



Türkçe CCNA EĞİTİM Notları

# INTERNETWORKING TEMELLERI

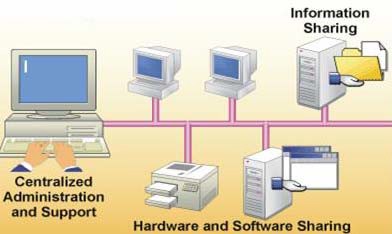
**Network Nedir, Ne İşe Yarar?**

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.



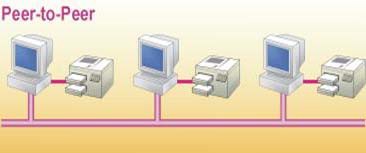
Bilgisayarlar networklerde çeşitli görevler üstlenebilirler. Genel olarak bir bilgisayar bir networkte client (istemci) ya da server (sunucu) rollerinden birini üstlenir.

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

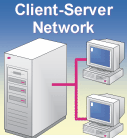
Network ortamında kaynak ya da veri isteyen bilgisayarlara ise Client adı verilir. Client sadece kendisinden donanımsal olarak büyük olan Server lardan değil gerektiğinde diğer client’ lardan da kaynak ya da veri talebinde bulunabilir.

# 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 bilgisayarlarla bir Client – Server yapısında hareket ederler. Her bilgisayar kaynaklarını ya da sahip olduğu datayı istediği kadarıyla kullanıma açabilir ya da açmaz.



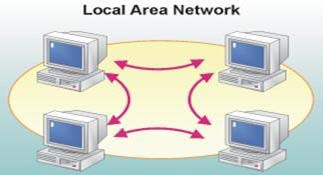
Client / Server mimarisinde is, adından da anlaşılacağı gibi hem donanımsal hem de yazılımsal olarak diğer bilgisayarlardan üstün, atanmış bir ana bilgisayar vardır.



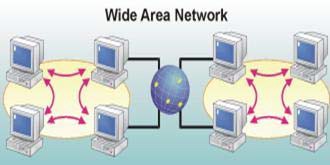
Burada Server olarak atanmış bilgisayarın yetersiz kaldığı durumlarda networke başka serverlarda dahil edilebilir. Örneğin gelişmiş bir networkte Mail Server’ın, DHCP ve DNS gibi serverların farklı bilgisayarlarda bulunması performansı olumlu yönde etkileyebileceği için önerilebilir.

# LAN, WAN ve MAN

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

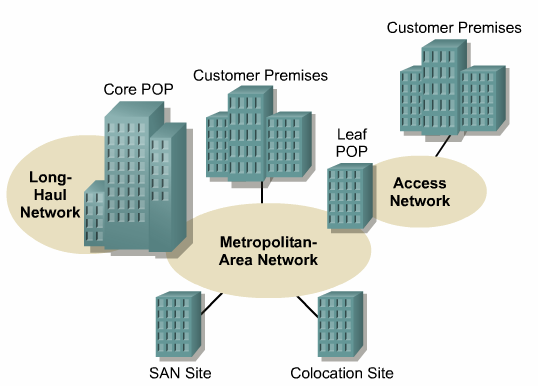


Birbirine yakın yerlerde konumlandırılmış ve kablolar ile fiziksel olarak birbirlerine bağlanmış yapıdaki networkler LAN olarak adlandırılırlar. Örneğin bir binada bulunan bütün bilgisayarların birbirlerine bağlanmasıyla oluşan yapı bir Local Area Network’ tür.



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.

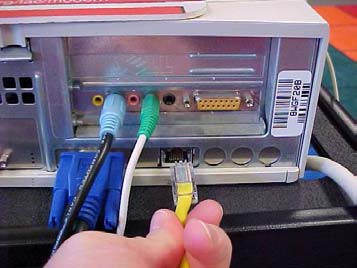


Örnek vermek gerekirse birden fazla şubesi bulunan bir bankanın kullanabileceği bir yapı diyebiliriz.

# 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 üretimden 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 il 3 okteti 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 okteti de firmaya bırakır. Bu şekilde bir standart sağlanırken aynı MAC adresine sahip Ethernet kartlarının üretilmesi de engellenmiş olur.

(RJ-45 ile sonlandirilmis UTP kablonun Ethernete takilmasi)

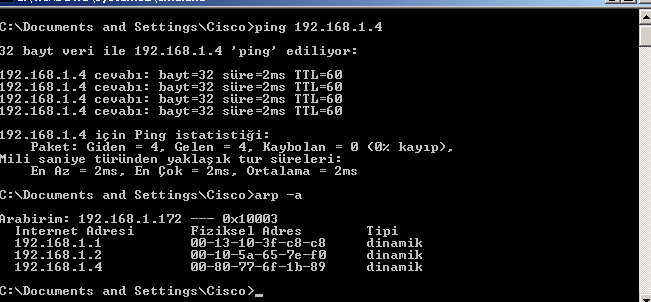
Bir bilgisayarın MAC adresi komut satırında “ipconfig /all” yazılara öğrenilebilir. ( Windows 9x ortamında ipconfig.exe yerine Winipcfg.exe kullanılır.)



Bir sistem yöneticisi kendi bilgisayarından diğer bilgisayarların MAC adreslerini de öğrenmek isteyebilir. Sözgelimi 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.

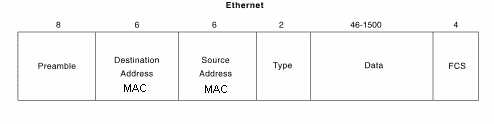
Bir bilgisayara en az bir kere ulaşmış olmak kaydıyla, komut satırında “arp –a” yazılarak o bilgisayarın MAC adresi öğrenilebilir.



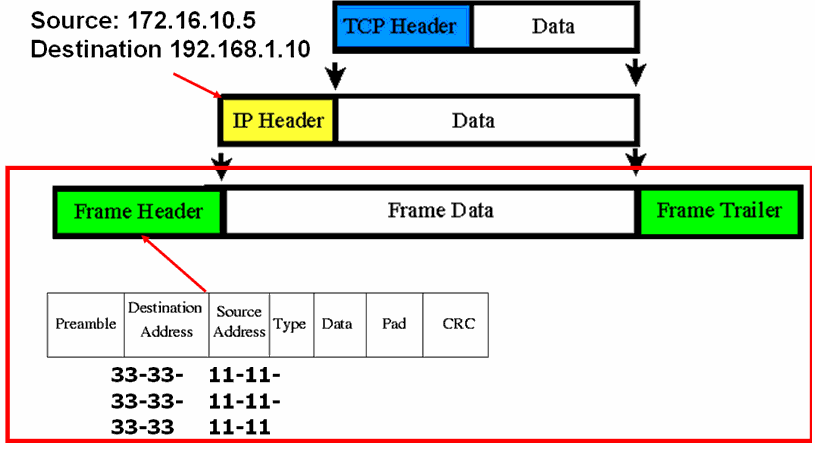
Komut satırından alınmış şekilde, önce 192.168.1.4 bilgisayarına bir kez ulaşmak için ping atılmış. Daha sonra kullanılan “arp- a” komutu ile oturum boyunca ulaşılan tüm bilgisayarların bellekteki fiziksel adres tablosuna erişilmiştir.

(ARP Protokolü ileride detaylı olarak anlatılacaktır.)

Ethernet teknolojiler IEEE 802.3 standardi ile tanilanmistir ve ethernetin datayi framl e’ ler halinde tasidigi soylenebilir. Genel olarak ethernet frame’ leri asagidaki gibidir.



Destination ve Source adresler hedef ve kaynak cihazlarin fiziksel adreslerini ifade eder. FCS degeri ise datanin saglikli iletilip iletilmediginin kontrol edilmesini saglayan bir degerdir.



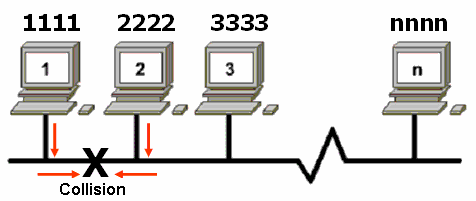
OSI Referans modeli icerisinde detayli anlasilabiliecek datanin iletimi sirasin data katmanlar ilerler. Ve her katmanda data uzerine o katmanin calisma mantigi icerisinde gereken bilgi etiketlenir. Sekilde simdilik onemli olan kismin sirasiyla TCP Header ve IP Header eklenir. LAN icerisinde data 2. katmanda, yani fiziksel adresler yardimiyla haberlesecektir. Frame Header ile bu bilgiler dataya eklenir.

