switch(config)#interface Ethernet 0/3
Switch(config-if)#vlan-membership static 3
Switch(config-if)#exit
Switch(config)#interface Ethernet 0/4
Switch(config-if)#vlan-membership static 3
Switch(config-if)#exit
Switch(config)#
```

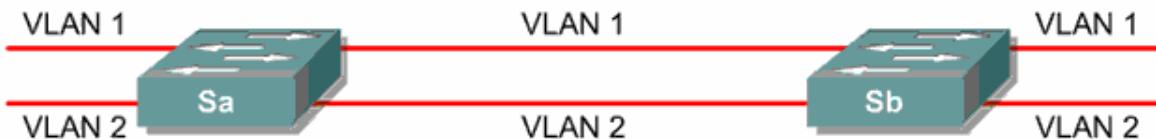
2950 Seri Switchler için VLAN Üyeliği:

```
Switch#configure terminal
Switch(config)#interface Ethernet 0/2
Switch(config-if)#switchport Access vlan 2
Switch(config-if)#exit
Switch(config)#interface Ethernet 0/3
Switch(config-if)# switchport Access vlan 3
Switch(config-if)#exit
Switch(config)#interface Ethernet 0/4
Switch(config-if)# switchport Access vlan 4
Switch(config-if)#exit
Switch(config)#[/pre>
```

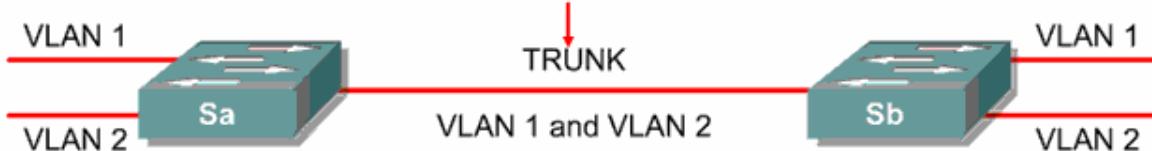
Trunk ve Trunk Konfigürasyonu

Trunk bağlantılar cihazlar arasında VLAN'ları taşımak amacıyla kullanılırlar ve VLANların tümünü ya da bir kısmını taşımak üzere biçimlendirilebilirler. Sadece Fast ya da Gigabit Ethernet üzerinde desteği vardır. Cisco switch'ler trunk bağlantı üzerindeki VLAN'ları tanıtmak için iki ayrı yöntem kullanır: **ISL** ve **IEEE802.1q**.

No VLAN Tagging



VLAN Tagging



Bir Switch üzerindeki bir porta trunk ing konfigürasyonu şu şekilde olur:

1900 Seri Switch için:

```

Switch#configure terminal
Switch(config)#interface fastethernet 0/20
Switch(config-if)#trunk on
Switch(config-if)#exit
    
```

2950 Seri Switch için:

```

Switch#configure terminal
Switch(config)#interface fastethernet 0/20
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
    
```

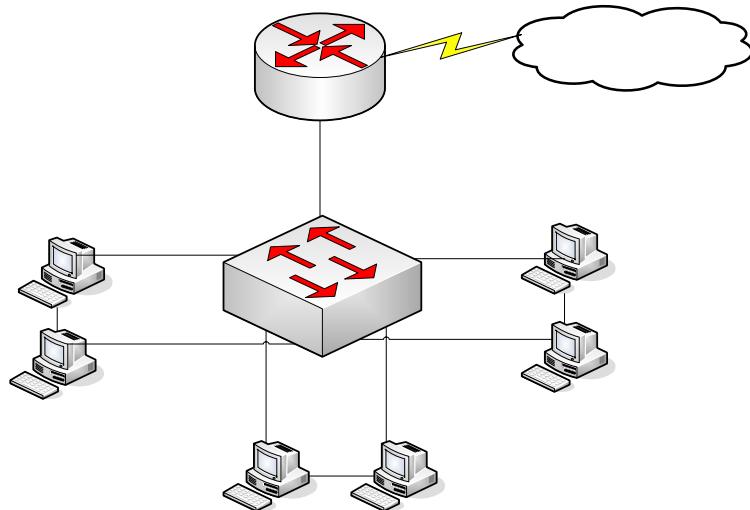
Inter-Switch Link (ISL): Cisco switch'ler tarafından kullanılır. Bu yöntem "external tagging" adı verilen, paketin orijinal boyunu değiştirmeyen, ancak 26 byte'lık bir ISL başlığını pakete ekleyerek, cihazlar arasında VLAN tanınmasını sağlayan bir yöntemdir. Ayrıca paketin sonuna paketi kontrol eden 4-byte uzunluğunda FCS (frame check sequence) alanı ekler. Paket bu eklentilerden sonra sadece ISL tanıyan cihazlar tarafından tanınabilir.

IEEE 802.1q: IEEE tarafından geliştirilen bu standart yöntem, farklı markadan switch ya da router arasında, bir bağlantı üzerinden çok VLAN taşımak amacıyla kullanılır. Gelen paket üzerine tanımlanan standarda uygun bir başlık yerleştirilir ve cihazlar arasında pakete ait VLAN'ın tanınması sağlanır.

VLAN'lar Arasında Yönlendirme

Bir VLAN'a bağlı cihazlar kendi aralarında iletişim kurabilir, broadcast'lerini gönderebilirler. VLAN'ların network'ü fiziksel olarak böldükleri varsayıldığı için VLAN'lar arasında cihazların iletişim kurabilmesi ancak 3. katman bir cihaza yardımıyla olacaktır.

Bu durumda yapılacak bir router üzerinde her VLAN için bir bağlantı eklemek ve Router üzerinde gerekli konfigürasyonları yaparak iletişimini sağlamaktır.



Böyle bir topoloji üzerinde çalıştığımızı varsayıalım:

```
Switch#configure terminal
Switch(config)#interface fastethernet 0/1
Switch(config-if)#trunk on
Switch(config-if)#exit
Switch(config)#interface fastethernet 0/2
Switch(config-if)#vlan-membership static 1
Switch(config-if)#exit
Switch(config)#interface fastethernet 0/3
Switch(config-if)#vlan-membership static 1
Switch(config-if)#exit
Switch(config)#interface fastethernet 0/4
Switch(config-if)#vlan-membership static 2
Switch(config-if)#exit
Switch(config)#interface fastethernet 0/5
Switch(config-if)#vlan-membership static 2
Switch(config-if)#exit
Switch(config)#interface fastethernet 0/6
Switch(config-if)#vlan-membership static 3
Switch(config-if)#exit
Switch(config)#interface fastethernet 0/7
Switch(config-if)#vlan-membership static 3
Switch(config-if)#exit
```

İlgili portları ilgili VLAN'lara atadık ve Router'a bağlantının sağlandığı FastEthernet0/1 portunda trunking'i aktif hale getirdik. Şimdi sıra Router üzerinde gerekli konfigürasyonu yapmaya geldi.

Bunun için Router'ın Fastethernet 0/0 interface'sı altında sanal interface'ler oluşturmak, bu sanal interfacelere ip adresleri atamak ve encapsulation standardını belirlemek gerekir.

NOT: Gerçek Interface'in ip adresi olmamalı.

```
Router#configure terminal
Router(config)#interface fastethernet 0/0
Router(config-if)#no ip address
Router(config-if)#no shutdown
Router(config-if)#interface fastethernet 0/0.1
Router(config-subif)#encapsulation isl 1
Router(config-subif)#ip address {ip adresi} {subnet maskı}
Router(config-subif)#exit
Router(config-if)#interface fastethernet 0/0.2
Router(config-subif)#encapsulation isl 2
Router(config-subif)#ip address {ip adresi} {subnet maskı}
Router(config-subif)#exit
Router(config-if)#interface fastethernet 0/0.3
Router(config-subif)#encapsulation isl 3
Router(config-subif)#ip address {ip adresi} {subnet maskı}
Router(config-subif)#exit
```

Burada öncelikle 3 adet VLAN için Router üzerinde 3 adet sanal interface oluşturuldu. Hemen arkasından kullandığımız switchin 1900 serisi olduğunu varsayıarak encapsulation metodunu belirledik ve o sanal interface'in hangi VLAN ile bağlantılı olduğunu belirledik.

2950 Seri Switchler 802.1q metodunu desteklediği için bu switchlerden konfigürasyonumuz:

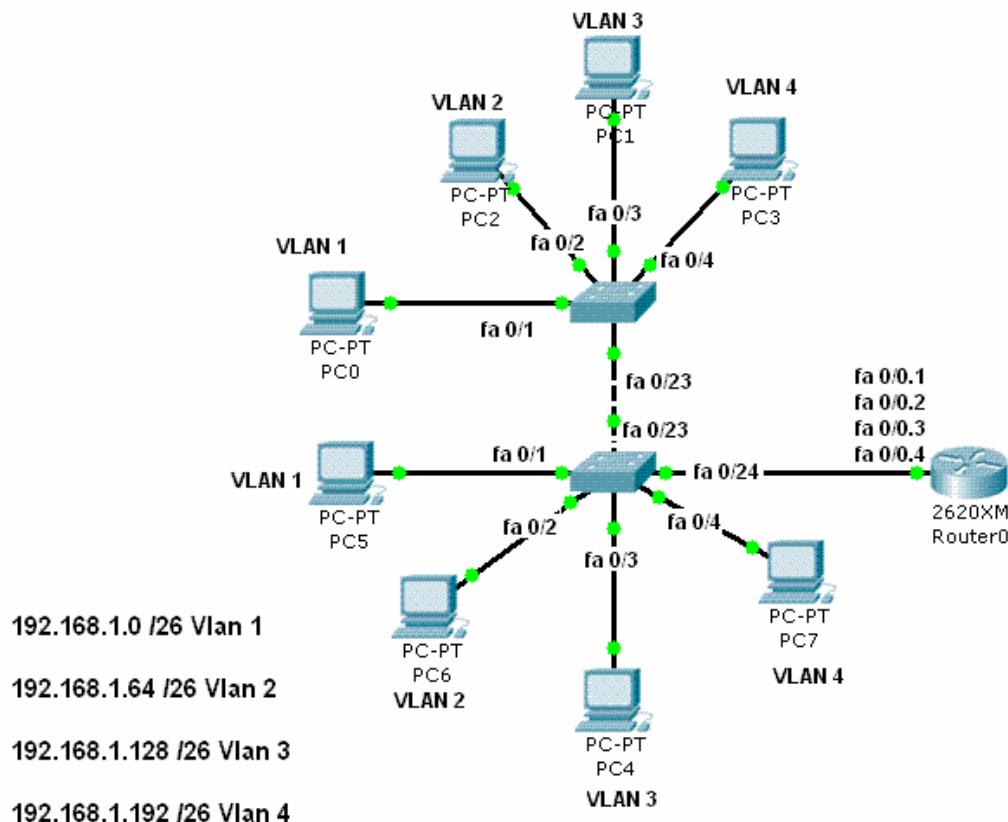
```
Router(config-if)#interface fastethernet 0/0.1
Router(config-subif)#encapsulation dot1q 1
Router(config-subif)#ip address {ip adresi} {subnet maskı}
```

Şeklinde olacaktır. (802.1q = dot1q)

İşlemimiz VLAN üzerindeki ip adreslerinden birini (önerilen ilk useable ip adresini) sanal interface'e vererek tamamlandı.

NOT: Cisco 1900 serisi switchler sadece isl encapsulation metodunu desteklerler 2950 serisi switchler ise sadece 802.1q'yu destekler. Bu yüzden bu iki switch arasında trunking gerçekleştirilemez.

Laboratuar Çalışması



Laboratuar çalışmamızda VLAN 1 de dahil olmak üzere SwitchA ve SwitchB ye bağlı 4 adet VLAN var. SwitchA ve SwitchB fa0/23 portlarından birbirlerine bağlanmış ve bu portlarda trunk uygulanmıştır,

Aynı şekilde SwitchB fa0/24 portundan Routera bağlanmış ve bu porttada trunk uygulanmıştır.

Encapsulation dot1q kullanılmıştır.

Switch ve Router running-config dosyaları aşağıdadır.

```
SwitchA#show running-config
!
version 12.1
!
hostname SwitchA
!
interface FastEthernet0/1
  switchport mode access
!
interface FastEthernet0/2
  switchport access vlan 2
  switchport mode access
!
interface FastEthernet0/3
  switchport access vlan 3
  switchport mode access
!
```

```

interface FastEthernet0/4
switchport access vlan 4
switchport mode access
!
-----
!
interface FastEthernet0/23
switchport mode trunk
!
interface FastEthernet0/24
switchport mode access
!
!
interface Vlan1
ip address 192.168.1.11 255.255.255.192
!
ip default-gateway 192.168.1.1
!
line con 0
!
end

```

SwitchA#show vlan

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/5, Fa0/6, Fa0/7 Fa0/8, Fa0/9, Fa0/10, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15 Fa0/16, Fa0/17, Fa0/18, Fa0/19 Fa0/20, Fa0/21, Fa0/22, Fa0/23
		Fa0/24, Gig1/1, Gig1/2
2 egitim	active	Fa0/2
3 muhasebe	active	Fa0/3
4 yonetim	active	Fa0/4
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

SwitchA#

SwitchB#show running-config
version 12.1

```

!
hostname SwitchB
interface FastEthernet0/1
switchport mode access
!
interface FastEthernet0/2
switchport access vlan 2
switchport mode access
!
interface FastEthernet0/3
switchport access vlan 3

```

```

switchport mode access
!
interface FastEthernet0/4
switchport access vlan 4
switchport mode access
!
interface FastEthernet0/23
switchport mode trunk
!
interface FastEthernet0/24
switchport mode trunk
!
!
interface Vlan1
ip address 192.168.1.12 255.255.255.192
!
ip default-gateway 192.168.1.1
!
line con 0
!
end

```

SwitchB#show vlan

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/5, Fa0/6, Fa0/7 Fa0/8, Fa0/9, Fa0/10, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15, Fa0/16, Fa0/17, Fa0/18, Fa0/19, Fa0/20, Fa0/21, Fa0/22, Fa0/23 Fa0/24, Gig1/1, Gig1/2
2 egitim	active	Fa0/2
3 muhasebe	active	Fa0/3
4 yonetim	active	Fa0/4
1002 fddi-default	active	
1003 token-ring-default	active	
1004 fddinet-default	active	
1005 trnet-default	active	

SwitchB#

```

Router#show running-config
!
version 12.2
!
hostname Router
!
interface FastEthernet0/0
no ip address
interface FastEthernet0/0.1
encapsulation dot1Q 1 native
no ip address
!
interface FastEthernet0/0.2
encapsulation dot1Q 2

```