IP Header ve Frame Header arasindaki en onemli fark frame’ lerin TTL (Time To Live) degerine sahip olmamasidir. Dolayisi ile ikinci katmanda olusabiliecek bir dongu donguyu yasayan cihazlar kapatilmadigi surece devam edecektir.

# CSMA/CD

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

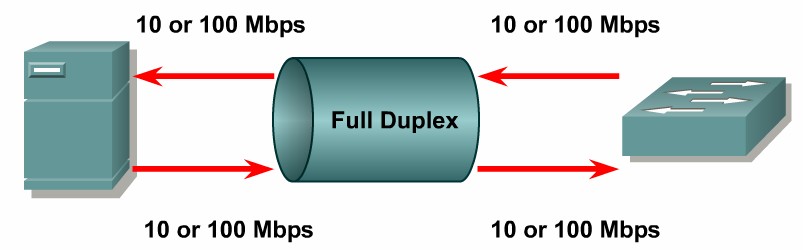
İki bilgisayarın birden kabloyu kullanmaya çalışması collision olarak adlandırılır. Her ikisinin de trafiği kaybolur. Ve hattın boş olduğu anı yakalamak için yeniden beklemeye ve hat dinlenmeye başlanır.



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

# Full Duplex

Full Duplex ethernet ayni anda hem data iletimini hem de data alinmasini saglar.



IEEE 802.3x ile tanimlanmis Full Duplex calisma icerisinde, her iki cihaz da Full Duplex modda oldugu zaman sonuc alinabilecektir. Ornegin, host to switch, switch to switch yada switch to router baglantilarinda her iki tarafta Full Duplex modda calisabiliyor olmalidir.

Half Duplex calisma icerisinde collisionlardan dolayi bant genisligini yuzde 50 yada altmisi kullanilabilirken Full Duplex calisma da bant genisliginin tamami kullanilabilir. Sozgelimi 10 Mbps olan bir hattin Half Duplex ile 5-6 Mbps’ i kullanilabilir, oysa Full Duplex calisma da ideal olarak ayni bant genisliginde 20 Mbps’e ulasildigi varsayilir.

Full Duplex icerisinde CSMA/ CD ‘ den bahsedilemez. Cunkü ayni anda hem iletim hem alim olabilecegi icin Carrier Sense olmayacaktir.

# KABLO Standartları

Bilgisayarlar ethernet kartlarına takılan kablo aracılığıyla birbirine bağlanırlar. Networkün yapısına göre farklı özelliklerde kullanabilecek bir çok çeşit kablo standardı vardır. Ana başlıkları şöyle sıralayabiliriz;

-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 kablolarıdır. 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 inç genişliğindedir. Yaygın olarak kullanılır. Verileri sağlıklı olarak 185 metre uzağa iletebilirler. Thinnet koaksiyel kablolar RG-58 standardı olarak değişik biçimde üretilmektedir.

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

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

# Mesafe Koaksiyel kablo

185 m Thinnet

500 m Thicknet



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

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



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



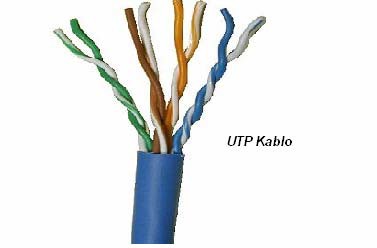
BNC kablo konnektörü kablonun ucunda yer alır. T konnektör ise koaksiyel kabloyu networkadaptörüne bağlamak için kullanılır. Barrel konnektör ise iki koaksiyel kablonun birbirine bağlanmasını sağlar. Sonlandırıcılar ise kablonun sonunda yer alırlar.

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)



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 ileten yüksek teknoloji iletim ortamlarıdır. Fiber-optik kablolar hızlı ve yüksek kapasiteli veri iletimi için uygundur. Özellikler 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 işareti için 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 kırılmalara karşı koruyan ve yine bir cam tabaka olan Cam Örtü ve tüm cam kısmı koruyan Koruyucu Kılıf olarak adlandırılabilecek 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.



# Ethernet Kablolama Sistemi

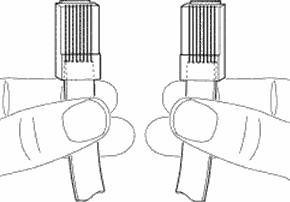
Ethernet network'lerinde dört çeşit kablolama sistemi kullanılır:

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

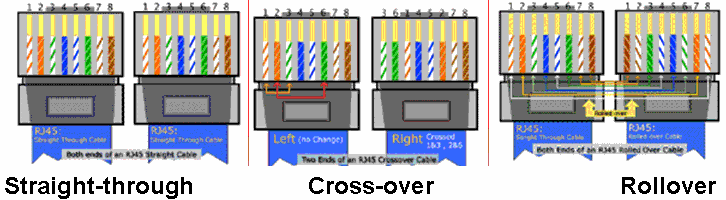
# UTP Kablo Yapimi

UTP kablo yapmak hem kolay hem de eglenceli bir istir. Bunun icin kablo ve RJ-45 ile bir de Jack pensesine ihtiyacimiz var.

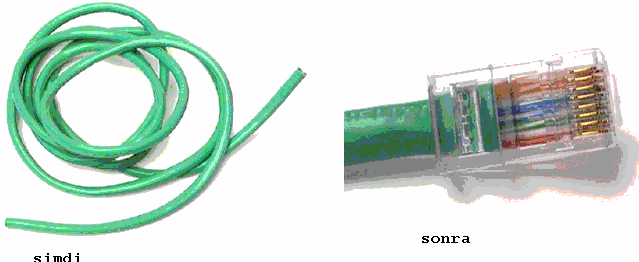
Kablo yapilirken dikkat edilecek unsurlardan biri de kablonun Duz, Cross yada Roll Over mi olacagi ve bu cesitlerin renk siralari.



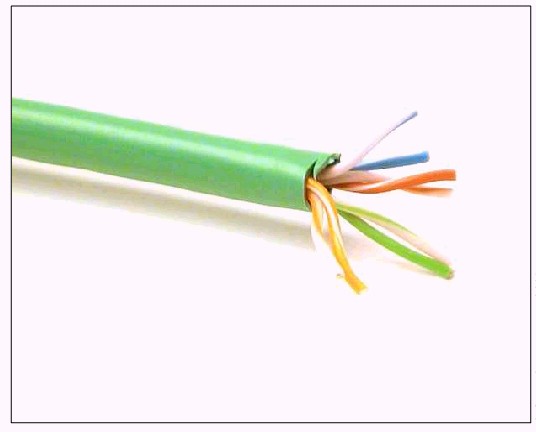
Kablolar sekilde ki gibi tutularak renk siralarina bakilip cinsi hakkinda bilgi edinilebilir.

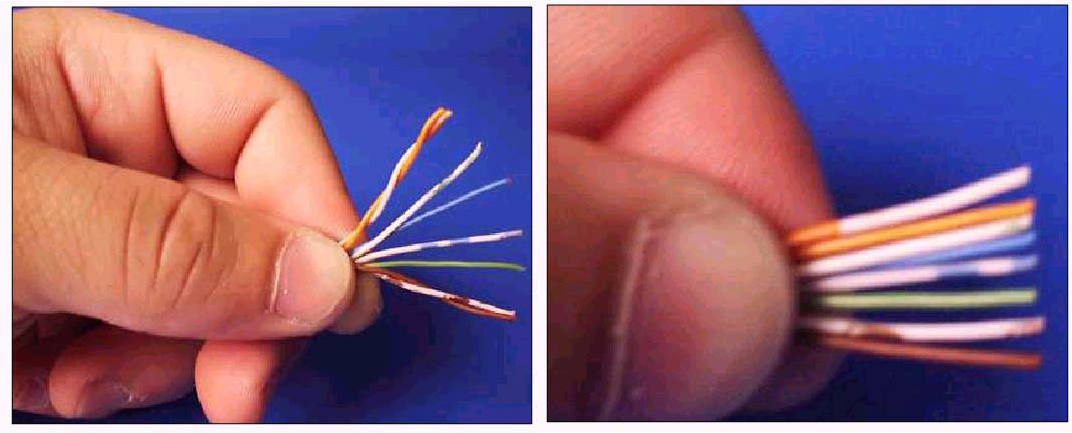


Sekilde üç çeşit kablonun renk siralari gorulmektedir. Simdi adim adim kablomuzu yapacagiz.



Kablomuzun ucundan resimdeki gibi bir kismini soyiyoruz. Bunun icin Jack Pensesi de kullanilabilir.

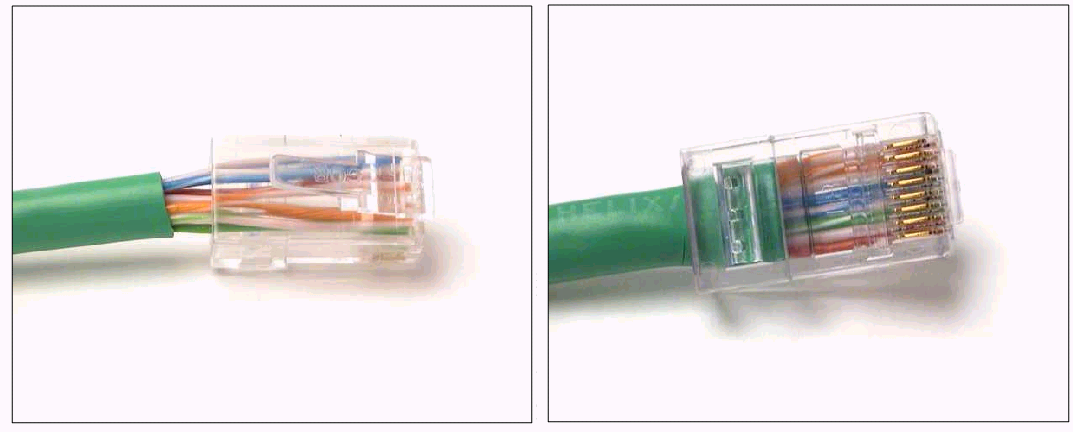




Yapacagimiz kablonun cinsine gore renk siralarina uygun sekilde kablo uclarini siralayip duzenliyoruz.



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.



Ilk sekildeki gibi kablo uclarini RJ-45 icine yerlestirdikten sonra Jack pensesi ile sikistirip kablonun RJ-45 e tam olarak oturmasini sagliyoruz.



# Network Cihazları

Bir networkü sade bilgisayar ekleyerek genişletemeyiz. Bu bize kablolamanın zorlaşması, sinyal zayıflaması gibi sebeplerden sorun yaratır. Bu sebeple bir networkü genişletmek, güvenliğini sağlamak ve aynı zaman da hiyerarşi 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.



# Switch

Kendisine bağlı cihazlara adından da anlaşılacağı 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 geçebilirler. Bu sayede Hub’a göre daha yüksek bir performans sağlanacaktır. 8 -12 – 16 – 24 – 36 – 48 port lu olarak ya da şaseli üretilebilirler. Şaseli switchlerde boş yuvalar vardır ve gerektiğinde port eklenebilmektedir.



# Repeater

Repater 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 yönüyle repater'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.



# 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 trafiğini kontrol eden aygıtlardır.

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



# 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 gerekir.

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



# Gateway

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

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

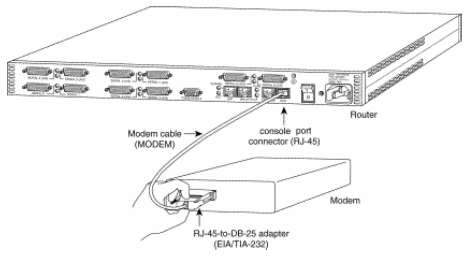


# Router

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 Systtem) dolayısıyla programlanabilirler ve gerekli konfigürasyonlar yapıldığında bir uzak networke erişmek için mevcut birden fazla yol arasında kullanabilecekleri en iyi yolun seçimini yapabilirler (Best Path Determination).

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





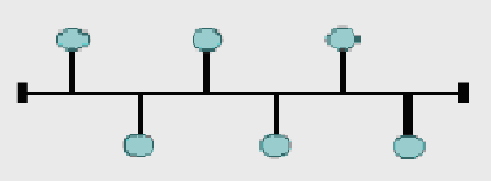
İlerleyen bölümlerde Router’ ların konfigürasyonları detaylı olarak anlatılacaktır. Routerlar ve Routing işlemi CCNA sıvalarının ana omurgasıdır.

# NETWORK TOPOLOJİLERİ

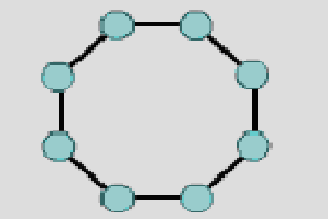
Topoloji dediğimiz de bir ağın fiziksel ya da mantıksal yapısını anlamalıyız. Networkü oluşturan cihazların fiziksel yerleri, kabloların bağlanış şekilleri, iletişimde kullanılan protokoller gibi birçok unsur network topolojilerini belirler.

# Fiziksel Topolojiler:

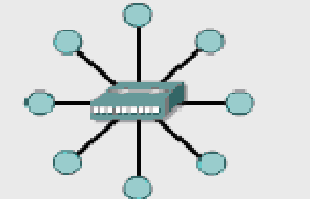
Bus Topoloji: Bütün terminallerin tek bir doğrusal bir kablo ile birbirlerine bağlanmışlardır. Burada hatta gönderilen sinyal bütün terminallere gider. Sinyal hedefe ulaşana ya da bir sonlandırıcıya gelene kadar hatta dolaşır. Çok az miktarda kablo kullanılıyor olması avantaj gibi görünse de ana kabloda 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.



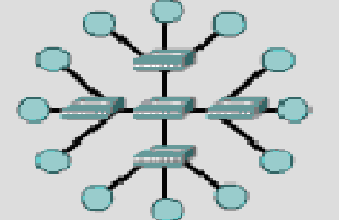
Ring Topoloji: Bu topolojide adından da anlaşılacağı gibi dairesel bir yapı söz konusudur. Hatta gönderilen sinyaller hedefe ulaşana kadar tüm terminallere uğrar. Tüm terminaller eşit haklara sahiptir.



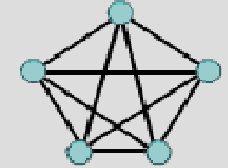
Star Topoloji: Star Topolojide her bilgisayar switch ya da hub dediğimiz network cihazlarına direk bağlıdır. Hatta gönderilen sinyal önce switch ya da hub’a gelir ve buradan hedefe gönderilir. Böyle bir yapının en büyük avantajı yeni bilgisayarlar ekleyerek büyümek çok kolaydır, yönetilmesi ve sorun 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.



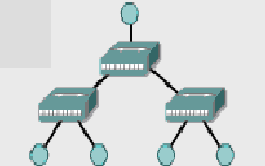
Extended Star Topoloji: Adından anlaşılacağı gibi Star Topolojinin geliştirilmesiyle ortaya çıkmıştır. Birden fazla yıldız topolojinin bir araya gelmesiyle oluşmuş bir yapıdır diyebiliriz.



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



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



# Mantıksal Topolojiler:

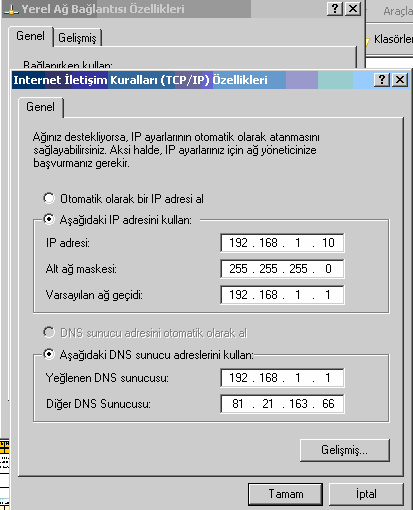
Broadcast Topoloji: Bir bilgisayar hatta gönderdiği bir sinyali diğer bütün bilgisayarların alacağı bir yayın şeklinde yapar. Yayın hedefe ulaştığı ana kadar bütün terminalleri tek tek dolaşır.

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 datası yoksa direk bir sonraki terminale aktarır. Bu şekilde çalıştığında bir repeater görevi de üstlenmiştir.

# IP Adresleme

Bilgisayarlar veya diger cihazlar networklere fiziksel olarak baglanmanin yaninda mantiksal olarak ta dahil olmalidir. Bunun icin ayni networkte ki iki cihazin ayni ip networkunde olmasi gerekir, yani Network adresleri ayni olmalidir.

Bilgisayarlara ip adresleri Static veya Dinamik olarak verilebilir. Dinamik olarak ip adresi atamasi icin en guzel ornek DHCP Server’ dir. Dynamic Host Configuration Protokolun kisaltmasi olan DHCP konfigurasyonu Router uzerinde de yapilabilmektedir. Ilerleyen bolumlerde bir routerin nasil DHCP server olarak konfigure edilebilecegi anlatilmistir.