```
ip address 192.168.1.65 255.255.255.192
!
interface FastEthernet0/0.3
encapsulation dot1Q 3
ip address 192.168.1.129 255.255.255.192
!
interface FastEthernet0/0.4
encapsulation dot1Q 4
ip address 192.168.1.193 255.255.255.192
!
interface Serial1/0
no ip address
shutdown
ip classless
!
line con 0
!
end
```

Router#

VLAN Trunking Protocol (VTP)

VTP Vlan konfigutasyonun butun networke yayilmasini saglayan bir mesajlasma protokoludur.

VTP Layer 2 framelerini kullanir ve VLAN' larin butun network icinde yonetilmesini, silinmesini, eklenmesini ya da yeniden adlandirilmasini saglar. Dolayisiyla VTP networkteki butun switchlerin ve VLAN konfigurasyonlarının merkezi bir sekilde yonetilmesini saglar.

VTP protokolunun calisma prensibi icinde ortamda VTP Server ve VTP clientlar bulunur. Ayni domain de bulunan VTP Clientlar , serverdan VLAN bilgilerini alırlar.

- VLAN'lar VTP Server da olusturulur.
- VLAN bilgileri client switchlere gonderilir.
- Ayni domain icinde bulunan switchler VLAN bilgilerini alırlar.
- Bu gelismeden sonar artik client switchlerde portlar VTP Server da olusturulan VLAN' lara atanabilir.
- VTP Client olarak konfihure edilen switchlerde VLAN olusturulamaz.
- Farkli domainlerde bulunan switchler VLAN bilgilerini paylasmazlar.

Switchler VTP bilgilerini almamak uzerede configure edilebilirler, Bu switchler VLAN bilgilerini Trunk portlarından gonderirken kendisine gelen bilgileri almaz ve kendi VLAN database' ini yapiilandirmaz. Switchleri vu sekilde calismasi VTP Mode Transparent olarak adlandirilmistir. Bu modda calisan Switchler VTP domaine katilmazlar.

Güvenlik acisindan VTP domainlerine password verilebilir. Bu durumda password o domain de bulunan butun switchlerde configure edilmelidir.

Gonderilen VTP mesaflari VTP database' inden revision numarası ile birlikte tutulurlar, her mesaj ile bu numara artırılır. Daha büyük bir revision numarası ile gelen bilgiler switchler tarafından daha yeni olarak Kabul edilir ve gelen VLAN bilgileri eskilerinin üzerine yazılır.

Buraya kadar anlattıklarımızın ışığında VTP domainlerinde switchlerin 3 ayrı modda çalışabileceklerini söyleyebiliriz.

- VTP Server
- VTP Client
- VTP Transparent

Konfigurasyon:

```
Switch# vlan database
Switch(vlan)# vtp domain domain-name
Switch(vlan)# vtp {server | client | transparent}
Switch(vlan)# vtp password password
Switch(vlan)# vtp v2-mode (version2)
```

Ornek

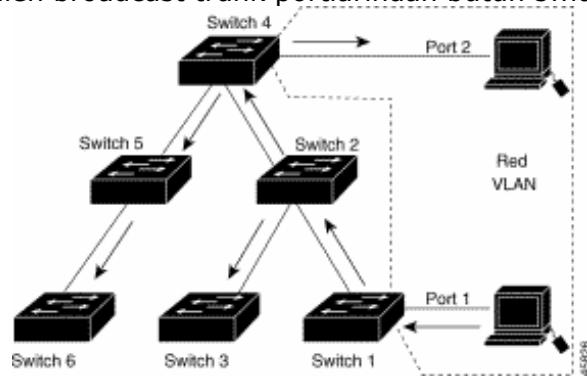
```
Switch# vlan database
Switch(vlan)# vtp domain corp
Switch(vlan)# vtp client
```

VTP Pruning (Budama)

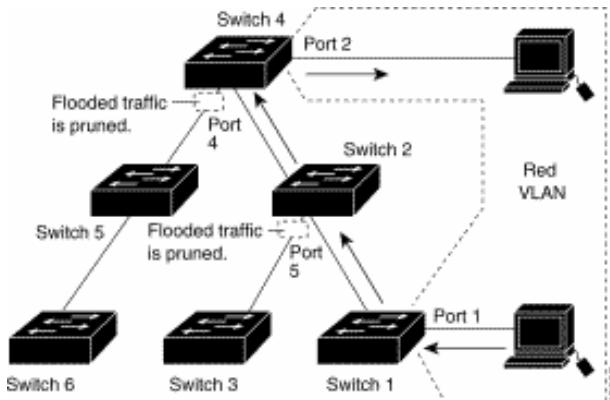
VTP Pruning networkteki broadcast, multicast, unknown unicast gibi gereksiz flood edilen paketleri azaltarak network bant genisligi kullanimini artirir. Cisco Switchlerde default olarak disable durumdadır.

VLAN1' de VTP Pruning enable edilemezken diger VLAN' larda edilebilir ve VTP Server da Pruning enable edildiginde ise butun domainde (tabi ki VLAN 1 disinda) enable olur.

Asagidaki sekilde Switch1'in 1. portu ve Switch4' un 2. portu Red VLAN'1 uye durumdalar. Hoslardan birinden gonderilen broadcast trunk portlarindan butun switchlere gider.



Red VLAN'a uye portlari olmayan Switch 2-3-5-6' da aynı sekilde bu broadcast alacaktır. Bunu onlemek icin VTP Pruning enable edilebilir.



Switch4 un 4. portu ve Switch2' nin 5. portunda Red VLAN trafigi budanmistir. (VTP Pruning enable)

Konfigurasyon

```
Switch# vlan database
Switch(vlan)# vtp pruning
```

Belirli bir VLAN ise

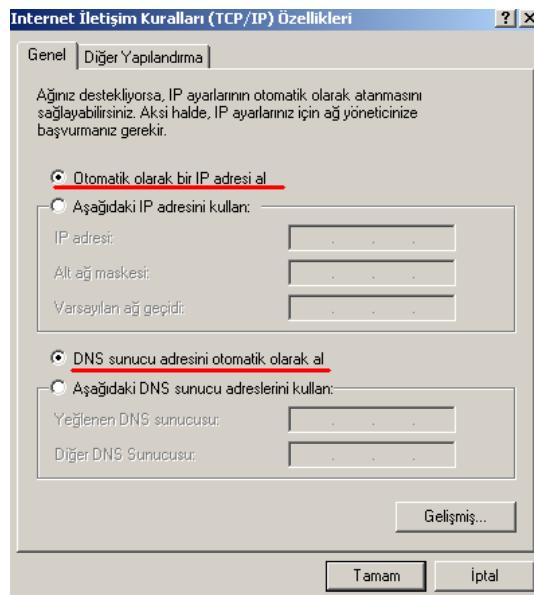
```
Switch(config-if)#switchport trunk pruning vlan remove vlan
```

Komutuyla pruning disinda bırakılabilir.

DHCP (Dynamic Host Configuration Protocol)

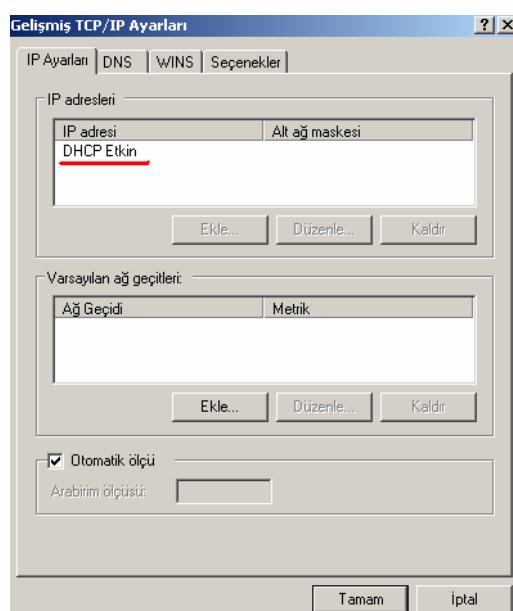
DHCP , DHCP kullanmak üzere yapılandırılmış bilgisayarlara merkezi ve otomatik olarak ip adresi atanması ile TCP/IP bilgilerinin yapılandırılmasını ve bunların yönetilmesini sağlar. DHCP' nin uygulanması manuel olarak ip adresinin verilmesi nedeniyle ortaya çıkan bazı problemlerin azalmasını sağlar.

Bir bilgisayarı DHCP kullanmak üzere yapılandırmak için bilgisayarın TCP/IP konfigürasyonunda "Otomatik olarak ip al" seçeneğini aktif etmek yeterlidir. İstendiğinde DNS sunucunun ismi de otomatik olarak DHCP Server' da alınabilir bunun için de " DNS sunucusunu otomatik olarak al " seçeneği aktif hale getirilmelidir. Bu işlemler yapıldıktan sonra bilgisayarlar DHCP istemci durumuna gelecektir.



(Bir bilgisayının DHCP istemci olarak ayarlanması)

Bu ayarlar yapıldıktan sonra TCP/IP konfigürasyonunun gelişmiş sekmesine baktığımızda DHCP' nin etkin olduğunu görebiliriz.



DHCP istemci DHCP Server ile haberleşmeye geçmesi ve ip adresini elde etmesi birkaç adımlık bir haberleşme ile sağlanır, Bu birkaç adımı basit bir şekilde inceleyeceğiz olursak:

- İstemci bilgisayar başlangıçta DHCP Server adresini bilmediği için broadcast yolu ile ip adres isteğini ortama yayar.
- İsteği alan DHCP Server, uygun olan bir ip adresini istemciye kiralama teklifinde bulunur. (Ip adresleri DHCP Server' lar tarafından belirli sürelerle istemcilere kiralanırlar, tamamen verilmezler)
- İstemci ip adres bilgilerini alır.
- DHCP Server veritabanında ip adresinin kiralandığı ve kiralama süresi bilgilerini yazar.

Özellikle büyük işletmelerde IP konfigürasyonu ile ilgili çıkabilecek sorunların çözülmesinde ya da olası değişikliklerin düzenlenmesinde DHCP Server ile TCP/IP konfigürasyon bilgilerini dağıtmak akıllıca bir çözüm olacaktır.

Bunun için bir bilgisayarı DHCP Server atamak yeterli olabileceği gibi istendiğinde Router' larda gerekli konfigürasyonlar yapıldığında DHCP hizmeti verebilirler.

DHCP Server kullanarak istenirse oluşturulacak ip havuzundan ip adresleri rast gele dağıtılabılır ya da MAC adreslerine bazı ip adresleri reserve edilebilir ve istenirse bazı ip adreslerinin hiçbir şekilde dağıtılmaması sağlanabilir.

Cisco Router'ın DHCP Server Olarak Konfigüre Edilmesi

Cisco Router'ıarda DHCP server default olarak çalışır durumdadır. Herhangi bir nedenle daha önceden DHCP Server devre dışı bırakıldıysa;

Router(config)# service dhcp

komutu ile DHCP Server aktif hale getirilebilir. Yine istendiği zaman başına "no" konularak devre dışı bırakılabilir.

Router(config)# no service dhcp

Router'ın DHCP hizmeti verebilmesi için, hangi aralıklarda hangi networke ait ip adreslerinin dağıtılacığı bilgisinin Router'a bildirilmesi gereklidir.

Bunun için şu komutlar yazılmalıdır:

Router(config)#ip dhcp pool *poolismi*

Router(Config-dhcp)# network *ip_araligi* mask *subnet_maski*

Örneğin:

Router(Config)# ip dhcp pool Academytech

Router(Config-dhcp)#network 192.168.0.0 mask 255.255.0.0

İstersek bu networkteki bazı ip adreslerinin ya da bir ip adres aralığının istemci bilgisayarlara dağıtılmasını engelleyebiliriz. Bunun için "**ip dhcp excluded**" komutunu kullanmalıyız. Komutun genel kullanımı şu şekildedir;

Router(config)#ip dhcp excluded-address *baslangic_ipsi bitis_ipsi*

Örneğin ilk örnekte belirttiğimiz ip adres aralığına ait adreslerden 192.168.1.1 'den 192.168.1.10 'a kadar olan ip adreslerinin dağıtılmamasını istersek;

Router(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.10

Komutunu yazmamız gereklidir.

Bununla birlikte DNS ip adresi, etki alanı adı, NetBios Server ip adresi ve Default Gateway gibi adresleri de konfigürasyonunu yaptığımızda Router ile dağıtabiliriz. Bu komutların genel kullanımı ise şöyledir:

```
Router (config-dhcp)#domain-name academytech.com
```

```
Router (config-dhcp)#dns-server dns_server_ip_adresi
```

```
Router(config-dhcp)#netbios-name-server server_ip_adresi
```

```
Router(config-dhcp)#default-router routerin_ip_adresi
```

İstenirse ip adreslerinin reserve edilebileceğinden bahsetmiştim. Bunun için ip adresi reserve edeceğimiz bilgisayarın MAC adresini bilmemiz gereklidir. Örneğin MAC Adresi 00-11-2F-B2-12-B2 olan bir bilgisayara 192.168.1.100 ip adresini reserve edelim. Bu durumda yeni bir havuz oluşturmalıyız:

```
Router(config)#ip dhcp pool Academytech-Lab
```

```
Router(config-dhcp)#host 192.168.1.100 mask 255.255.0.0
```

```
Router(config-dhcp)#client-identifier 0100-11-2F-B2-12-B
```

Burada MAC adresinin başında yer alan " 01 " ifadesi network kartının Ethernet için tasarlandığı anlamına gelir.

Ip adreslerinin dağıtırken olası çakışmaları önlemek için gerekirse Router'ın ip adreslerini kiraya vermeden önce kullanımında olup olmadığını denetlemesini sağlayabilir ve kira süresini de konfigüre edebiliriz.

```
Router(config)# ip dhcp ping packets ping_sayısı
```

```
Router(config-dhcp)#lease gün saat dakika
```

Ayrıca:

Router# show ip dhcp binding reserve_edilmiş_adres

Reserve ettiğimiz ip adresleri hakkında bilgi,

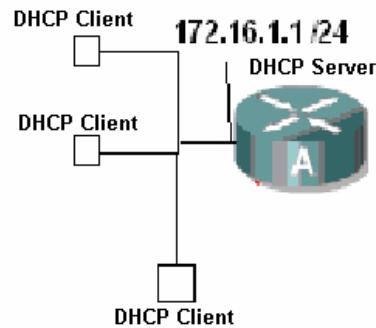
Router# show ip dhcp conflict

Komutu ile dhcp' de çakışan ip adreslerini görüntüleyebilir,

Router# show ip dhcp server statistics

Komutu ile dhcp server hakkında istatistiksel bilgileri alabiliriz.

Şimdi örnek olucak bir konfigürasyon yapalım.



DHCP Client' lara DHCP Server tarafından otomatik olarak 172.16.1.1 / 24 networkünde ip adresleri dağıtılmış olacak.

Senaryoyu biraz daha geliştirmek için 172.16.1.2 – 172.16.1.5 arasıdan ki ip adreslerinin dağıtılmamasını istedığımızı de düşünelim.

```

Router(config)#ip dhcp pool Academytech
Router(dhcp-config)#network 172.16.1.0 255.255.255.0
Router(dhcp-config)#exit
Router(config)#ip dhcp ex
Router(config)#ip dhcp excluded-address 172.16.1.2 172.16.1.5
Router(config)#

```

Burada ip havuzumu oluşturduk ve dağıtılmamasını istemediğimiz ip aralığını router' a bildirdik.

Söz gelimi etki alanı adımız "AcademyTech", Dns Server'ın ip adresi:"172.16.1.2" ve Default Gateway'da 172.16.1.1 olsun. Bu bilgilerinde DHCP tarafından dağıtılmamasını istersek konfigürasyona şu şekilde devam etmeliyiz:

```

Router(config)#ip dhcp pool Academytech
Router(dhcp-config)#domain-name AcademyTech
Router(dhcp-config)#dns-server 172.16.1.2
Router(dhcp-config)#default-router 172.16.1.1
Router(dhcp-config)#exit
Router(config)#

```

00:16:15 bağlandı OtoAlqila 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma

Artık bilgisayarlarımıza DHCP Client olarak ayarladıkten sonra ip adreslerinin bizim router üzerinde yaptığız konfigürasyona uygun olarak alacakları.

Bilgisayarın ip konfigürasyonunda görüldüğü gibi bizim istediğimiz şekilde bir çalışma oldu.

Running Konfigürasyona baktığımız da ise DHCP ile ilgili şu bilgileri göreceğiz:

```
ip dhcp excluded-address 172.16.1.2 172.16.1.5
ip dhcp pool Academytech
  network 172.16.1.0 255.255.255.0
  domain-name AcademyTech
  dns-server 172.16.1.2
  default-router 172.16.1.1
```

DHCP kullanmaktan vazgeçtiğimiz andan itibaren DHCP hizmetini devre dışı bırakabiliriz.