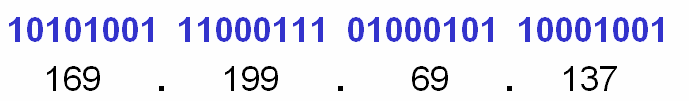
(Statik olarak ip adresi atanmasi)

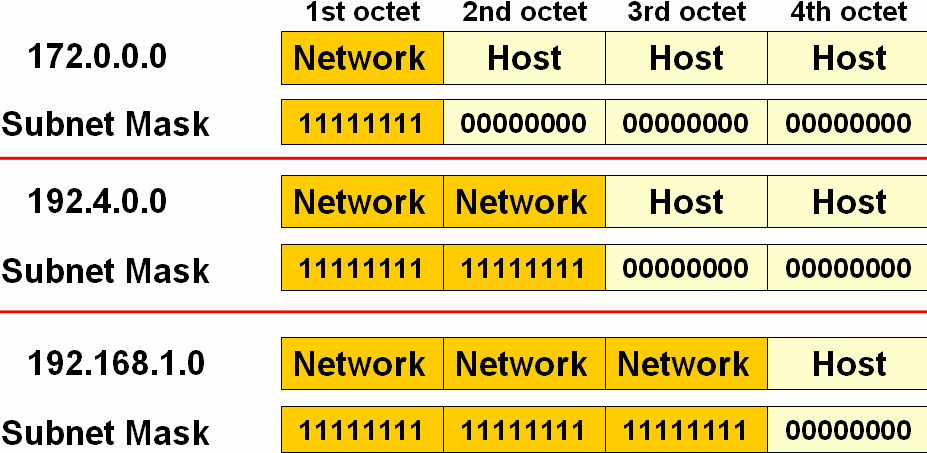
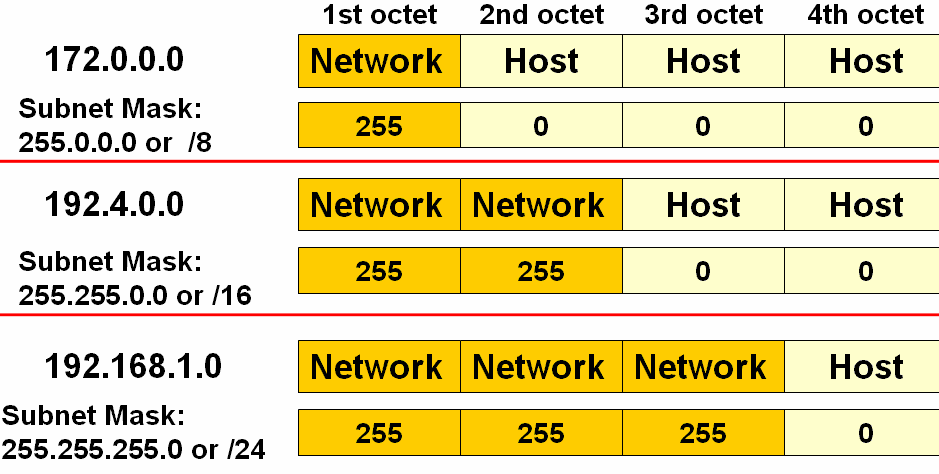
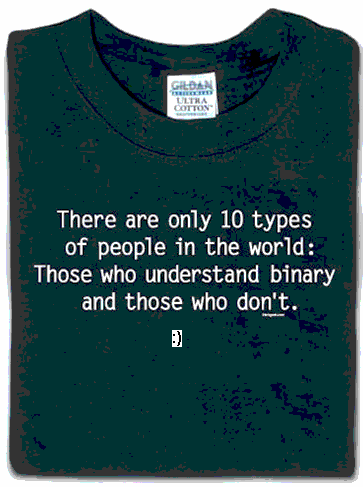
Burada Alt Ag Maskesi ifadesi dikatinizi cekmistir. Baska bir deyisle genellikle turkcemizde de kullanilan Subnet Mask.

Subnet Mask bizim icin onemli, cunki daha ileride deginecegimiz bir networku alt networklere ayirabilmemiz icin Subnet Mask ile oynamamiz gerekecek. Cunki Subnet Mask ile ip adresi binary durumda and islemine sokuldugunda network adresini verir.

Ip adresler 4 oktetten ve her oktette 8 Bitten olusur.

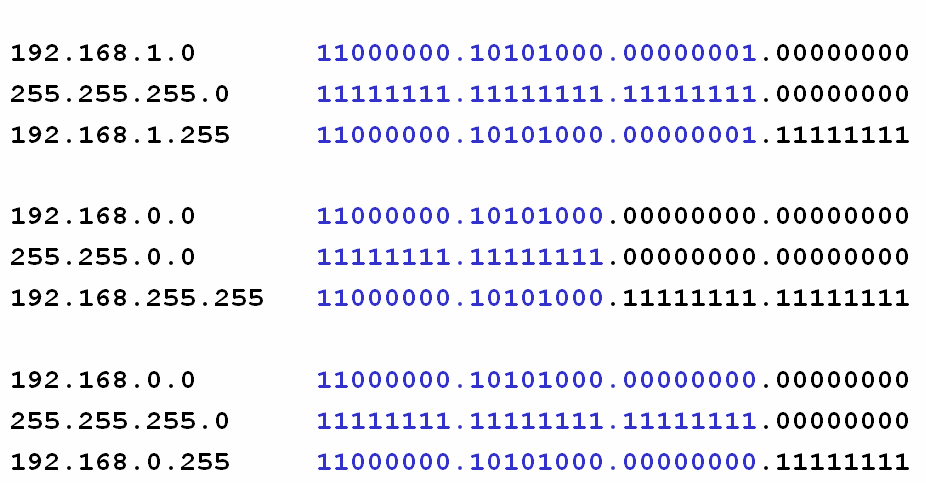




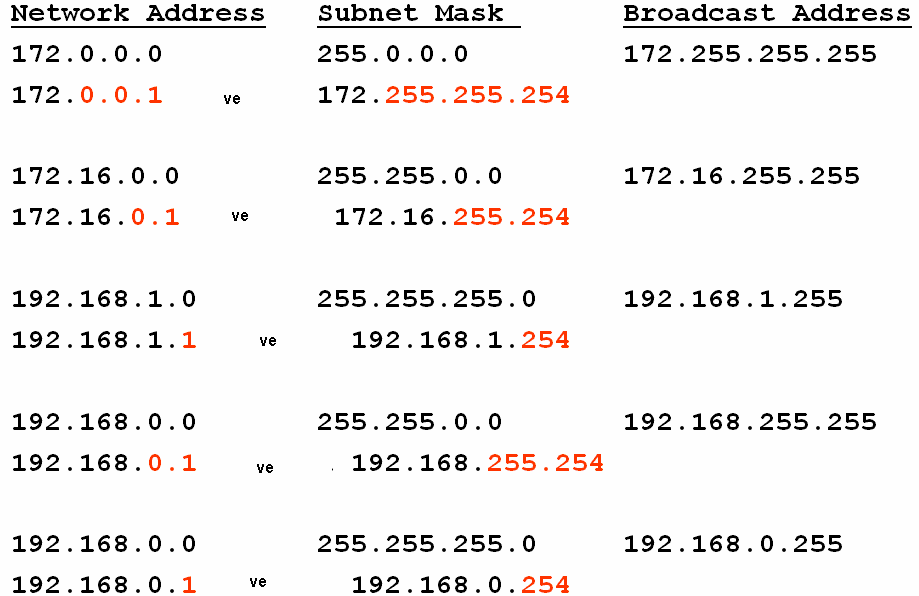


Ilk sekilde subnet masklar ikinci sekilde o subnet masklarin Binary gosterimi mevcut. Ilk resimde ki /8, /16, /24 gibi ifadeler goruyorsunuz. Bunlar Subnet Maski ifade eder, daha dogrusu Subnet Maskin Binary gosterimi icineki yoplam 1 sayisidir.

Ornek Bazi Binary gosterimler:



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



# TCP / IP Modeli

TCP/IP ile olarak DARPA (Defense Advanced ResearchProjectsAgency) 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 standartlaştığı 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 ayrılabilecek 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 (Bootsrap Protocol), DHCP (Dynamic Host Configuration Protocol), ARP (Adres Resolution Protocol) ve RARP (Reverse Address Resolution Protocol) gibi protokoller çalışmaktadır.

**Ağa Giriş Katmanı** ise OSI modelinde ki 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 büyüdükçe 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 azaltmış, 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 Katamanı 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 ağda 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 anlaşılabilecek 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 arasıdan gerçekleştirilen data transferinde datanın sağlıklı bir şekilde hedefe gönderilip gönderilmediğini 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 büyüme eğilinde 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 sağlanabilmiştir. Adresleme Dinamik ya da statik olarak yapılabilir. Sabit adresleme el ile yapılan adreslemedir. Dinamik adresleme de ise otomatik olarak ip dağıtacak örneğin DHCP 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 geçirmediği 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. Doluysa bekler ve dinlemeye devam eder
4. Data transferinde çarpışma 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

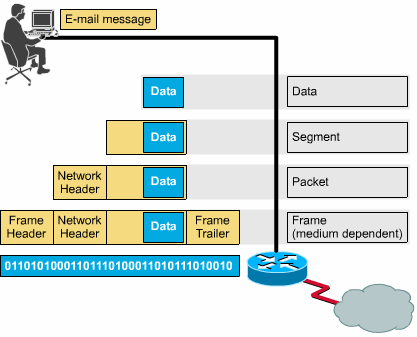
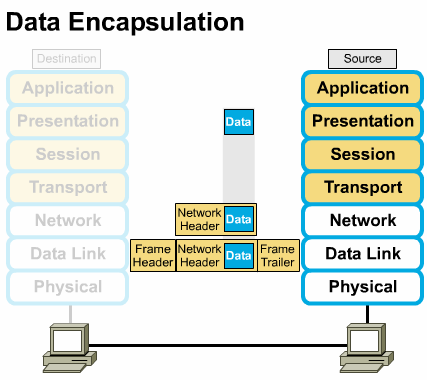
# Fiziksel Katman ( Physical Layer)

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 bölümlere 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 şekil konun daha iyi anlaşılmasını sağlayacaktır.



# 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 içerisinden 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 bazılar 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 başarıyı garantileyemez. Yani güvenilir bir aktarım servisi sağlamaz. Uygulamalar güvenli ve sıralı paket dağıtımı 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 telafisi 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 sıcaklıktan 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 standardı ile belirlenmiştir ve iki bilgisayar arasındaki iletişimde, hedef bilgisayarda, varsayılan ağ geçidinde veya routerlarda oluşan hatalar ICMP mesajı olarak kaynak bilgisayara bildirilir.

Farklı durumlara gore 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 detayli incelenecektir.

# 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 (Bootsrap Protocol)

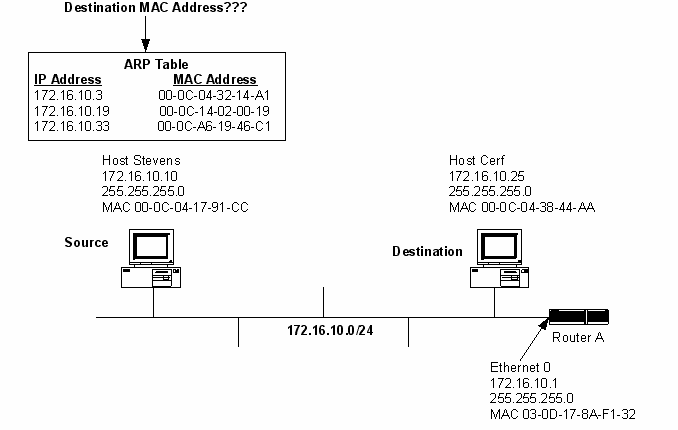
UDP tabanlıdır ve RARP protokolü gibi suncu - istemci ortamında çalışır. IP adresi isteyen bilgisayarlar bu isteklerini bir broadcast ile bildirirler. BOOTP sunucu ise ip adresini, kendi ip adresini ve varsayılan ağ geçidi adresini bir broadcast ile networke gönderiri. İstemciler MAC adreslerine balkarlar 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 dağıtımı yapılabilir. Sunucu – istemci ortamında çalışırlar ve istemcilerde ip adreslerini otomatik olarak alacaklarına dair bir konfigürasyon yapılmalıdır. DHCP ile belirlene ip adresleri, subnet masklar, dns server adresleri, varsayılan ağ geçidi gibi adresler dağıtılabilir, 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 bi rip 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.

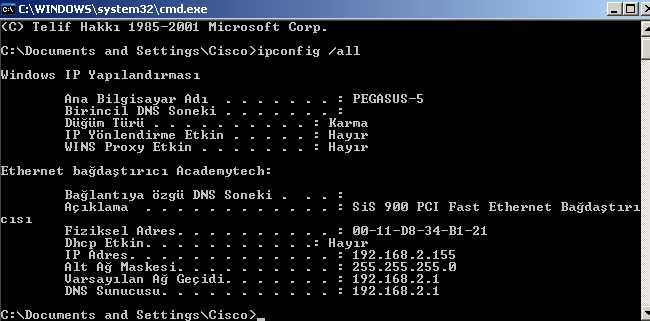
Her ip sınıfının subnet maskıda belirlenmiştir buna göre; A sınıfı için subnet mask: 255.0.0.0,

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 .alışıyor veya çalışabiliyorsa “classless” aksi durumda “classfull” denir.

Bilgisayarımızdan komut sistemini açıp “ipconfig /all” komutunu verdiğimizde kullandığımız bilgisayarın ip konfigürasyonunu görebiliriz.



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

Gerçekten de böyle olsaydı mevcut ip adreslerimiz çoktan 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 çıkarken, 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 çıkıyorsa bu ip adresi belki de sizin o an ziyaret etmek istediğiniz bir sitenin ip adresi olabilir ve siz browser’ınız a sitenin adını yazdığınız da 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 dağıtılabilir.

Ip adreslerinin dağıtılması sırasında subnet maskların standar verilmesi ciddi sorunlara sebep olacaktı. 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\*255 = 65025 ‘ dir.

Bumun 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 subnetworke bölmek istiyorsunuz;

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

Subnet Maskın 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 kullanılabilecek 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;

* + - 1. 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 | 11111111.11111111.100000000.00000000 | /17 |
| 255.255.255.128 | 11111111.11111111.111111111.01000000 | /25 |
| 255.255.255.252 | 11111111.11111111.111111111.11111100 | /30 |

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

Örneğin elimizde subnet maskı 255.255.255.128 olan 192.168.1.141 gibi bir ip var. 192.168.1.141 = 11000000.10101000.00000001.10001101

255.255.255.128 = 11111111.11111111.11111111.10000000

AND (çarpıyoruz) = 11000000.10101000.00000001.10000000

Network Adresi = 192. 168. 1. 128

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

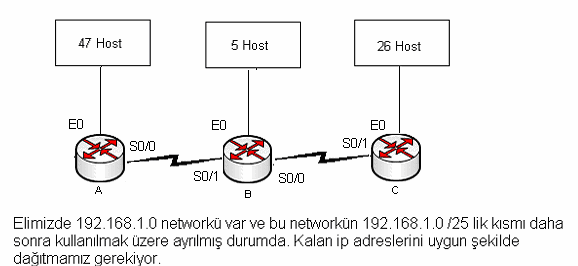
***NOT:*** Routerın Ethernet interface’ ine ip adresi atanırken önerilen networkün ilk ip adresini atamaktır.

# Classfull - Classless IP Adresleri

Classfull adreslerde subnet masklar ip adresinin hangi sınıfa ait oladuğuyla direk ilgilidir. İp 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 protokolleri Classfull protokollerdir ve subnet maskı 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’ dır. Classless adreslemeye VLSM (Variable Length Subnet Mask) veya CIDR (Classless Inter Domain Routing) denir.

# IP Subnetting Örnek Çalışma



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 kullanılmalı. Çünkü 192.168.1.0 ‘ dan 192.168.1.127 ‘ ye kadar olan ip ler daha sonra kullanılmak üzere ayrılmış durumda.

C routerı için 192.68.1.192 /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.

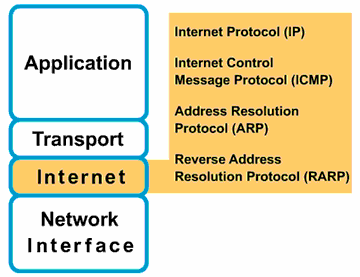
Network aralıklarımızı detaylı incelersek;

Son Oktet Yeri İçerdiği 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 yapılaması gereken network aralıklarını bulup seçenekler arasıdan ki uygun ip adreslerini ilgili yerlere atamaktır.

# ICMP (Internet Control Message Protocol)



Ucuncu katman yani Internet yada Network katmani olarak adlandirdigimiz katman IP bazindan yonledirmenin yapildigi katmandir. IP data iletimi ve yonlendirme icin belki de en iyi cozumdur.