```
| Router(config)#no service dh  
| Router(config)#no service dhcp  
| Router(config)##
```

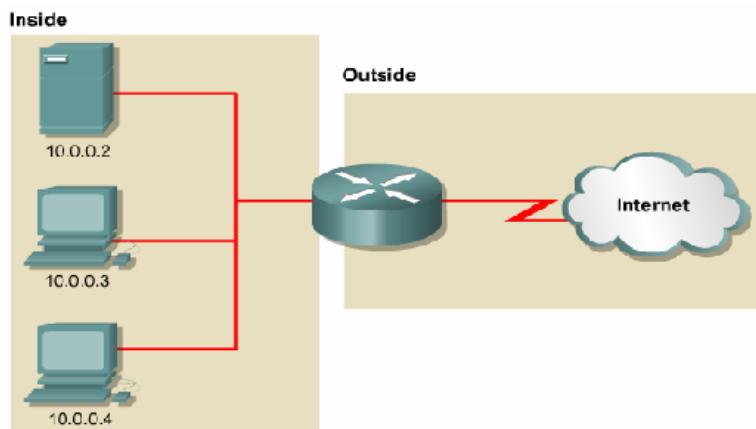
Network Address Translation

Internette gideceğimiz yeri bulmak için IP adresleri kullanırız. Ama her IP adresini internet ortamında kullanamıyoruz. Bazı özel IP adresleri vardır. Bu adresler, daha doğrusu IP adres aralıkları kendi yerel ağlarımızda kullanmamız için ayrılmıştır. Bunlar Address Allocation for Private Internets (özel internetler için adres payı) diye tanımlanır, kısaca Private Addresses (özel adresler) diyoruz. Internette kullandıklarımıza da Public (Halka Açık) Addresses diyoruz.

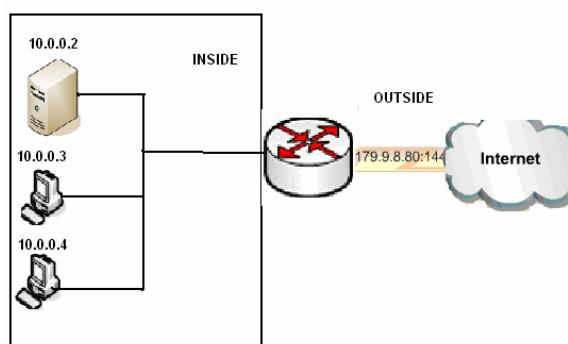
Özel IP adresleri RFC 1918 ile belirlenmiştir ve;

10.0.0.0 ile 10.255.255.255
172.16.0.0. ile 172.31.255.255
192.168.0.0 ile 192.168.255.255 arasındadır.

İç networkümüzde kullanmamız için ayrılan bu ip adresleri internette kullanılamazlar ve biz de bu ip adresleri ile internete erişemeyiz. Dolayısıyla internet ortamına girerken public bir ip adresine sahip olmamız gereklidir. Bu durumda bize NAT yani Network Address Translation yardımcı oluyor ve NAT konfigürasyonu yapıldıktan sonra iç networkümüzdeki herhangi bir ip adresine sahip bilgisayar dışarı çıkarken bizim istedigimiz bir ip adresine dönüştürür, mesela modemin ip adresine.



Bu topolojide 10.0.0.0/24 networküne ait bilgisayarlar internete erişeceklər. Karşımıza "inside" ve "outside" olmak üzere iki kavram çıxır. Yine topolojiden anlaşılıcagı gibi inside iç networkümüz ve outside da dış network yani internet yada hedef network. Bu kavramlar önemlidir zira NAT konfigürasyonu sırasında Adres dönüştürme işleminde inside ve outside olarak kullanılacak interface'ler belirlenmelidir.



(Topoloji ve ip adresleri dikkatle incelendiğinde NAT işlemi görülecektir.)

NAT'ı 3 başlık altında inceleyebiliriz:

- Static NAT : Birebir iç bloktaki IP adreslerini dış IP adreslerine çevirme.
- Dynamic NAT: Bir havuz yaratarak dinamik olarak içerdeki adresleri bu havuzdaki dış IP bloklarıyla eşleme
- Overloading: Bütün makinaları makina sayısına oranla daha az IP adresiyle dışarıya çıkarma

NAT Konfigürasyonu



Böyle bir senaryoda Static NAT uygulaması yapacak olursak Routerlar şu şekilde konfigüre edilmeli.

```
RouterA(config)#ip nat inside source static 192.168.4.2 10.1.1.1
RouterA(config)#interface ethernet 0/0
RouterA(config-if)#ip nat in
RouterA(config-if)#ip nat inside
RouterA(config-if)#exit
RouterA(config)#interface seri
RouterA(config)#interface serial 0/1
RouterA(config-if)#ip nat out
RouterA(config-if)#ip nat outside
RouterA(config-if)#exit
RouterA(config)#_
```

0:41:30 bağlandı OtoAlqila 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yanısı
(RouterA için konfigürasyon)

```
RouterB(config)#ip nat inside source static 192.168.4.2 10.1.1.2
RouterB(config)#inter
RouterB(config)#interface ethernet 0/0
RouterB(config-if)#ip nat inside
RouterB(config-if)#ip nat inside
RouterB(config-if)#exit
RouterB(config)#interface serial 0/1
RouterB(config-if)#ip nat out
RouterB(config-if)#ip nat outside
RouterB(config-if)#exit
RouterB(config)#_
```

0:43:26 harıldırı OtoAlqila 9600 8-N-1 Kavdır büvh SAYI Yakala Yazdırma yanısı
(RouterB için konfigürasyon)

Bu yapılan konfigürasyon ile A routerı için iç networkte bulunan 192.168.3.2 ip adresinin Router Serial interface'inden çıktıktan sonra 10.1.1.1 ip adresine, B Routeri için iç networkte bulunan 192.168.4.2 ip adresinin Router Serial interface'inden çıktıktan sonra 10.1.1.2 adresine dönüşmesini sağlamış olduk.

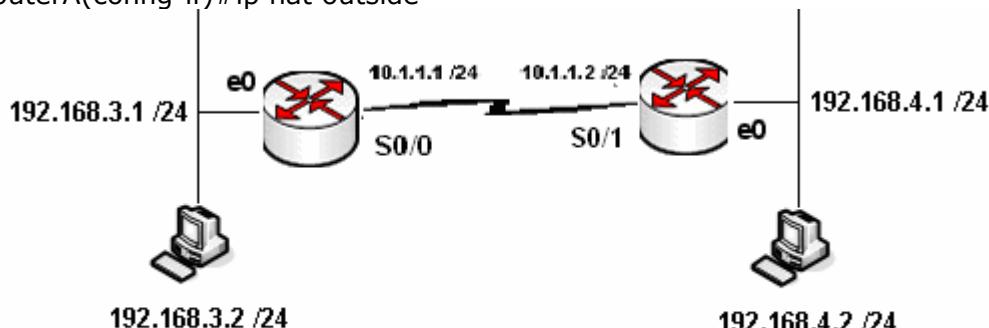
RouterA#sh ip nat translations	Pro Inside global	Inside local	Outside local	Outside global
	---	192.168.4.2	---	---
RouterA#_	10.1.1.1			

10:42:56 bağlandı OtoAlgıla 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı

(Router A için Ip NAT Translations Tablosu)

Benzer bir senaryo üzerinde Dynamic NAT uygulayabiliriz. Konfigürasyonda bazı farklılıklar olacaktır. Başta da bahsettiğimiz gibi Dynamic NAT için "outside" tarafında bir ip adres havuzu oluşturulmalı ve "inside" tarafta bir Access list yazılmalıdır.

```
RouterA#configure terminal
RouterA(config)# ip nat pool AcademyTech 10.1.1.1 10.1.1.5 netmask 255.255.255.0
RouterA(config)# access-list 1 permit 192.168.3.0 0.0.0.255
RouterA(config)#ip nat inside source list 1 pool AcademyTech
RouterA(config)#interface ethernet 0
RouterA(config-if)#ip nat inside
RouterA(config-if)#exit
RouterA(config)#interface serial 0
RouterA(config-if)#ip nat outside
```



Overload uygulamasında tüm bir network aynı ip adres üzerinden çıkarılabilir. Burada ip adresi belirtmek yerine değişken interface kullanmak gerekir.

Router B üzerinde NAT konfigürasyonumuz şu şekilde yapılacaktır:

```
RouterB(config)#access-list 1 permit 192.168.4.0 0.0.0.255
RouterB(config)#ip nat inside source list 1 ?
    interface Specify interface for global address
    pool      Name pool of global addresses

RouterB(config)#ip nat inside source list 1 interface serial 0/0
RouterB(config)#interface et
RouterB(config)#interface ethernet 0/0
RouterB(config-if)#ip nat inside
RouterB(config-if)#exit
RouterB(config)#interface ser
RouterB(config)#interface serial 0/0
RouterB(config-if)#ip nat out
RouterB(config-if)#ip nat outside
RouterB(config-if)#

```

2:33:50 bağlandı OtoAlgıla 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı

Overload uygulamasında da interface' lerin "inside" ya da "outside" oldukları belirtilmelidir.

```
!
router ospf 101
  network 10.1.1.0 0.0.0.255 area 0
  network 192.168.4.0 0.0.0.255 area 0
!
ip nat inside source list 1 interface Serial0/0 overload
ip classless
!
--More--
```

2:35:58 baglanıldı OtoAlgila 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı
(Overload işlemi Running Konfigürasyondan incelenebilir)

```
RouterB#show ip nat translations
Pro Inside global      Inside local        Outside local       Outside global
icmp 10.1.1.2:512     192.168.4.3:512   192.168.3.2:512   192.168.3.2:512
icmp 10.1.1.2:516     192.168.4.2:512   192.168.3.2:512   192.168.3.2:516
RouterB#
```

2:42:56 baglanıldı OtoAlgila 9600 8-N-1 Kaydır büyh SAYI Yakala Yazdırma yankısı
(IP Nat çevrimlerinin görüntülenmesi)

WAN Teknolojileri

WAN yani Wide Area Network Teknolojilerinin anlasilmasi icin bazi terimlerin onceden bilinmesi fayda saglayacaktir. Bu terimleri kisaca su sekillerde tanimlayabiliriz.

Customer Premises Equipment (CPE) : Musteri tarafindan kullanılan cihazlara genel olarak verilen addir.

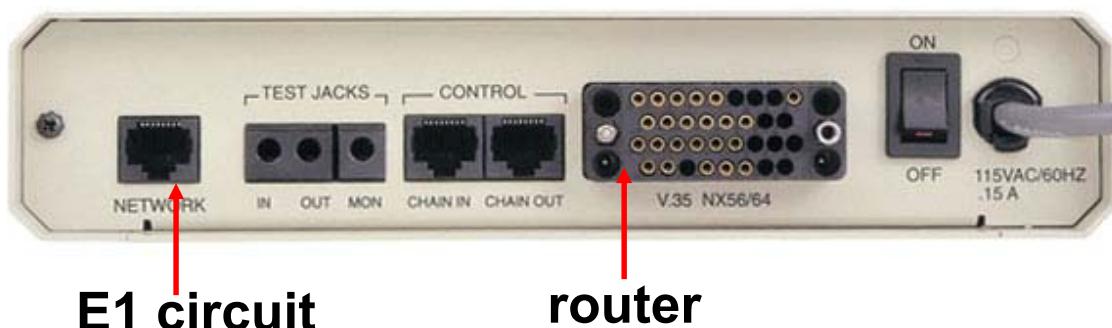
Demarcation (Demarc) : Servis saglayinin hizmet saglayacagi ve bu hizmet ile ilgili desteklerini surdurecegi, musteriye en yakin noktadir. Bu noktadan sonra olusabilecek olasi hatalar ile ilgili servis saglayaci sorumluluk kabul etmez, musterininkendisinin cozum bulmasi gereklidir.

Synchronous: Seri baglantilarda her iki noktada ki cihazların birbirlerine data gonderimi sirasinda hızlarını eşitlemeye çalışıkları durumdur şeklinde anlatılabilir. Bu tarz bir iletimde bizim için önemli olan nokta upload ve download hızlarının eşit olmasıdır.

Asynchronous: Dial-Up modemler örnek olarak gösterilebilir. Baglantılarda ki her nokta veri iletim hızlarının eşit olduğunu kabul eder ama eşit olmadığı durumlarda eşitemek için bir çalışma yapmaz. Bu durumda upload ve download hızları da birbirinden farklı olacaktır.

Data Services Unit/ Channel Services Unit (CSU/DSU) : DTE olan musteri ekipmanında clock üretimi saglayacak cihazlardır. WAN asılnda DTE networklerin DCE networkler üzerinden birbirlerine bağlanan LAN'ları toplulugudur şeklinde tanımlanabilir. Bu durumda ornegin DTE olan Routerlarda data iletimini başlatacak DCE bir cihaza ihtiyaç olacaktır, ornegin modem.

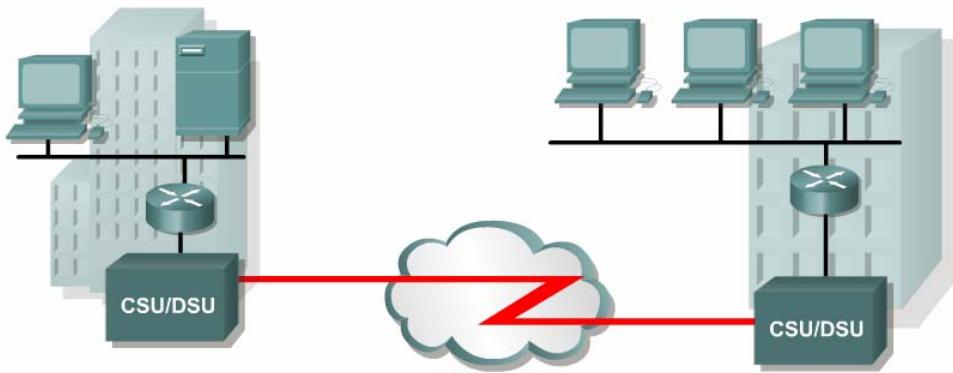
Lanoratuar ortamlarında kullandığımız özel kablolar ile DCE cihazlar yerine clock üretimini routerlara yaptırıyordu. Gerçek dünyada uctan uca DTE ve DCE olan kablolar ile bağlantı sağlamak imkansız olacağı açık olduğuna göre mevcut hatlar üzerinden iletimin sağlanabilmesi için DSU/CSU cihazlara ihtiyaç vardır.



WAN Baglantilari

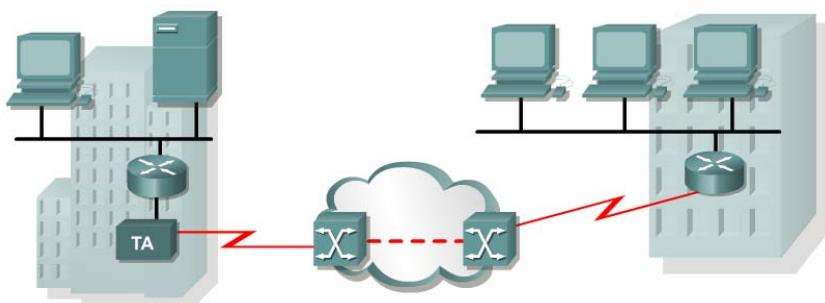
Kiralik Hatlar (Leased Lines): Kiralik hatlar tek bir firmaya atanmış, noktadan noktaya bağlantının sağlandığı senkron iletim hatlarıdır. Senkron iletişim kullanılmamasından dolayı upload ve download hızları eşittir. 45 Mbps'e kadar hız desteklemektedir. Bağlantı kurulduktan sonra hat devamlı aktif durumdadır.

Bu tur baglantılarda daha sonra detaylı inceleyeceğimiz HDLC, PPP veya SLIP protokolleri kullanılır.



Devre Anahtarlama (Circuit Switching): Asenkron iletisim c̄sidīdir ve dusuk bant genisligine ihtiyac duyulan durumlarda onerilebilir. Bu baglantılarda artık neredeyse tamamen vazgeçilen Dial-ip modemler veya ISDN hatları kullanılır.

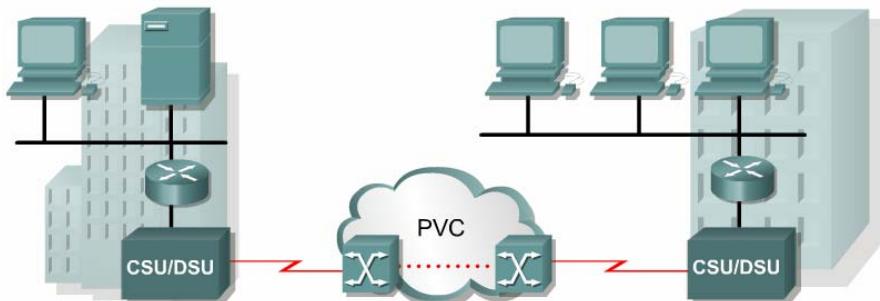
Burada modemler arasında bağlantı kurulduktan sonra hattın sürekli aktif kalması maliyeti artıracak için pek tercih edilmeyecektir ama söz gelimi zaten var olan bir hatta yedek olması ve o hat koptugunda devreye girip, hat tekrar aktif olduğundan devreden çıkış sağlanabildiğinde son derece kullanışlı olacaktır.



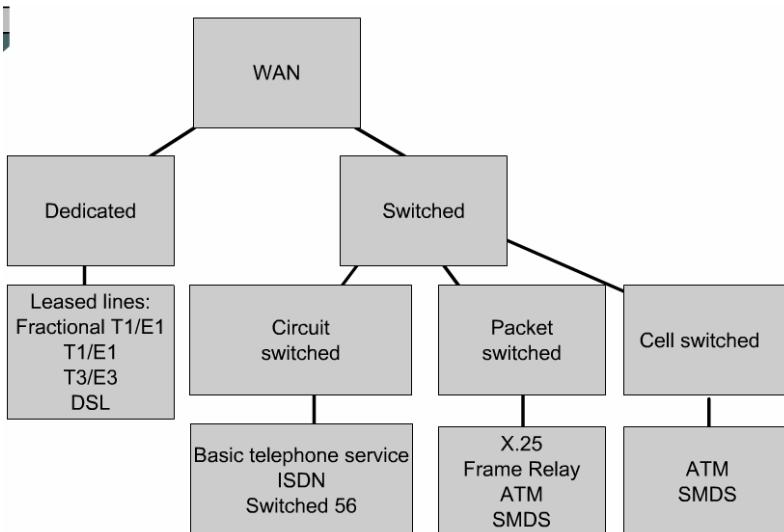
Asenkron iletisim sozkonusu olduğu için upload ve download hızları eşit değildir. HDLC, PPP ve SLIP protokollerini kullanabiliriz.

Paket Anahtarlama (Packet Switching): Geniş alan ağlarında sabit miktarlarda datanın gönderilmesi durumunda en uygun çözüm kiralık haralar olacaktır. Fakat ağımızda belirli zaman aralıklarında yüksek datalar gönderilirken bazen çok daha az data gönderilimi sozkonus ise bant genisliginin paylaşımı esasına göre tasarlanmış Packet Switching kullanılabilir.

Servis sağlayıcılar burada bir miktar bant genisliğini garanti ederken, garanti etmediği ama mümkün olduğunda kullanmasına izin verdiği daha yüksek bir bant genisligide sağlarlar.



Bu tur baglantılarda detaylı inceleyeceğimiz Frame-Relay ile birlikte X.25 ve ATM protokollerini kullanılmaktadır.



HDLC

HDLC; IBM tarafından geliştirilmiş standart bit tabanlı bir protokoldür. HDLC (High-Level Data-Link Control) ; data link katman protokolüdür. Cisco' nun geliştirdiği HDLC ile diğer üretici şirketlerin geliştirdiği HDLC iletişim kuramaz. Bu diger üreticiler icinde geçerlidir. Yani butun HDLC protokollerine üreticisine ozeldir diyebiliriz.

HDLC ; adres alanı , çerçeveye alanı, kontrol dizisi alanı (FCS), ve protokol tür alanını içeren çerçeveyelememi tanımlar. HDLC hata düzeltimi aynen Ethernet gibi yapar. HDLC alt bilgisinde FCS alanını kullanır. Alınan çerçevede hatalar olmuş ise çerçeveyi düzeltmeden iptal eder.

HDLC Çerçevelemesi :

HDLC ISO frame					
Flag	Address	Control	Data (Payload)	FCS	Flag
1 byte	1 byte	1 or 2 bytes	1500 bytes	2 (or 4) bytes	1 byte

İki router'ı HDLC kullanarak haberleşirmek için aşağıdaki komut satırları kullanılır.

Serial Interface de encapsulation 'ı HDLC olarak ayarlamak :

```

Router(config)#int ser 0
Router(config-if)#encapsulation hdlc
Router(config-if)#
  
```

Status

Yaptığımız konfigürasyonu görmek için show interface serial0 kullandık :

```

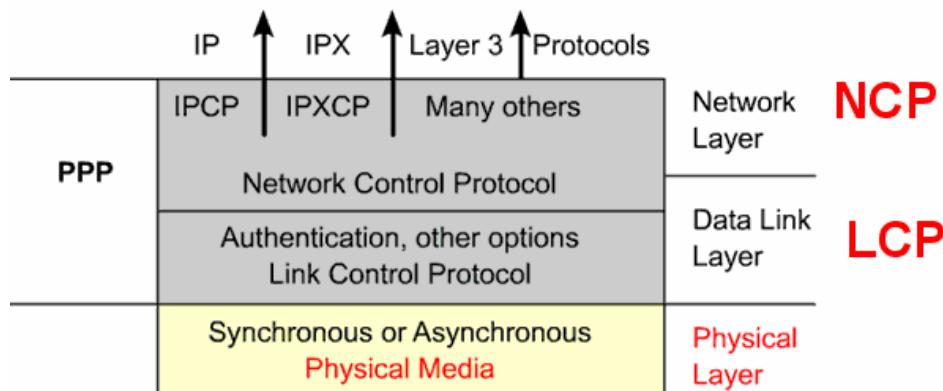
Router#sh int serial0
Serial0 is up, line protocol is up
  Hardware is HD64570
  Internet address is 172.19.1.13/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 1000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of show interface counters never
  
```

PPP

PPP HDLC protokolune gore butun uretici firmalarin Routerlari tarafindan desteklendiği icin daha cok tercih edilen encapsulation metodudur.

PPP' butun olarak incelemek biraz zordur cunki aslinda PPP iki tane aly protokolden olusur. Bunlar soyle siralayabiliriz;

1. Link Control Protocol (LCP)
2. Network Control Protocol (NCP)



LCP point to point baglantinin saglanması icinkullanılırken NCP network katmani protokollerinin konfigurasyonu icin kullanılır.

LCP, Authentication, sikistirma, hata kontrol ve birden fazla yol arasında load balancing gibi ozellikleri saglar.

PPP oturumlar opsiyonel olan secimler ile birlikte 5 adimda olusur.

1. Link establishment - (LCPs)
2. Authentication - Optional (LCPs)
3. Link quality determination - Optional (LCPs)
4. Network layer protocol configuration (NCPs)
5. Link termination (LCPs)

```

Router#show interfaces serial0/0
Serial0/0 is up, line protocol is up
  Hardware is HD64570
  Internet address is 10.140.1.2/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
  rely 255/255, load 1/255
  Encapsulation PPP, loopback not set, keepalive
  set (10 sec)
    LCP Open ← LCP
    Open: IPCP, CDP/CP ← NCP
  Last input 00:00:05, output 00:00:05, output
  hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0
  drops
  5 minute input rate 0 bits/sec, 0 packets/sec

```

PPP Authentication

Router 'ın seri interface 'lerinde PPP tanımı yapmak için "encapsulation ppp" komutu kullanılır. Bağlantının sağlandığı her iki uçtaki interface' lerin ikisinde de PPP aktif yapılmalıdır.

```
Router(config)#interface ser 0
Router(config-if)#encapsulation ppp
Router(config-if)#exit
Router(config)#+
```

Ayrıca Router 'lara "hostname" komutunu kullanarak bir isim verilmelidir.

```
Router(config)#hostname kadıköy
kadıköy(config)#+
```

Ve karşı tarafın bağlantı yapacağı sırada kullanacağı kullanıcı adı ve şifresi global konfigürasyon modundayken tanımlanmalıdır. Kullanılan şifre tüm router 'larda aynı olmak zorundadır. Daha sonra bir kimlik doğrulama metodu da belirlemek gereklidir. Bunun için öncelikle interface moda girilerek "ppp authentication" komutun kullanılır.

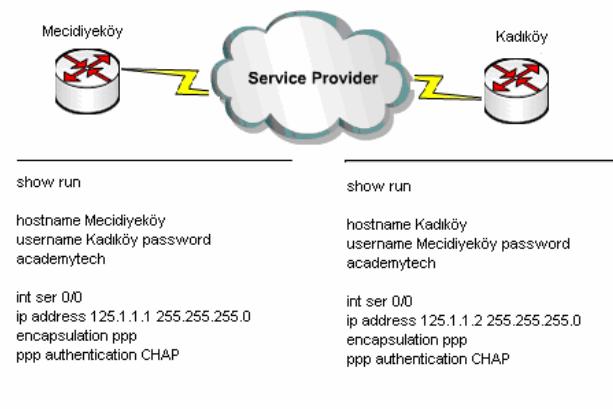
Daha önce bahsettiğimiz PAP yada CHAP metodlarından biri seçilir. Dikkat edilmesi gereken seçilen metodun her iki router da ortak seçilmesidir. Eğer bir router da PAP diğer router da CHAP seçildiyse iletişim kurulamayacaktır.

```
kadıköy(config)#username meciidyeköy password academytech
kadıköy(config)#int ser 0
kadıköy(config-if)#ppp authentication chap
kadıköy(config-if)#+
```

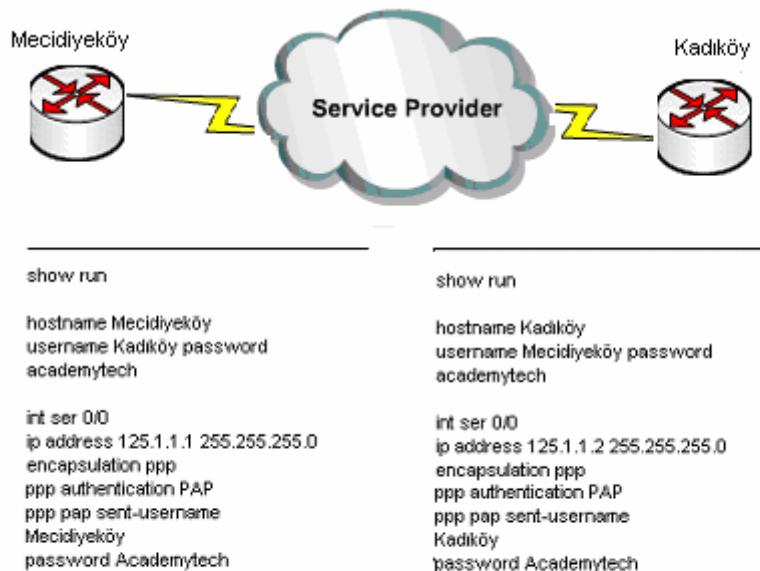
Eğer authentication metodunu PAP seçmiş olsaydık , interface içerisinde PAP 'ı aktif etmemiz gerekecekti. Çünkü Cisco IOS 11.1 ve sonrasında PAP default olarak disable durumdadır.

```
kadıköy(config)#int ser 0
kadıköy(config-if)#ppp pap sent-username meciidyeköy password academytech
```

CHAP KONFIGÜRASYONU :

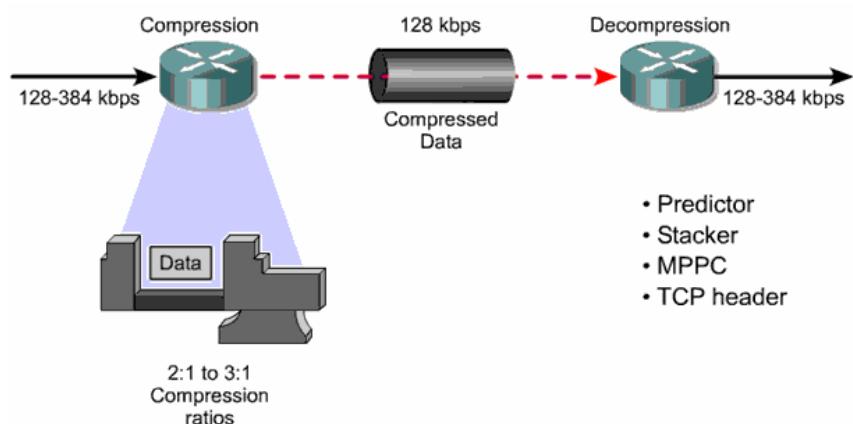


PAP KONFIGÜRASYONU :



PPP Compression

PPP datayı sıkıştırabilme özelliği ile düşük bant genişliğinde daha yüksek performans sağlayabilmektedir.



4 farklı Compression tipi vardır.

1. Predictor
2. Stacker
3. MMPC
4. Tcp Header Sıkıştırma

Hatalı PPP Konfigurasyon Ornekleri

Mismatched WAN encapsulations



```

hostname Pod1R1
username Pod1R2 password Cisco
interface serial 0
ip address 10.0.1.1 255.255.255.0
encapsulation ppp

```

```

hostname Pod1R2
username Pod1R1 password cisco
interface serial 0
ip address 10.0.1.2 255.255.255.0
encapsulation HDLC

```

Mismatched IP addresses



```

hostname Pod1R1
username Pod1R2 password cisco
interface serial 0
ip address 10.0.1.1 255.255.255.0
encapsulation ppp
ppp authentication chap

```

```

hostname Pod1R2
username Pod1R1 password cisco
interface serial 0
ip address 10.2.1.2 255.255.255.0
encapsulation ppp
ppp authentication chap