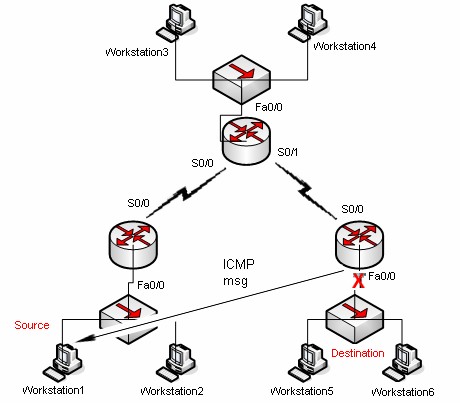
Fakat IP ile ilgili datanin iletimi sirasinda herhangio bir sebeple fail olma durumu oladugunda bu durumu kontrol edecek hata mesajlarina sahip olmamasi gibi bir sorun vardir. Sozgelimi yanlis konfigurasyonlar, donanimsal arizalar yada Routing Table’ lar ile ilgili sorunlarda IP bir hata mesaji dondurmez.

ICMP IP’ nin bu acigini gidermek uzere gelistirilmis bir protokoldur. Bahedilen durumlarda ICMP ilgili bir mesaji dondurur ve problem cozumlerde Network Muhendislerine yardimci olur.

Amcak burada IP bazinda iletimin guvenilir olmadigini, ICMP mesajlari ile bu guvenilirligin saglandigini soylmek yanlis olur. Datanin guvenilir sekilde iletilmesi bir ust katman olan Transport katmani ve bu katmanda calisan protokoller tarafindan saglanmalidir.

Genel olarak ICMP mesajlari iki ana baslik altinda incelenebilir.

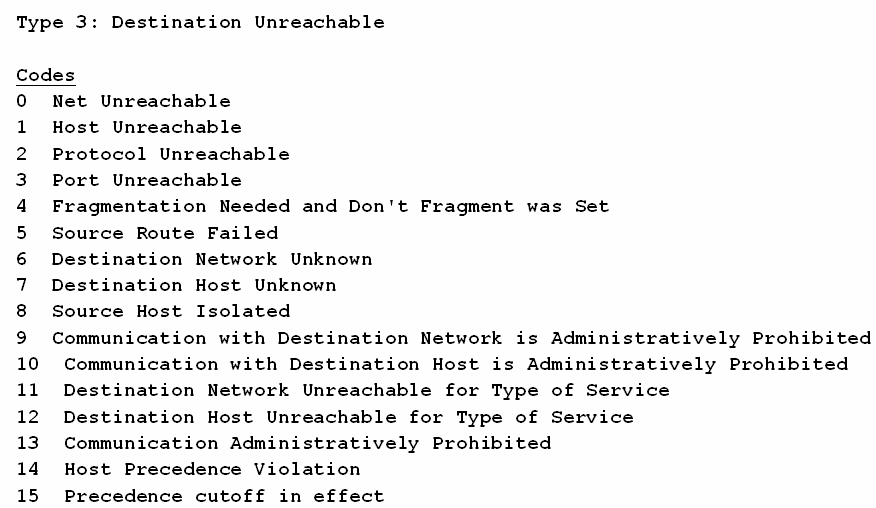
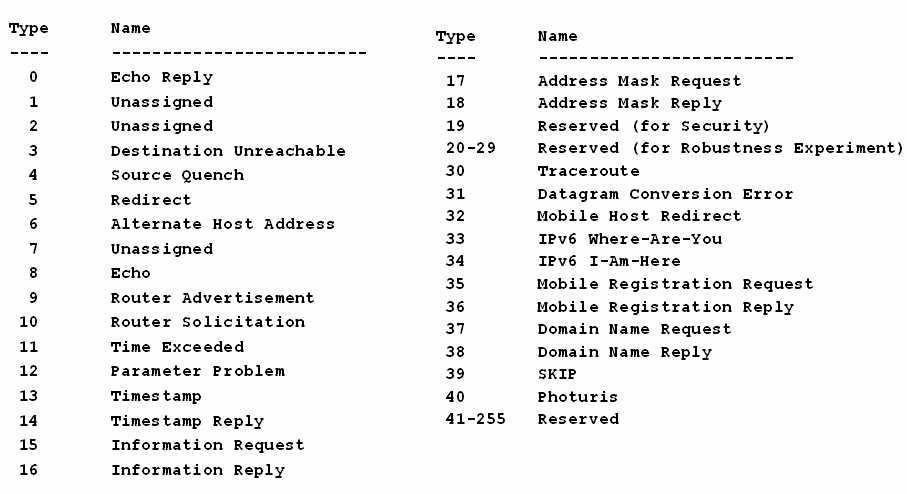
* + - * 1. Hata Raporlama Mesajlari
        2. Durum Kontrol Mesajlari



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 ulastirilamadigi 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 enapsule edilmis frameler icerisine gomulmuslerdir. Dolayisiyla ICMP mesajlari tarafindan olusturulmus hata mesajlari kendi ICMP mesajlarini yaratmazlar.

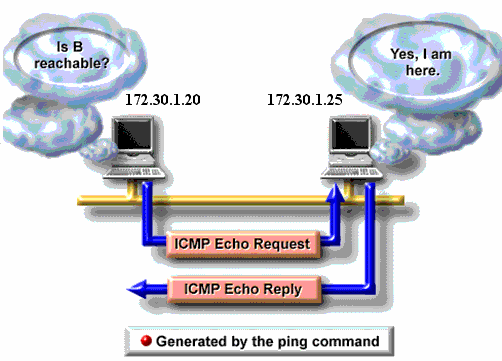
ICMP mesajlari Type’ lardan ve Code’ lardan olusur.



(Type 3, Hedefe ulasilamiyor mesaj code’lari)

# Ping ve Trace

Ping ve Trace komutlari network muhendislerine bir cok problemin teshisinde yardimci olar. Her iki komutta ICMP Echo Reuest ve ICMP Echo Reply mesajlari ile calisir.



Ping komutu ile ping istegini gonderen cihaz ICMP Echo Request’ te bulunur. ICMP mesajlarindaki Echo Request Type’i 8 ve Code’ u 0’ dir.

Hedef ip adresi Echo Request mesajini aldiginda gonderen cihaza Echo Reply ICMP mesajini gonderir. Bu mesajin Type’i 0 ve Code’u da 0’ dir.

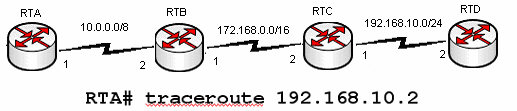
Trace komutu ise kaynak ve hedef ip adresleri arasinda ki olasi problemleri anlamaya yarar. Burada olasi problemler dememizin sebebi kaynak ve hedef ip adresleri arasindan birden fazla yol varsa her defasindan farkli yollar izlenebilir.

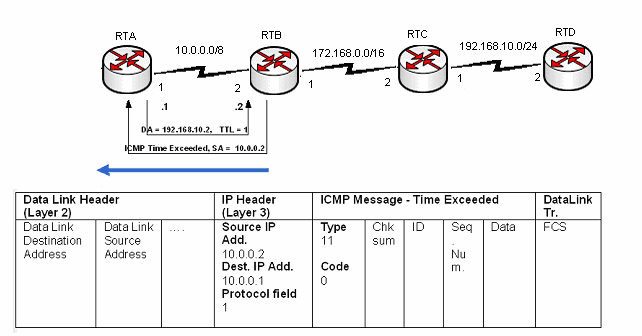
Trace komutu bilgisayarlarda, tracert *(ip adresi)*

Routerlarda, traceroute *(ip adresi)*

seklinde kullanilir. Traceroute calisirken ping (ICMP Echo) mesajlarini kullanir.