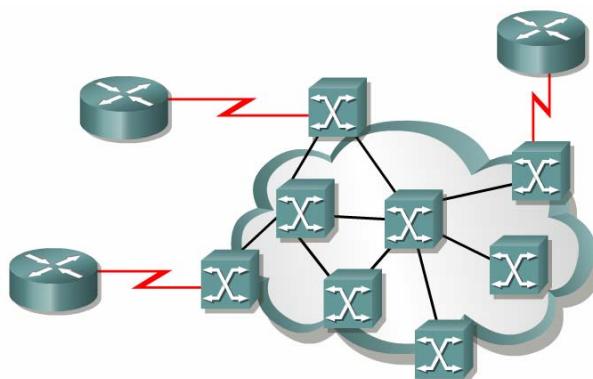
```

Frame Relay

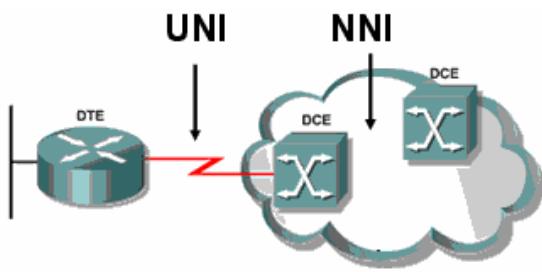
Frame Relay Packet Switching teknolojisyle tımlı bir WAN servisidir ve OSI referans modelinin Data-Link Katmanında çalışır. Frame Relay HDLC'ın bir alt bileşeni olan Link Access Procedure for Frame Relay (LAPF) protokolünü kullanır.

Burada data帧eler halinde müşterinin DTE cihazlarından diğer nokta veya noktalardaki DTE cihazlarına DCE cihazlar üzerinden taşınır. Burada ki DCE cihazlar ya da DCE network telekom firmalarının sağladığı network ve cihazları, kontrol ve konfigürasyonu bu firmalar tarafından yapılır.

Frame Relay networklerinde genellikle 56 Kbps ve 2 Mbit arasında genişlikleri kullanılmaktadır fakat 45 Mbit'e kadar desteklenmektedir.



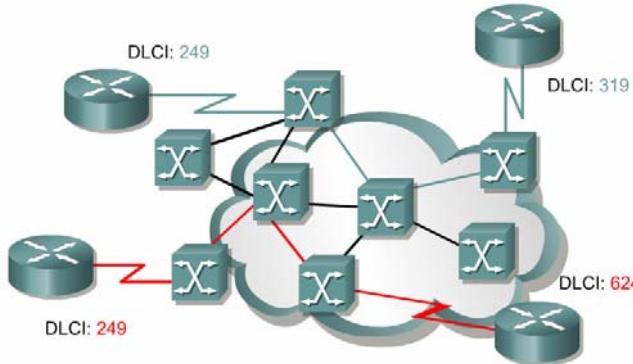
Frame Relay networklerinde müşteri ve servis sağlayıcı arasında ki bağlantıya User-To Network Interface (UNI), birbirinden farklı servis sağlayıcılara ait cihazların bağlandıkları noktalara ise Network-To-Network Interface (NNI) denir.



Frame Relay Networkleri oluştururken servis sağlayının vereceği DLCI numaralarının tanımlanması önemlidir. Çünkü servis sağlayıcı yada telekom bu DLCI numaralarına kendi switchleri üzerinden yol verecek ve iki nokta arasında sanal bir devre oluşturarak bağlantının kurulmasını sağlayacaktır.

Burada bahsettiğimiz sanal devreler pek kullanılmayan Switched Virtual Circuits (SVCs) ve Permanent Virtual Circuits (PVCs) olmak üzere ikiye ayrılır.

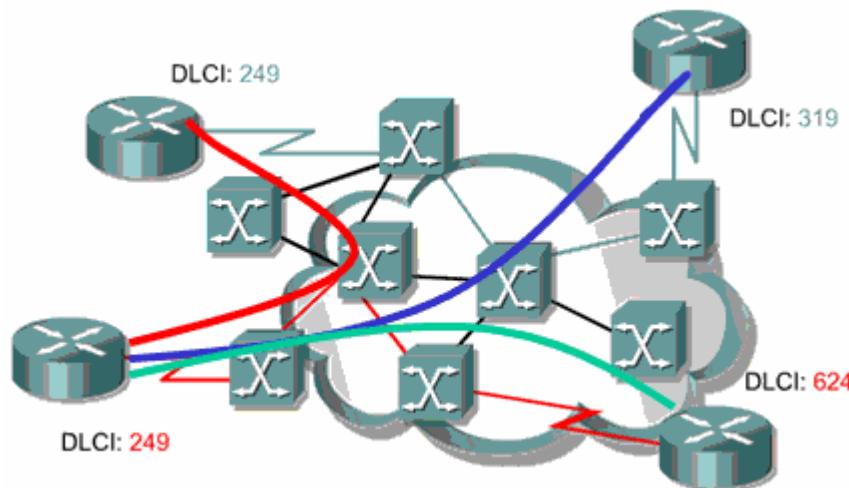
SVCs'lerde iki nokta arasında kurulan bağlantı için geçerli olan yol dinamik olarak değişmekteken PVCs'lerde iki nokta arasında sabır bir yol tanımı yapılır ve manuel olarak değiştirilmemiş surekli o yol kullanılır.



Sözgelimi şekilde kırmızı olarak işaretlenen yol söz konusu iki nokta arasındaki PVC'yi belirtir. Fakat burada SVC ile bir çalışmadan bahsetseydik mevcut yol yerine alternatif yollardan biri de kullanılabiliyor olacaktı.

Cunki SVC'ler geçici olarak oluşturulurlar ve bu bağlantıların oluşturulması için bir çevrim (call setup) gerekmektedir.

Frame Relay konusun bir router birden fazla noktası arasında birebir bağlantı yapılması gereken durumlarda her noktası için ayrı PVC'ler oluşturulabilir. Merkez noktası üzerinde yapılacak ve her noktaya erişim için farklı olan DLCI numaraları ile bu mümkün olacak ve o dakikadan itibaren merkez Frame Relay router bütün noktalara aynı anda hizmet verebilecektir.



Frame Relay Headers

Frame Relay ile konfigüre edilmiş routerlar iki farklı Frame Relay Header'i desteklerler.

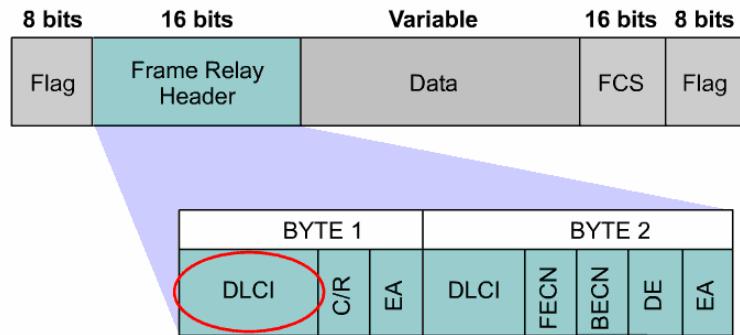
1. Cisco
 2. IETF

Cisco adindan da anlasildigi gibi Cisco ozeldir ve ortam da Cisco disinda ureticilere ait Routerlar varsa kullanilamaz. Bununla birlikte Frame Relay framelerine 4 Byte lik headerler ekledigi icin önerilen degildir.

IETF ise birden fazla ureticiyi destekler ve frame'lere Cisco' nun aksine sadece 2 byte' lik headerlar ekler.

Bu headerleriniceriginde bizim icin oneli olacak DLCI' lar vardir.

IETF Frame Relay Frame

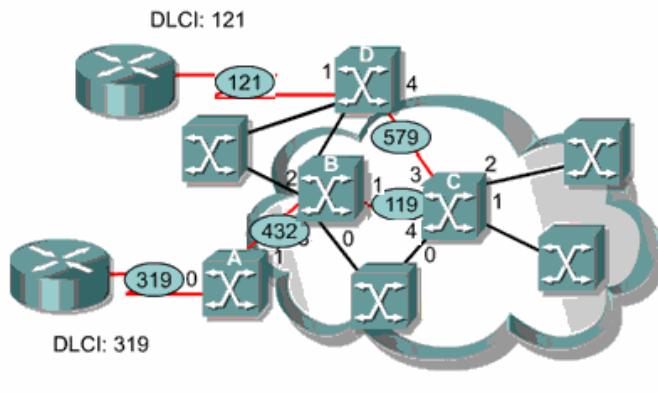


DLCI

Data Link Connection Identifier' in kısaltması olan DLCI musteri cihazı ve frame relay switch arasındaki sanal devreyi tanımlaya yarar.

DLCI numaraları servis sağlayıcılar tarafından belirlenen mantıksal adreslerdir denebilir. 0-15 ve 1008 – 1023 arasında ki numaralar özel amaçlar için ayrıldığından servis sağlayıcılar tarafından 16-1007 arasındaki numaralardan seçilerek atama yapılır.

DLCI				
A	VC	Port	VC	Port
319	0	432	1	
B				
B	VC	Port	VC	Port
432	3	119	1	
C				
C	VC	Port	VC	Port
119	4	579	3	
D				
D	VC	Port	VC	Port
579	0	121	1	



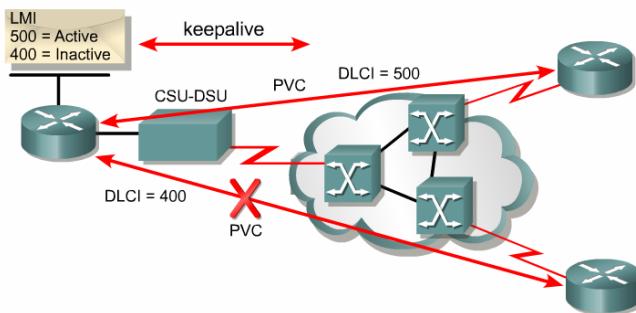
Sekilde A, B, C ve D switchleri üzerindeki DLCI yönlendirmeleri incelediğinde DLCI mantığı daha iyi anlaşılacaktır.

LMI

Local Management Interface (LMI) DTE cihazlar ve Frame Relay switchler arasındaki signaling standartıdır.

Cisco Routerlar 3 çeşit LMI Type'i destekler.

1. Cisco
2. Ansi
3. q933a



Frame Relay networki ve DTE Router için LMI type aynı olmadığı takdirde çalışmayacaktır. Türkiye'de kullanılan LMI Type Ansi' dir. Routerda Cisco IOS 11.2 ve üzeri varsa LMI Type tanımlaya gerek kalmaz, Router Frame Relay networkundeki LMI Type'i algılar.

Frame Relay Switchler konfigüre edilen PVC'lerin durumlarını belirtmek için LMI' i kullanırlar. PVC'ler 3 ayı durumda olabilirler.

1. Active State: Routerların data transferi yapabildiği, bağlantının aktif olduğunu belirttiğimiz durumdur.

2. Inactive State: Frame Relay switch ile Localimiz arasında ki bağlantının aktif olduğu ama uzaktaki Router ile uzaktaki Frame Relay switch bağlantısının düzgün çalışmaması durumdur.

4. **Deleted State:** CPE ve Frame Relay switch arasında herhangi bir servisin çalışmaması durumdur.

```
1w2d: Serial0/0 (in): Status, myseq 142
1w2d: RT IE 1, length 1 type 0
1w2d: KA IE 3, length 2 yourseg 142, myseq 142
1w2d: PVC IE 0x7, length 0x6, dlci 100, status 0x2, bw0
(debug frame-relay Imi)
```

Burada 0x2 aktif durumu gösterir, diğer durumlar şu şekildeidir;

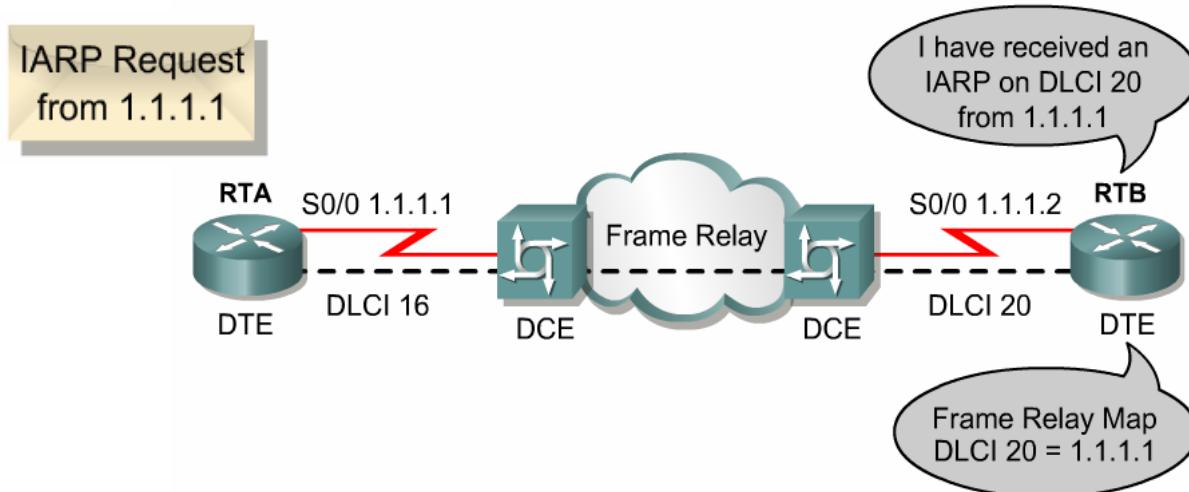
0x0: Inactive

0x4: Deleted

DLCI Mapping

Frame Relay networklerinin konfigürasyonu sırasında önemli bir adımda servis sağlayıcılarının Frame Relay switchlerinde yol verdikleri DLCI numaralarının next hop 3. katman adreslerine map edilmesidir.

Burada mano işlemi dinamik ve static olmak üzere iki şekilde yapılabilir. Static map işleminde frame relay map komutu kullanılır. Dinamik map işleminde ise Inverse ARP protokolu çalışır. Burada Inverse ARP her DLCI için Inverse ARP Request mesajı gönderir ve aldığı cevap ile data-link katman adresi DLCI numarası ve Network Katmanı adresi Next Hop ip adresini map eder.



Kısaca Inverse ARP LAN'ı da ARP protokolü gibi çalışır.

Static Map

```
Router(config-if)#frame-relay map protocol protocol-address
    dlci [broadcast] [ietf | cisco]
```

Buradaki ip adresi remote ip adresi DLCI numarası ise local DLCI numarasıdır.

```
Router(config-if)#frame-relay map ip 10.1.1.1 101 broadcast
```

Dinamik Map

```
Router(config-if)# frame-relay interface-dlci dlci-number
```

Buradaki DLCI numarası local DLCI'dır.

```
Router(config-if)#frame-relay interface dlci 100
```

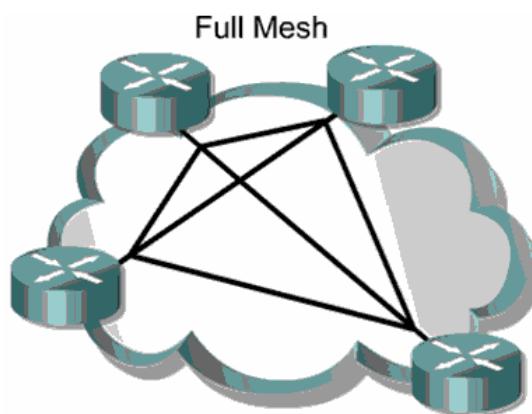
Frame Relay networklerinde Frame Relay encapsulation kullanılır. Cisco ve Ietf olmak üzere 2 ayrı standardı vardır, default olarak Cisco'dur. Sistemde Cisco dışında Routerlar var ise RFC 1490 ile tanımlanmış Ietf standartı kullanılmalıdır.

```
RTB(config)#interface serial 0/0
RTB(config-if) #encapsulation frame-relay
RTB(config-if) #frame-relay map ip 131.108.123.2 48 broadcast
RTB(config-if) #frame-relay map ip 131.108.123.3 49 broadcast ietf
RTB(config-if) #frame-relay map ip 131.108.123.4 50 broadcast
```

Encapsulation Frame Relay seçildikten sonra Frame Relay ao komutunda encapsulation seçilmeyebilir, bu durumda frame relay encapsulation geçerli olacaktır.

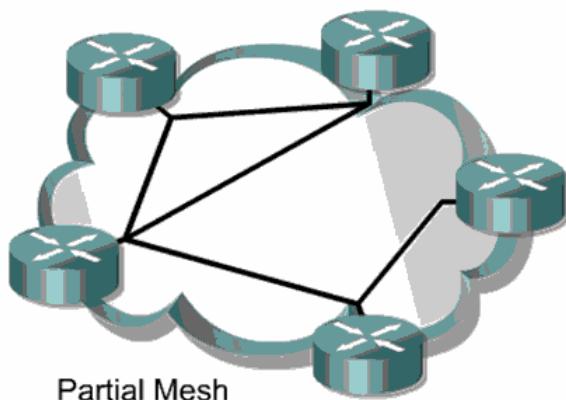
Frame Relay Map satırlarının ikincisindeki gibi farklı bir encapsulation seçilirse geçerli olan o olacaktır. Örneğimiz de ikinci satır için geçerli olan encapsulation ietf'dir.

Frame Relay Topolojileri



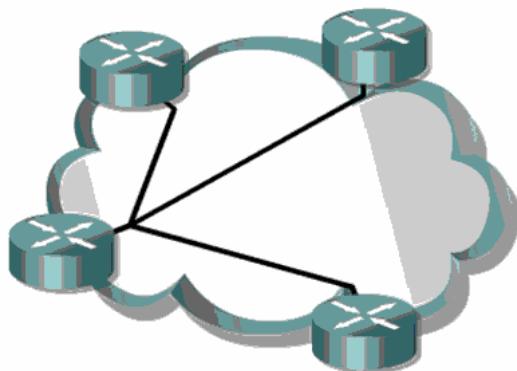
Mesh Topolojisinde esas olan Tüm noktalar arasında ayrı birer PVC olmasıdır. Oldukça pahalı bir topolojidir. Fakat bağlantılarından biri down olduğunda bile bir çok alternatif yoldan hedefe ulaşılabilir.

Full Mesh ve Partial Mesh olarak ikiye ayrılır.



Hub and Spoke Topoloji en çok kullanılan Frame Relay topolojidir ve Star Topoloji olarak da anılır. Bu topoloji genellikle birden fazla uzak networkun merkezi bir routera bağlanması ile sekillenir.

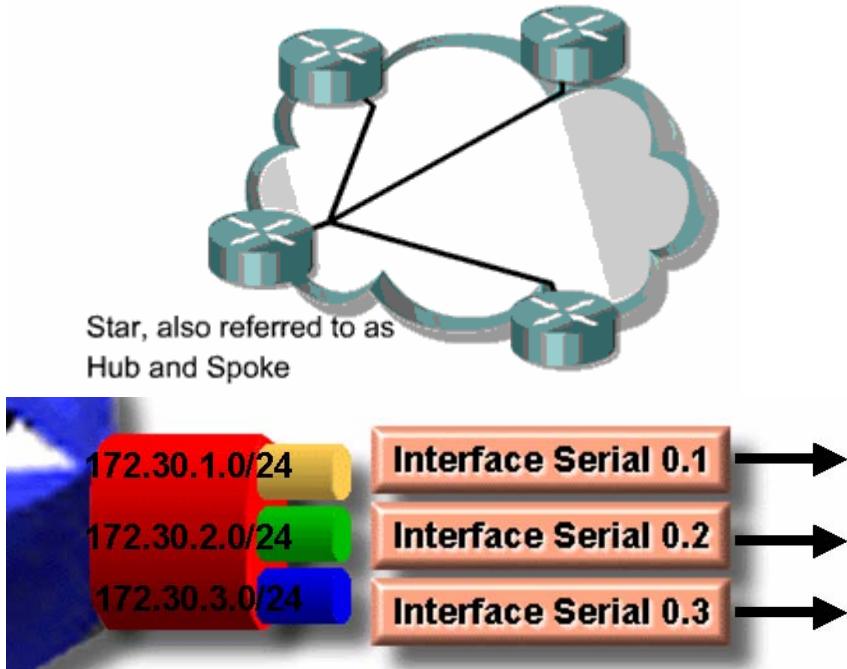
Hub and Spoke



Burada merkez router multipoint bağlantı yada point to point bağlantı sağlayabilir.

Frame Relay Sub-Interface Konfigurasyonu

Frame Relay networklerinde sozgelimi merkezde olan bir router birden fazla sayıda sahip subeye hizmet verebili, subeyle baglantiyi saglanabili. Burda interfacein altında sub interfaceler olusturmak gerekir.



Burada her sub-interface farkli bir networke ait ve farkli bir remote baglanti icinde. Her baglanti icin ayri PVC' ler mevcut.

Hun and Spoke olarak adlandirilan Frame Relay topolojilerinde kullanilan bu yontem ile ilgili uygulama ilerde yapilacaktir. Simdilik ornek olmasi acisindan cisco.com ' dan alionan konfihurasyonu veriyorum.

```
RTA(config)#interface serial S0/0.1 multipoint
RTA(config-subif)#ip address 1.1.1.1 255.255.255.0
RTA(config-subif)#frame-relay interface-dlci 18
RTA(config-fr-dlci)#exit
RTA(config-subif)#frame-relay interface-dlci 19
RTA(config-fr-dlci)#exit
RTA(config-subif)#exit
RTA(config)#interface serial S0/0.2 point-to-point
RTA(config-subif)#ip address 2.1.1.1 255.255.255.0
RTA(config-subif)#frame-relay interface-dlci 20
RTA(config-fr-dlci)#^Z
```

Hub and Spoke topology Frame Relay networklerinin en çok kullanılan seklidir. Point to multi point veya sub-interfaceler ile point to point olarak tasarlabilir. Fakat bu topology, Point to multipoint networklerde routing islemi icin Routing Protokoller kullanilmassa Split Horizon kuralindan dolayi sorun yasatacaktir.

Cunku Split Horizon kurali geregi bir Router aldigı update' i aldigı interfaceden geri gondermez. Bu durumda Split Horizon kurali devre disi bırakılmalıdır.

```
Router(config-if)#no ip split-horizon
```

Split Horizon kurali Link State protokollerini ornegin OSPF protokolune etkilemez.

Frame Relay Show Komutları:

Asagidaki show komutlari Cisco' nun CNAP egitimi icin ongordugu program slaytlarindan alınmıştır.

```
Router#show frame-relay pvc 110

PVC Statistics for interface Serial0 (Frame Relay DTE)

DLCI = 110, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE,
INTERFACE = Serial0
input pkts 14055      output pkts 32795    in bytes 1096228
out bytes 6216155     dropped pkts 0      in FECN pkts 0
in BECN pkts 0        out FECN pkts 0     out BECN pkts 0
in DE pkts 0          out DE pkts 0
out bcast pkts 32795  out bcast bytes 6216155
```

```
Router#show frame-relay map
Serial2 (up): IP 131.108.122.2 dlci 20(0x14,0x0440),
dynamic
CISCO, BW= 56000, status defined, active
```

```
Router#show frame-relay lmi
```

```
LMI Statistics for interface Serial0 (Frame Relay DTE)
LMI TYPE =
CISCO
Invalid Unnumbered info      0 Invalid Prot Disc 0
Invalid dummy Call Ref      0 Invalid Msg Type 0
Invalid Status Message       0 Invalid Lock Shift 0
Invalid Information ID       0 Invalid Report IE Len 0
Invalid Report Request       0 Invalid Keep IE Len 0
Num Status Enq. Sent 113100  Num Status msgs Rcvd 113100
Num Update Status Rcvd 0     Num Status Timeouts 0
```

```
show interface serial 0/0
Serial0 is up, line protocol is up
Hardware is CD2430 in sync mode
MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
LMI enq sent 112971, LMI stat recv 112971, LMI upd recv 0, DTE LMI up
LMI enq recv 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
FR SVC disabled, LAPF state down
Broadcast queue 0/64, broadcasts sent/dropped 32776/0, interface broadcasts 1
Last input 00:00:00, output 00:00:03, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
<Output Omitted>
```

Frame Relay Switch Konfigurasyonu

Laboratür ortamında Frame Relay uygulamaları için Frame Relay Switche ihtiyacı vardır. Fakat Frame Relay switch olmadığı durumlarda bir Router Frame – Relay Switch olarak konfigüre edilebilir.

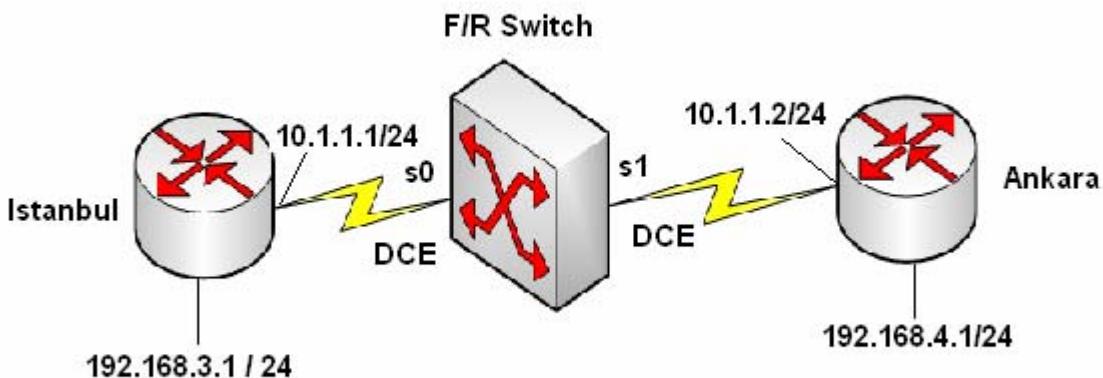
Bunun için Routerlara takılan DTE kablolarına DCE kablolarla Frame Relay Switch olarak konfigüre edilecek Router'a bağlanır ve interfacelerine "clock rate" komutu verilir. Burada interfacelere interface type' in DCE olduğunda söylenir.

Örnek:

```
interface Serial0
no ip address
encapsulation frame-relay
no fair-queue
clockrate 64000
frame-relay lmi-type ansi
frame-relay intf-type dce
frame-relay route 100 interface Serial1 101
```

Frame Relay Point-To-Point Konfigurasyonu

Frame Relay konfigurasyonumuzda kullanacağımız topoloji şu şekildedir;



Burada DLCI numaraları İstanbul için 100, Ankara için 101'dir ve bir Router laboratuvar ortamında Frame Relay Switch olarak konfigüre edilmişdir. IP adresleri atandıktan sonra, Frame Relay çalışma için;

İstanbul Routerinde;

```
Router(config)#interface Serial0/0
Router(config-if)#encapsulation frame-relay
Router(config-if)#frame-relay interface-dlci 100
Router(config-if)# frame-relay lmi-type ansi
```

Ankara Routerinda;

```
Router(config)#interface Serial0/1
Router(config-if)#encapsulation frame-relay
Router(config-if)# frame-relay interface-dlci 101
Router(config-if)#frame-relay lmi-type ansi
```

Konfigurasyonları yapılmıştır.

İstanbul Routeri Running-Config

```
sh running-config
Building configuration...

Current configuration : 636 bytes
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
!
!
!
memory-size iomem 10
ip subnet-zero
!
!
!
interface Ethernet0/0
 ip address 192.168.3.1 255.255.255.0
!
interface Serial0/0
 ip address 10.1.1.1 255.255.255.0
 encapsulation frame-relay
 no fair-queue
 frame-relay interface-dlci 100
 frame-relay lmi-type ansi
!
interface BRI0/0
 no ip address
 shutdown
 isdn x25 static-tei 0
!
ip classless
ip route 192.168.4.0 255.255.255.0 10.1.1.2
ip http server
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

Ankara Routeri Running-Config

```
sh run
Building configuration...

Current configuration : 632 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
memory-size iomem 10
ip subnet-zero
!
!
interface Ethernet0/0
 ip address 192.168.4.1 255.255.255.0
 half-duplex
!
interface Serial0/0
 no ip address
 shutdown
!
interface Serial0/1
 ip address 10.1.1.2 255.255.255.0
 encapsulation frame-relay
 frame-relay interface-dlci 101
 frame-relay lmi-type ansi
!
ip classless
ip route 192.168.3.0 255.255.255.0 10.1.1.1
ip http server
!
!
!
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

Frame-Relay Switch Routeri Running-Config

```

sh run
Building configuration...

Current configuration:
!
version 11.2
no service password-encryption
no service udp-small-servers
no service tcp-small-servers
!
hostname Router
!
frame-relay switching
!
interface Ethernet0
  no ip address
  shutdown
!
interface Serial0
  no ip address
  encapsulation frame-relay
  no fair-queue
  clockrate 64000
  frame-relay lmi-type ansi
  frame-relay intf-type dce
  frame-relay route 100 interface Serial1 101
!
interface Serial1
  no ip address
  encapsulation frame-relay
  clockrate 64000
  frame-relay lmi-type ansi
  frame-relay intf-type dce
  frame-relay route 101 interface Serial0 100
!
no ip classless
!
line con 0
line aux 0
line vty 0 4
login
!
end

```

İstanbul Routeri Frame Relay PVC ve Frame Relay Map

```

ISTANBUL#show frame-relay pvc
PVC Statistics for interface Serial0/0 (Frame Relay DTE)

      Active     Inactive     Deleted     Static
Local       1           0           0           0
Switched    0           0           0           0
Unused      0           0           0           0

DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0
  input pkts 36          output pkts 33          in bytes 2766
  out bytes 2644         dropped pkts 0          in FECN pkts 0
  in BECN pkts 0         out FECN pkts 0         out BECN pkts 0
  in DE pkts 0           out DE pkts 0
  out bcast pkts 2       out bcast bytes 68
  pvc create time 00:24:43, last time pvc status changed 00:22:03
ISTANBUL#show frame-relay map
Serial0/0 (up): ip 10.1.1.2 dlci 100(0x64,0x1840), dynamic,
                 broadcast,, status defined, active
ISTANBUL#

```

Ankara Routeri Frame Relay PVC ve Frame Relay Map

```
ANKARA#show frame-relay pvc
PVC Statistics for interface Serial0/1 (Frame Relay DTE)
      Active     Inactive     Deleted     Static
Local       1           0           0           0
Switched    0           0           0           0
Unused      0           0           0           0

DLCI = 101, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/1

  input pkts 32          output pkts 32          in bytes 2610
  out bytes 2573         dropped pkts 0          in pkts dropped 0
  out pkts dropped 0    out bytes dropped 0
  in FECN pkts 0         in BECN pkts 0          out FECN pkts 0
  out BECN pkts 0        in DE pkts 0           out DE pkts 0
  out bcast pkts 7       out bcast bytes 469
  pvc create time 00:23:13, last time pvc status changed 00:23:13
ANKARA#show frame-relay map
Serial0/1 (up): ip 10.1.1.1 dlci 101(0x65,0x1850), dynamic,
                  broadcast,, status defined, active
ANKARA#_
```

İn-32.23.kaçanlık Details QANAS-N1 Kavşak İhvanı SAVT Vakala Vazgeçme Vakası

Frame Relay Switch PVC ve Route

```
FRSW#show frame-relay route
Input Intf   Input Dlci   Output Intf   Output Dlci   Status
Serial0      100        Serial1      101        active
Serial1      101        Serial0      100        active
FRSW#_
```

Building configuration...