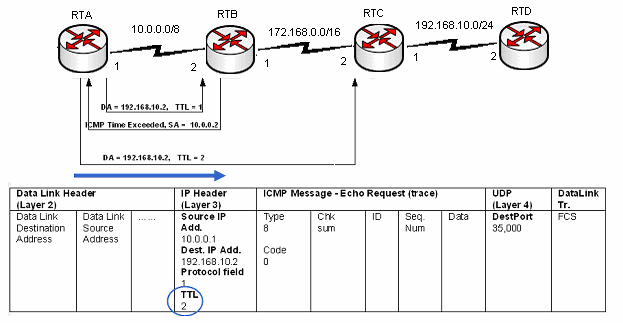
# Traceroute Ornegi



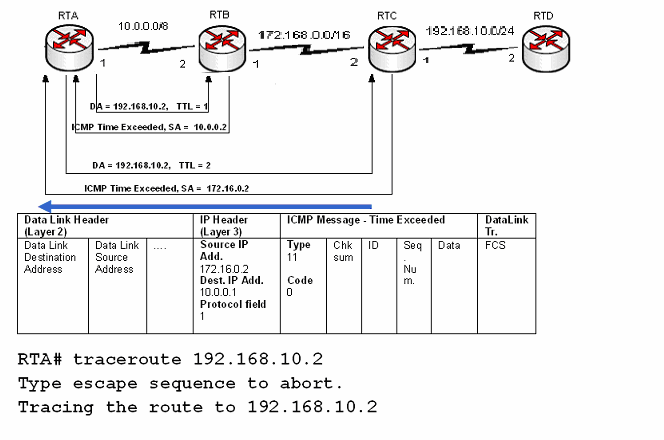


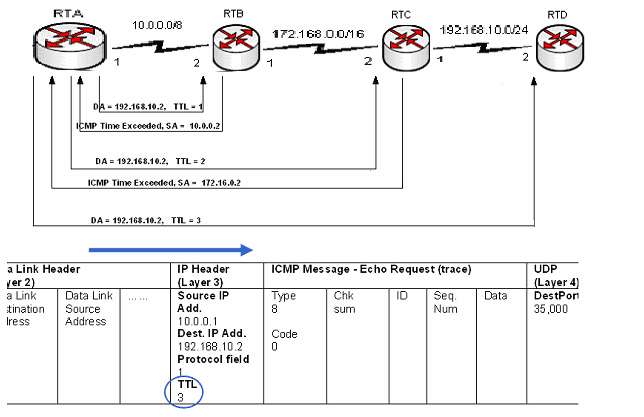
Traceroute basladiktan sonra IP basligindaki TTL degerini 1 yaparak ICMP Echo Requestte bulunur. RTB bu istegi aldigi zaman TTL degerine bakar ve bu deger 1 ise bir sonraki Routera gonderir, 0 ise Istek Zaman Asimi mesajini geri gonderir.

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



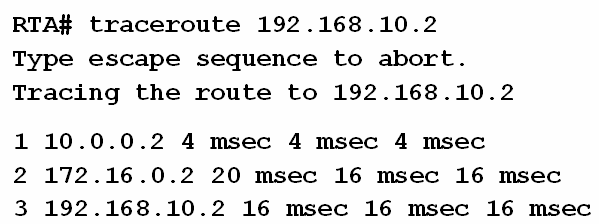
Artik RTB mesaji alip TTL degerine baktiginda 0 degil 1 gorecek ve dolayisiyla bu istegi RTC routerina gonderecektir. RTB ile yasananlar bu sefer RTC ile de yasanacak ve TTL degeri 0 olarak gelen Echo Requeste RTC Istek zaman asimi mesajini geri gonderecek. Burada RTC nin dondurecegi istek zaman asimi mesajinda source ip adresi olarak RTC’ nin adresi gorunecektir.





Bu sefer RTA TTL degerini 3 e cikararak yeni bir Echo Requestte bulunacaktir.Dolayisiyla paket RTD routerina kadar gidebilecektir. Burada TTL degerini 0 olarak alan RTD hedef ip adresi kendine direk bagli olan networkte bulundugu icin artik istek zaman asimi mesaji gondermez, ICMO Port Unreachable Mesajini geri dondurur. (Type=3, Code=3)

RTA routeri port unreachable mesajini trace ettigi network olarak algilar.



# 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. Routerın 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;

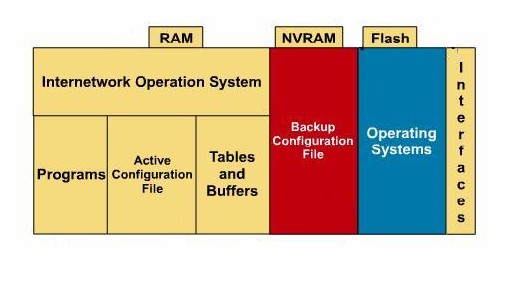
***Post;*** Router’ ın power tuşuna basıldığı anda devreye girer ve donanım testini gerçekleştirir.

***MiniIOS;*** Konsoldan giriş yapılarak ulaşılabilecek, IOS’ de bir sorun ile karşılaştığımızda sorun çözmemize yetecek kadar içeriğe sahip bölümdür. Burada TFTP servera erişilerek çeşitli yüklemeler yapılabilir.

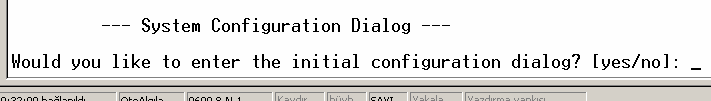
***Boostrap;*** Router’ ın çalışmasını sağlayan bir yazılımdır. Microsoft işletim sistemlerindeki “boot.ini” dosyasına benzetilebilir.

***ROM Monitör;*** Router’ ın BIOS’ u gibi düşünülebilir. Düşük seviyede hata ayıklama ve özelliklede ileride 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 faydalanılır.



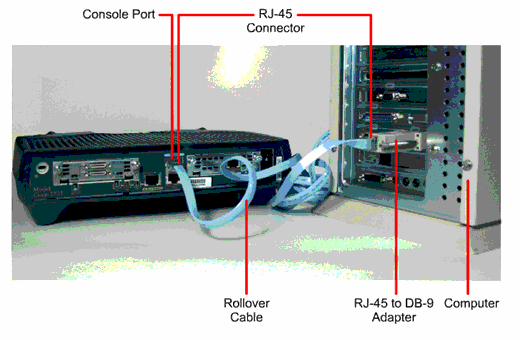
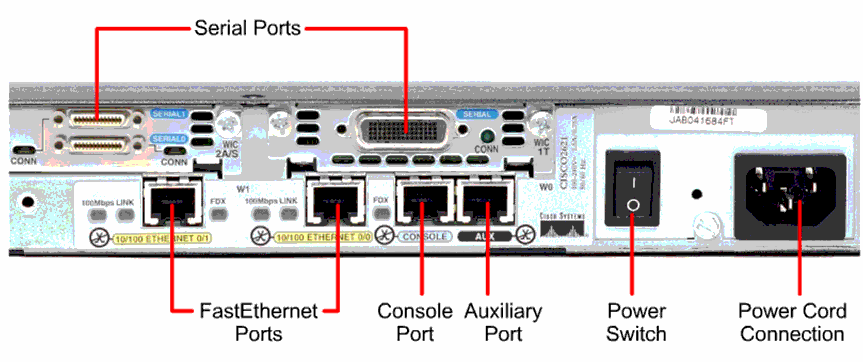
**NVRAM:** Non-Voltalite Ram’ in kısaltmasıdır. Yani kalıcı, silinmez bir RAM’ dir. Startup- Configuration denen başlangıç konfigürasyon dosyaları burada bulunur. Router açıldığından buradaki dosyayı alıp RAM’ de çalışmasını sağlar. NVRAM boş ise konfigürasyon için bir sihirbaz kullanmayı isteyip istemeyeceğimizi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



**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.





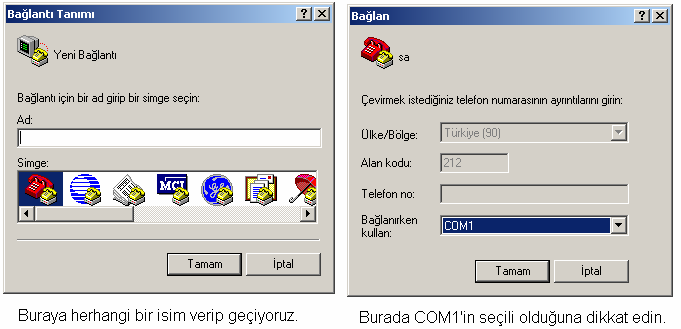
# IOS (Internetworking Operating System)

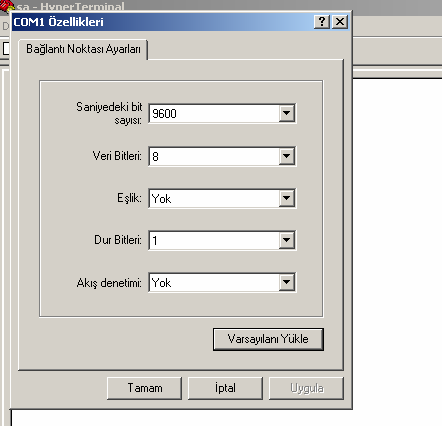
Adından da anlaşılacağı gibi IOS, Router ve Switch’ lerin yönetilmesinde kullanılan işletim sistemidir. IOS bize CLI (Command Line Interface) denen text görünümünde bir arayüz sağlar.

Bu arayüze erişmenin üç temel yolu vardır. Consol Port, Auxilary Port ya da Telnet vasıtasıyla 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 kablosuda denir. Hyper Terminal yardımıyla CLI’ e erişilebilir.