```
PVC Statistics for interface Serial0 (Frame Relay DCE)
DLCI = 100, DLCI USAGE = SWITCHED, PVC STATUS = ACTIVE, INTERFACE = Serial0

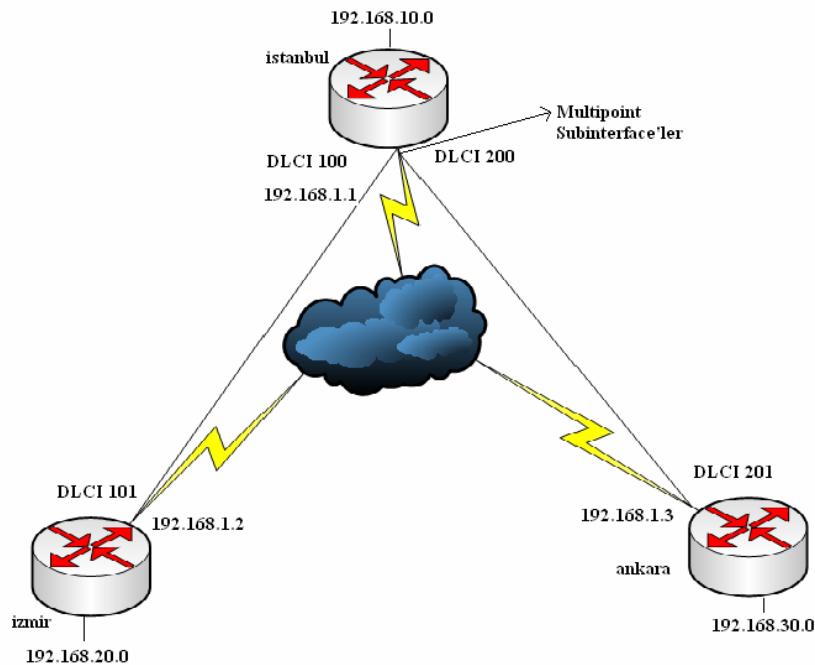
  input pkts 34          output pkts 36          in bytes 2678
  out bytes 2766         dropped pkts 1          in FECN pkts 0
  in BECN pkts 0         out FECN pkts 0          out BECN pkts 0
  in DE pkts 0           out DE pkts 0
  out bcast pkts 0       out bcast bytes 0
  pvc create time 00:28:58, last time pvc status changed 00:20:09
  Num Pkts Switched 34
```

PVC Statistics for interface Serial1 (Frame Relay DCE)

```
DLCI = 101, DLCI USAGE = SWITCHED, PVC STATUS = ACTIVE, INTERFACE = Serial1

  input pkts 37          output pkts 34          in bytes 2800
  out bytes 2678         dropped pkts 0          in FECN pkts 0
  in BECN pkts 0         out FECN pkts 0          out BECN pkts 0
  in DE pkts 0           out DE pkts 0
  out bcast pkts 0       out bcast bytes 0
  pvc create time 00:29:01, last time pvc status changed 00:22:59
  Num Pkts Switched 36
FRSW#_
```

Frame Relay Hub and Spoke MultiPoint Konfigurasyonu



Routerların Konfigurasyon Dosyaları

Router İstanbul

```
version 12.0
!
hostname Istanbul
!
interface Serial0
no ip address
encapsulation frame-relay
no frame-relay inverse-arp
!
interface Serial0.1 multipoint
ip address 192.168.1.1 255.255.255.0
no ip split-horizon
frame-relay interface-dlci 100
frame-relay interface-dlci 200
!
interface FastEthernet0
ip address 192.168.10.1 255.255.255.0
no ip directed-broadcast
no keepalive
!
router rip
network 192.168.0.0
!
```

Router Izmir

```

version 12.0
!
hostname Izmir
!
interface Serial0
  no ip address
  encapsulation frame-relay
  frame-relay lmi-type ansi
!
interface Serial0.1 point-to-point
  ip address 192.168.1.2 255.255.255.0
  no ip directed-broadcast
  frame-relay interface-dlci 101
!
interface FastEthernet0
  ip address 192.168.20.1 255.255.255.0
  no ip directed-broadcast
  no keepalive
!
router rip
  network 192.168.0.0
!
```

Router Ankara

```

version 12.0
!
hostname Ankara
!
interface Serial0
  no ip address
  encapsulation frame-relay
!
interface Serial0.1 point-to-point
  ip address 192.168.1.3 255.255.255.0
  frame-relay interface-dlci 201
!
interface FastEthernet0
  ip address 192.168.30.1 255.255.255.0
  no ip directed-broadcast
  no keepalive
!
router rip
  network 192.168.0.0
!
```

Frame Relay Switch

```

version 12.0
!
hostname FrameSwitchE
!
frame-relay switching
!
interface Serial0
  no ip address
  encapsulation frame-relay
  clockrate 56000
  frame-relay lmi-type ansi
  frame-relay intf-type dce
  frame-relay route 100 interface Serial1 101
  frame-relay route 200 interface Serial2 201
!
interface Serial1
  no ip address
  encapsulation frame-relay
  clockrate 56000
  frame-relay lmi-type ansi
  frame-relay intf-type dce
  frame-relay route 101 interface Serial0 100
!
interface Serial2
  no ip address
  encapsulation frame-relay
  clockrate 56000
  frame-relay lmi-type ansi
  frame-relay intf-type dce
  frame-relay route 201 interface Serial0 200

```

Frame Relay Map ve PVC

FrameSwitchE#show frame-relay route

Input Intf	Input Dlci	Output Intf	Output Dlci	Status
Serial0	100	Serial1	101	active
Serial0	200	Serial2	201	active
Serial1	101	Serial0	100	active
Serial2	201	Serial0	200	active

Istanbul#show frame-relay map

```

Serial0.1 (up): ip 192.168.1.2 dlci 100(0x64,0x1840), dynamic,
  broadcast,, status defined, active
Serial0.1 (up): ip 192.168.1.3 dlci 200(0xC8,0x3080), dynamic,
  broadcast,, status defined, active

```

Istanbul#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	2	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.1

input pkts 140	output pkts 77	in bytes 24656
out bytes 7774	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 69	out bcast bytes 7038	
pvc create time 00:31:06, last time pvc status changed 00:30:36		

DLCI = 200, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.1

input pkts 128	output pkts 103	in bytes 18760
out bytes 11810	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 72	out bcast bytes 8594	
pvc create time 00:31:07, last time pvc status changed 00:30:37		

Izmir#show frame-relay map

Serial0.1 (up): point-to-point dlci, dlci 101(0x65,0x1850), broadcast
status defined, active

Izmir#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	1	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 101, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.1

input pkts 58	output pkts 65	in bytes 6252
out bytes 9602	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 55	out bcast bytes 8562	
pvc create time 00:20:34, last time pvc status changed 00:20:34		

Ankara#show frame-relay map

```
Serial0.1 (up): point-to-point dlcI, dlcI 201(0xC9,0x3090), broadcast
status defined, active
```

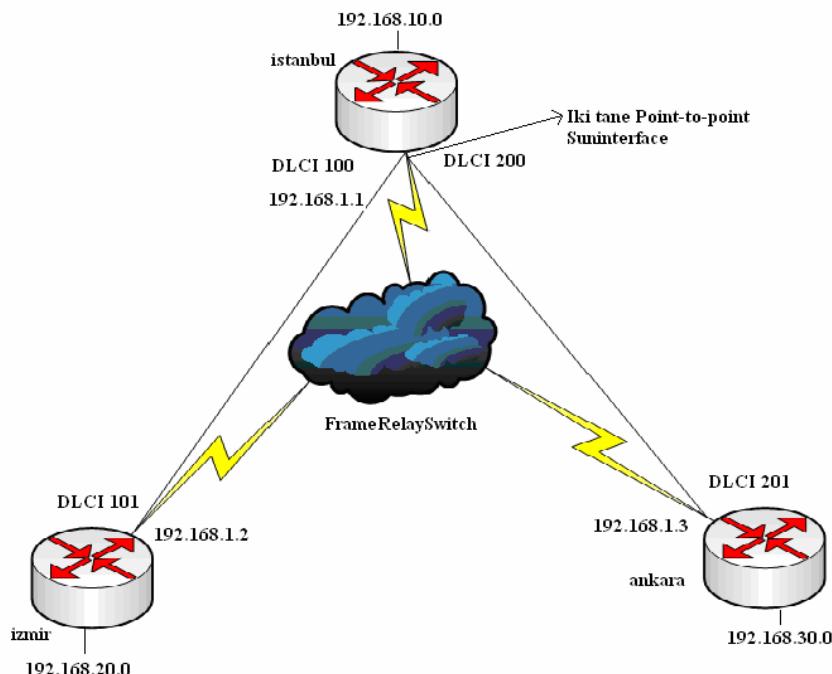
Ankara#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	1	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 201, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.1

input pkts 59	output pkts 78	in bytes 6484
out bytes 9496	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 63	out bcast bytes 7936	
pvc create time 00:19:30, last time pvc status changed 00:19:30		

Frame Relay Hub and Spoke Point-To-Point Konfigurasyonu**Router Konfigurasyon Dosyaları****Router İstanbul**

```
version 12.0
!
hostname İstanbul
!
interface Serial0
  no ip address
  encapsulation frame-relay
```

```

no frame-relay inverse-arp
!
interface Serial0.1 point-to-point
  ip address 192.168.1.1 255.255.255.0
  frame-relay interface-dlci 100
!
interface Serial0.2 point-to-point
  ip address 192.168.2.1 255.255.255.0
  frame-relay interface-dlci 200
!
interface FastEthernet0
  ip address 192.168.10.1 255.255.255.0
  no keepalive
!
router rip
  network 192.168.0.0
!
```

Router Izmir

```

version 12.0
!
hostname Izmir
!
interface Serial0
  ip address 192.168.1.2 255.255.255.0
  encapsulation frame-relay
  frame-relay map ip 192.168.1.1 101 broadcast
  no frame-relay inverse-arp
  frame-relay lmi-type ansi
!
interface FastEthernet0
  ip address 192.168.20.1 255.255.255.0
  no keepalive
!
router rip
  network 192.168.0.0
```

Router Ankara

```

version 12.0
!
hostname Ankara
!
interface Serial0
  ip address 192.168.2.2 255.255.255.0
  encapsulation frame-relay
  frame-relay map ip 192.168.2.1 201 broadcast
!
interface FastEthernet0
  ip address 192.168.30.1 255.255.255.0
  no ip directed-broadcast
  no keepalive
!
router rip
  network 192.168.0.0
!
```

Frame Relay Switch

```

version 12.0
!
hostname FrameSwitch
!
frame-relay switching
!
interface Serial0
no ip address
encapsulation frame-relay
clockrate 64000
frame-relay lmi-type ansi
frame-relay intf-type dce
frame-relay route 100 interface Serial1 101
frame-relay route 200 interface Serial2 201
!
interface Serial1
no ip address
encapsulation frame-relay
clockrate 64000
frame-relay lmi-type ansi
frame-relay intf-type dce
frame-relay route 101 interface Serial0 100
!
```

FrameSwitch#show frame-relay route

Input Intf	Input Dlci	Output Intf	Output Dlci	Status
Serial0	100	Serial1	101	active
Serial0	200	Serial2	201	active
Serial1	101	Serial0	100	active
Serial2	201	Serial0	200	active

Frame Relay Map ve PVC

Istanbul#show frame-relay map

```

Serial0.1 (up): point-to-point dlci, dlci 100(0x64,0x1840), broadcast
    status defined, active
Serial0.2 (up): point-to-point dlci, dlci 200(0xC8,0x3080), broadcast
    status defined, active
```

Istanbul#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	2	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.1

input pkts 123	output pkts 140	in bytes 23474
out bytes 25102	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0

in DE pkts 0	out DE pkts 0
out bcast pkts 120	out bcast bytes 23022
pvc create time 00:26:26, last time pvc status changed 00:24:46	

DLCI = 200, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.2

input pkts 89	output pkts 135	in bytes 14992
out bytes 25487	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 121	out bcast bytes 23536	
pvc create time 00:26:28, last time pvc status changed 00:24:08		

Izmir#show frame-relay map

Serial0 (up): ip 192.168.1.1 dlci 101(0x65,0x1850), static,
broadcast,
CISCO, status defined, active

Izmir#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	1	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 101, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

input pkts 155	output pkts 129	in bytes 26714
out bytes 22108	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 107	out bcast bytes 19820	
pvc create time 00:33:33, last time pvc status changed 00:31:23		

Ankara#show frame-relay map

Serial0 (up): ip 192.168.2.1 dlci 201(0xC9,0x3090), static,
broadcast,
CISCO, status defined, active

Ankara#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	1	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 201, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

input pkts 172	output pkts 108	in bytes 30389
out bytes 14884	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 87	out bcast bytes 12728	
pvc create time 00:37:06, last time pvc status changed 00:35:16		

ISDN

ISDN(Integrated Services Digital Network) var olan telefon ağı üzerinden sayısal hizmet vermek için geliştirilen bir teknolojidir. ISDN hat üzerinden ses, görüntü ve veri eş zamanlı olarak iletilebilir.

POTS un (Plain Old Telephone Service) aksine ISDN end-to-end dijitaldir. Dolayısıyla ISDN ile birlikte PCM'e (Pulse Code Modulation) ihtiyaç yoktur.

ISDN'in Avantajları:

ISDN dial-ip baglantılardan daha geniş bant genişliği sağlar.

Dial-up modemlerden daha hızlı çevrim sağlar.

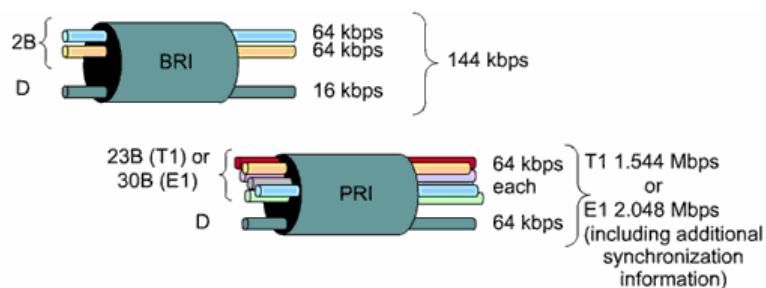
PPP encapsulation ile birlikte kullanılabilir.

ISDN'in Dezavantajları:

ISDN DSL veya kabloya göre daha yavaş ve daha pahalıdır.

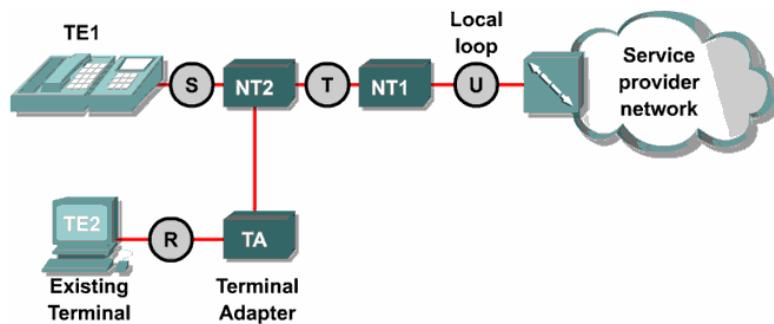
ISDN Kanalları

ISDN iki tür hat içerir. **BRI (Basic Rate Interface)** ve **PRI (Primary Rate Interface)**. Hem BRI da hem de PRI da B kanalları ve D kanalları mevcuttur. B kanalları veri taşımak için kullanılır. D kanalları kontrol ve işaretleme bilgisi taşıır. BRI hatlarında 2 adet 64 Kbps 'lik B kanalı ve bir adet 16 Kbps' lik D kanalı mevcuttur. T1 çerçevelemesini temel alan PRI 'lar 23 B+D ve E1 çerçevelemesini temel alan PRI 'lar 30 B+D olarak ifade edilir. 23 B+D 'ler Amerika'da ve 30 B+D 'ler ise Avrupa'da kullanılmaktadır.



Arayüz Türü	B Kanalları	D Kanalları	Açıklayıcı Terim
BRI	2	1	2B+D
PRI (T1)	23	1	23B+D
PRI (E1)	30	1	30B+D

ISDN Layer 1



TE1 : Bu sınıfındaki cihazlar direkt olarak ISDN ağına bağlanabilir.

TE2 : Bu sınıfındaki cihazlar ISDN standartlarını anlamazlar. ISDN ağına bağlanabilmeleri için bir terminal adaptör (TA) 'e ihtiyaç duyurlar.

NT1 : Fiziksel katman özelliklerini tanımlar. Cihazları ISDN ağına bağlar.

NT2 : Servis sağlayıcı cihazlardır.

TA : T2 kablolamasını T1 kablolamasına dönüştürür.



(BRI kart)



(Terminal Adapter)

BRI konfigüre ederken her kanal için verilen SPID (Service Profile Identifier) numarasına ihtiyaç vardır. SPID 'ler kullandığımız telefon numaralarına benzer. Internet Servis Sağlayıcısından bize verilen SPID numaralarını "**isdn spid1**" ve "**isdn spid2**" komutlarını kullanarak girebiliriz. Ayrıca konfigüre ederken servis sağlayıcının kullandığı switch türünü de router üzerinde belirtmemiz gerekiyor. Kullandığımız router 'ın ne tür switchlere destek verdiğine göre bilmek için "**isdn switch-type ?**" komutu kullanılabilir.

(Turkiyede basic-net3 kullanılmaktadır.)

```
Router(config-if)#isdn spid1 spid - numarasi
Router(config-if)#isdn spid2 spid - numarasi
```

PPP ve CHAP authentication kullanımı;

```
Gateway(config)#username ISP password class
Gateway(config)#isdn switch-type basic-dms100

Gateway(config)#interface bri 0
Gateway(config-if)#ip add 10.0.0.3 255.0.0.0
Gateway(config-if)#encapsulation ppp
Gateway(config-if)#ppp authen chap
Gateway(config-if)#isdn spid1 08443 213
Gateway(config-if)#isdn spid2 08132 344
```

```
R1#show isdn status
Global ISDN Switchtype = basic-ni
ISDN BRI0/0 interface
dsl 0, interface ISDN Switchtype = basic-ni
    Layer 1 Status:
        ACTIVE
            Layer 2 Status:
                TEI = 64, Ces = 1, SAPI = 0, State =
                MULTIPLE_FRAME_ESTABLISHED
                TEI = 65, Ces = 2, SAPI = 0, State =
                MULTIPLE_FRAME_ESTABLISHED
                    Spid Status:
                        TEI 64, ces = 1, state = 5(init)
                            spid1 configured, no LDN, spid1 sent, spid1 valid
                            Endpoint ID Info: epsf = 0, usid = 70, tid = 1
                        TEI 65, ces = 2, state = 5(init)
                            spid2 configured, no LDN, spid2 sent, spid2 valid
```

```
R2#show interface bri0/0.1
BRI0:1 is up, line protocol is up
    Hardware is BRI
        MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely
        255/255, load 1/255
            Encapsulation PPP, loopback not set, keepalive set
            (10 sec)
            LCP Open
            Open: IPCP, CDPCP
                Last input 00:00:01, output 00:00:01, output hang
                never
                Last clearing of "show interface" counters never
                Input queue: 0/75/0 (size/max/drops); Total output
                drops: 0
```

PRI konfigüre ederken öncelikle PRI kartın takılı olduğu port a girilir. Daha sonra "framing" komutuyla servis sağlayıcı tarafından belirlenen frame türü belirlenir. Daha sonra sabit timeslots numaraları konfigüre edilir. T1 için timeslots aralığı 1-24 ve E1 için timeslots aralığı 1-31 dir. Kullanılan "linecode" komutuyla fiziksel katmandaki sinyal modeli seçilir. Bu sinyal modellerinden HDB3 Amerika da B8ZS ise Kuzey Amerika da kullanılmaktadır. Son olarak router 'ın üzerindeki T1/E1 seri interface 'ine girilir. E1 için 1 ile 31 arası ve T1 için 1 ile 24 arasıdır. Bu frame relay de kullanılan subinterface gibi algılanmamalıdır. Çünkü frame relay da interface serial 0/0.16 şeklinde bit tanımlama bulunmaktaydı fakat PRI da ise interface serial 0/0:23 şeklinde bir tanımlama yapılacaktır. Bu tanımlamayla bir kanal açılacaktır.

PRI T1 konfigürasyonu ;

```
Router(config)#controller t1 1/0
Router(config-controller)#framing esf
Router(config-controller)#linecode b8zs
Router(config-controller)#pri-group timeslots 1-24
Router(config-controller)#interface serial3/0:23
Router(config-if)#isdn switch-type primary-5ess
Router(config-if)#no cdp enable
```

PRI E1 konfigürasyonu

```
Router(config)#controller e1 1/0
Router(config-controller)#framing crc4
Router(config-controller)#linecode hdb3
Router(config-controller)#pri-group timeslots 1-31
Router(config-controller)#interface serial3/0:15
Router(config-if)#isdn switch-type primary-net5
Router(config-if)#no cdp enable
```

DDR

DDR (Dial-on-Demand Router) iki veya daha fazla Cisco router'ın ISDN dial up bağlantı yapmasını sağlar. Genellikle PSTN veya ISDN kullanılarak gerçekleşen periyodik network bağlantılarında kullanılır. Böylece gerek duyulunca bağlantı gerçekleştir ve ödenecek ücret azalacaktır.

DDR bağlantı konfigürasyonu yapılırken öncelikle bağlantı kurulacak interface içerisinde ip adresi tanımlaması yapılır. Daha sonra static bir yönlendirme yapılır. Son olarak "**dialer-list**" komutu kullanılarak oluşturulan liste hangi tür paketlerin bu bağlantıyı aktif yapacağı belirlenir. Ve network bağlantısında kullanılacak arama bilgileri konfigüre edilir. Asagidaki calisma incelendiginde DDR in calisma mantigi daha iyi anlasilacaktir.

```

Router(config)# username ISP pass class
Router(config)# isdn switch-type basic-5ess
3 { Router(config)# dialer-list 1 protocol ip list 101
    Router(config)# access-list 101 deny tcp any any eq telnet
    Router(config)# access-list 101 deny tcp any any eq ftp
    Router(config)# access-list 101 permit ip any any

1 → Router(config)# interface bri 0
    Router(config-if)# ip add 10.0.0.3 255.0.0.0      Hedef network
    Router(config-if)# encapsulation ppp
    Router(config-if)# ppp authen chap
2 → Router(config-if)# dialer-group 1
4,5 → Router(config-if)# dialer map ip 10.0.0.4 name ISP 5554000

```

3. Routing table ilgili trafigin bri 0; üzerinden olacagini gösterdiği için bu interface in konfigürasyonu kontrol edilir.
4. Router bu interface deki "dialer-group 1" komutundan aynı id numarasına sahip dialer-list den bu trafige izin verilip verilmeyeceginin araştırılması gerektiğini anlar.
5. Bu trafige izin verilip verilmeyeceği ilgili "dialer-list 1 protocol ip list 101" de belirtilen 101 numaralı access list ile kararlaştırılır.
6. Trafige izin verilecek ise next hopu bulmak için dialer map' e basvurulur.
7. Dialer map kullanımdaysa data gönderilir, kullanımda degilse call setup işlemi başlar.

Burada artık bir kez bağlantı kurulduktan sonra access list ile belirlenen kriterlere uymayan paketler de gönderilecektir. Fakat sadece bu kriterlere uyan paketler konfigürasyona eklenebilecek idle-time suresini resetleyecektir.

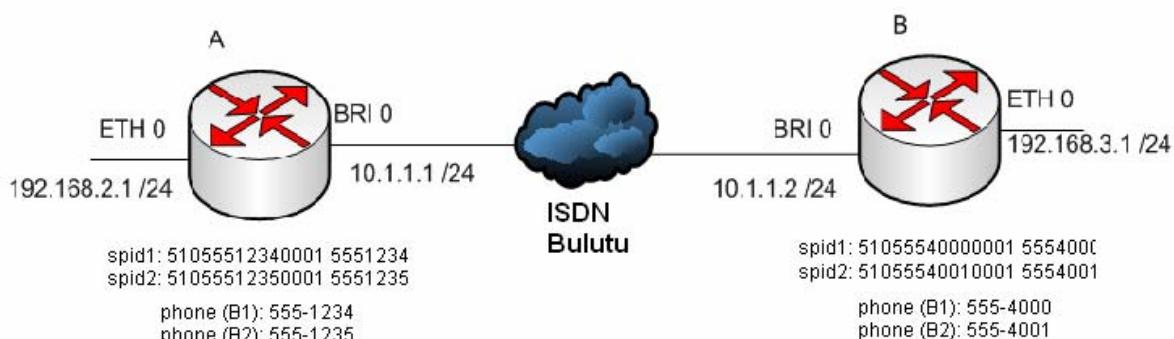
```

hostname Home
!
isdn switch-type basic-5ess
!
username Central password cisco
interface BRI0
  ip address 10.1.0.1 255.255.255.0
  encapsulation ppp
  dialer idle-timeout 180
  dialer map ip 10.1.0.2 name Central 5552000
  dialer-group 1
  no fair-queue
  ppp authentication chap
!
```

Dialer Load-Threshold Komutu

"dialer load-threshold" komutu BRI interface 'inin ikinci B kanalının ne zaman aktif olacağını söyler. Parametre olarak 1 ile 255 arası bir değer alır. Eğer 255 kullanılırsa birinci B kanalı %100 kullanıldığında ikinci B kanalı aktif edilir. İkinci bir parametre olarak "in" gelen trafiği , "out" giden trafiği, "either" her ikisinin hesaplanacağını router'a bildirir. **"dialer idle-timeout"** komutu en son iletilen paketin ardından ne kadar süre sonra bağlantının kopacağını belirtmektedir.

ISDN Konfigurasyon Ornegi



ISDN konfigurasyon ornegi içerisinde SPID numaraları ve telefon numaraları kullanılmıştır.

B kanallarının her ikisi birlikte kullanılacağı için her iki kanal için de telefon numaraları ve SPID numaraları verilmistir.

Her iki Router ' da ISDN networke BRI 0 portlarından bağlanmıştır.

Konfigurasyon içerisinde ppp authentication chap kullanılmıştır.

Yönlendirme için IGRP konfigurasyonu kullanılmıştır ve IGRP için AS numarası 100 olarak seçilmiştir.

RouterA

```
version 12.0
hostname RouterA
!
enable password cisco
!
username RouterB password 0 cisco
!
ip host RouterB 192.168.3.1
!
isdn switch-type basic-ni
!
interface FastEthernet0/0
 ip address 192.168.2.1 255.255.255.0
 no ip directed-broadcast
!
interface BRI0/0
 ip address 10.1.1.1 255.255.255.0
 encapsulation ppp
 dialer map ip 10.1.1.2 name RouterB 5554000
 dialer-group 1
 isdn switch-type basic-ni
 isdn spid1 51055512340001 5551234
 isdn spid2 51055512350001 5551235
 ppp authentication chap
!
router igrp 100
 passive-interface BRI0/0
 network 10.0.0.0
 network 192.168.2.0
!
ip route 0.0.0.0 0.0.0.0 10.1.1.2
!
dialer-list 1 protocol ip permit
!
end
```

RouterB

```

version 12.0
hostname RouterB
!
enable password cisco
!
username RouterA password 0 cisco
!
isdn switch-type basic-ni
!
interface BRI0
ip address 10.1.1.2 255.255.255.0
encapsulation ppp
dialer map ip 10.1.1.1 name RouterA 5551234
dialer-group 1
isdn switch-type basic-ni
isdn spid1 51055540000001 5554000
isdn spid2 51055540010001 5554001
ppp authentication chap
!
interface FastEthernet0
ip address 192.168.3.1 255.255.255.0
no ip directed-broadcast
!
router igrp 100
passive-interface BRI0
network 10.0.0.0
network 192.168.3.0
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
!
dialer-list 1 protocol ip permit
!
End

```

RouterA#**show inter bri 0**

BRI0 is up, line protocol is **up (spoofing)**
Hardware is PQUICC BRI with U interface
Internet address is 10.1.1.1/24
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, loopback not set
Last input 00:00:08, output never, output hang never

show isdn status

```

RouterA#show isdn status
Global ISDN Switchtype = basic-ni
ISDN BRI0 interface
    dsl 0, interface ISDN Switchtype = basic-ni
Layer 1 Status:
ACTIVE
Layer 2 Status:
    TEI = 64, Ces = 1, SAPI = 0, State = MULTIPLE_FRAME_ESTABLISHED
    TEI = 65, Ces = 2, SAPI = 0, State = MULTIPLE_FRAME_ESTABLISHED
Spid Status:
    TEI 64, ces = 1, state = 5(init)
        spid1 configured, spid1 sent, spid1 valid
        Endpoint ID Info: epsf = 0, usid = 70, tid = 1
    TEI 65, ces = 2, state = 5(init)
        spid2 configured, spid2 sent, spid2 valid
        Endpoint ID Info: epsf = 0, usid = 70, tid = 2
Layer 3 Status:
1 Active Layer 3 Call(s)
Activated dsl 0 CCBs = 1
    CCB:callid=8031, sapi=0, ces=1, B-chan=1, calltype=DATA
The Free Channel Mask: 0x80000002
Total Allocated ISDN CCBs = 1

```

RouterA#show dialer

BRI0 - dialer type = ISDN

Dial String	Successes	Failures	Last DNIS	Last status
5554000	1	8	00:02:49	successful

0 incoming call(s) have been screened.

0 incoming call(s) rejected for callback.

BRI0:1 - dialer type = ISDN

Idle timer (120 secs), Fast idle timer (20 secs)

Wait for carrier (30 secs), Re-enable (15 secs)

Dialer state is data link layer up

Dial reason: ip (s=10.1.1.1, d=192.168.3.1)

Time until disconnect 70 secs

Connected to 5554000 (denver)

BRI0:2 - dialer type = ISDN

Idle timer (120 secs), Fast idle timer (20 secs)

Wait for carrier (30 secs), Re-enable (15 secs)

Dialer state is idle

show isdn activeRouterA#**show isdn active****ISDN ACTIVE CALLS**

Call Type	Calling Number	Called Number	Remote Name	Seconds Used	Seconds Left	Seconds Idle	Charges Units/Currency
Out	5554000		RouterB	177	62	57	0

RouterA#**debug isdn events**

ISDN events debugging is on

RouterA#**ping denver**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:

00:14:04: ISDN BR0: Outgoing call id = 0x8032, dsl 0
 00:14:04: ISDN BR0: Event: Call to 5554000 at 64 Kb/s
 00:14:04: ISDN BR0: process_bri_call(): call id 0x8032, called_number 5554000, speed 64, call type DATA
 00:14:21474836479: CC_CHAN_GetIdleChanbri: dsl 0
 00:14:17179869184: Found idle channel B1
 00:14:19335326197: ISDN BR0: received HOST_PROCEEDING call_id 0x8032
 00:14:17179869184: ISDN BR0: received HOST_CONNECT call_id 0x8032
 00:14:17179869232: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up
 00:14:17179869248: %ISDN-6-CONNECT: Interface BRI0:1 is now connected to 5554000

00:14:19337989260: ISDN BR0: Event: Connected to 5554000 on B1 at 64 Kb/s

!!!!

Success rate is 80 percent (4/5), round-trip min/avg/max = 32/32/32 ms

RouterA#

00:14:05: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1, changed state to up
 00:14:10: %ISDN-6-CONNECT: Interface BRI0:1 is now connected to 5554000 RouterB

Kaynaklar**www.cisco.com****www.ietf.org****CNAP Official Curriculum**