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





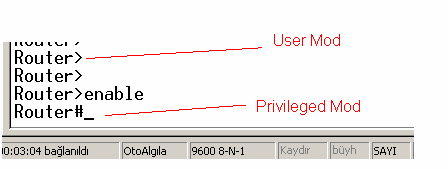
Burada “Varsayılanı Yükle” dediken sonra Tamam’ a basıyoruz ve routerımıza erişimimiz tamamlanıyor.

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

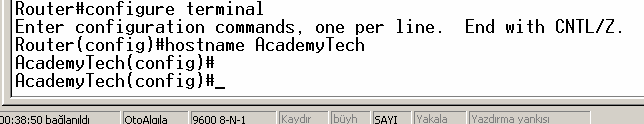
# Router Çalışma Modları

User Mod: Router’ ı açıp arayüze eriştiğimiz 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.



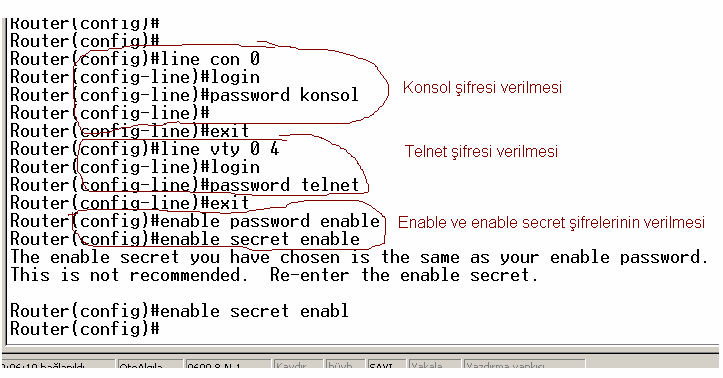
Global Configuration Mod: Config Mod diye de anılan bu moda geçmek için enable moda iken “configure terminal” yazılır ve entera basılır. Bu modda yapılan değişiklikler bürün Router’ı etkiler. Örneğin bu modda iken bir router’a isim verilebilir. Bu mod ileride detaylı anlatacağımız alt modlara ayrılır.



# 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.



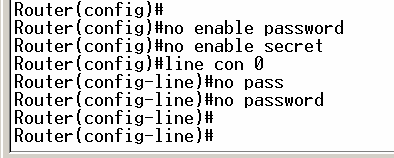
(Dikkat edilirse enable ve enable secret şifrelerinin aynı olmasına izin verilmiyor)



(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 vasıtasıyla erişileceği bilgisini vermiş oluyoruz. Bütün komutlar başına ”no” yazılarak geçersiz hale getirilebilir.

“ no enable secret” gibi bir komut ile enable secret şifresini kaldırabiliriz.

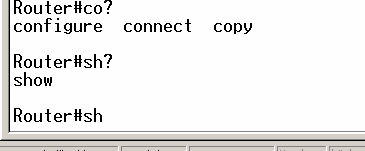


# Yardım Alma

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



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

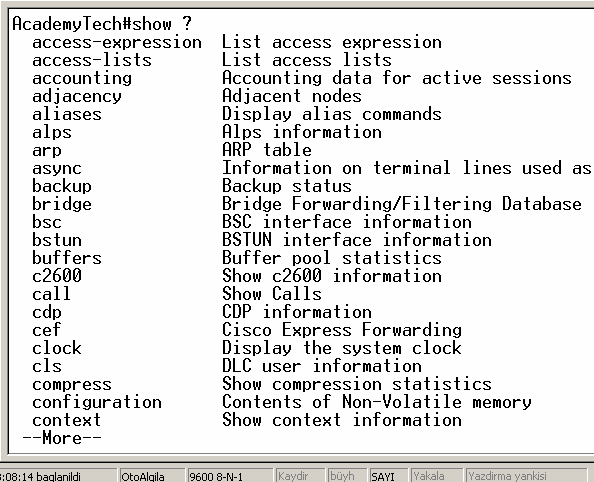


Konuyu tam olarak kavrama da AcademyTech laboratuarlarında sıkça uyguladığımız bir çalışma da (Clock uygulaması) aşağıda detaylı bir şekilde gösterilmiştir.



# Show Komutları

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



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

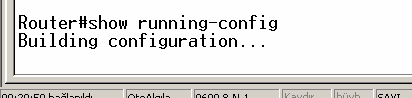
# Konfigürasyon Dosyaları

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 boşa zaman harcatan bir çok soru ile doludur ki önerdiğimiz ve uyguladığımız konfigürasyonu manuel yapmaktır.

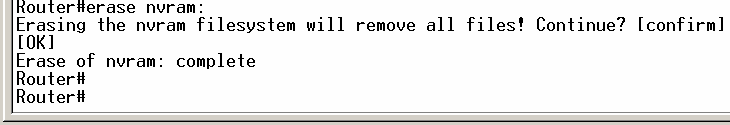
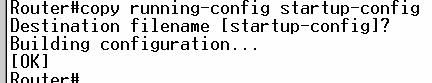


**Running-Config:** RAM’da bulunur ve Router’ın çalıştığı andaki konfigürasyonunu tutar. Router kapatıldığında buradaki bilgiler gider.



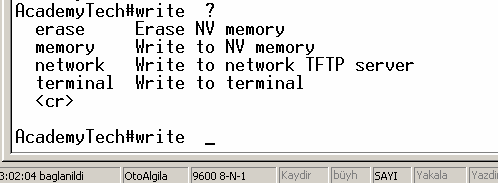
Bir Router yeniden başlatıldığı zaman startup-config dosyası dolu ise, IOS tarafından bu dosya alınıp RAM’a aktarılır ve dolayısıyla o artık Running-config olmuştur.

Bir router’ ın running-config ve startup-config dosyalarını “show” komutu ile görebilir, “copy” komutu ile birbirleri üzerine kopyalayabiliriz.

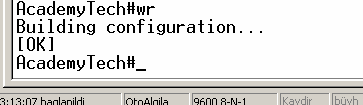


# Write Komutu

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



NOT: Kısaca “wr” yazmak Running Konfigürasyonumzu NVRAM’a kayıt edecektir.



# Şifre Kırma

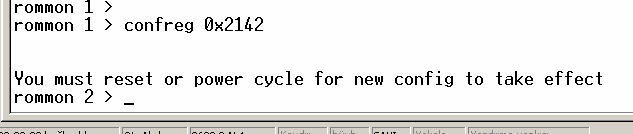
Routerın şifrelerini unuttuğunuzu ya da ikinci el bir Router aldığınızı ve bu router’ ın konfigürasyon dosyalarının hala üzerinde olduğunu dolayısıyla şifrelerini bilmediğinizi varsayalım. Böyle bir durumda şifreyi değiştirmek ve istersek eski konfigürasyonun bozulmamasını da sağlayarak bunu yapmak mümkündür. 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.



1. “confreg” komutu ile başlangıç register’ ı 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.



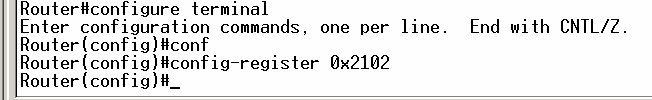
1. Router yeniden başlatılır. Açıldığında Router’ ın herhangi bir şifre sormadığını göreceksiniz.
2. Enable moda geçilir. Bu moda geçtikten sonra artık istediğimiz her şeyi yapabileceğimize göre, eski konfigürasyonu kaybetmek istemiyorsak, “copy

startup-config running-config” komutu ile o dosyayı alır ve şifreleri değiştirip yeniden NVRAM’ a kaydederiz.



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

1. Son olarak Rom Monitör’ e girip değiştirdiğimiz register’ ı eski haline getirip (0x2102) getirip Router’ ımızı yeniden başlatabilir ve eski konfigürasyon ve yeni şifreyle router’ın açıldığını görebiliriz.



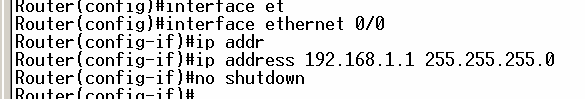
# Temel Router Konfigürasyonu

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

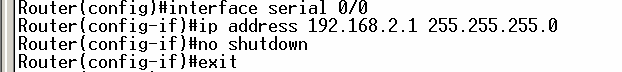
Bir router’ ın interfacelerinden herhangi birine ip adresi atamanın diğerinde farkı yoktur. Yapılacak işlemler sırasıyla interface konfigürasyon moduna geçmek, ip adresini subnet maskı ile birlikte yazmak ve “no shutdown” ile interface’ i aktif hale getirmektir.

Örnek bir çalışma olarak Router’ımıza şu ip adreslerini atayalım. Ethernet Interface Ip adresi : 192.168.1.1 / 24

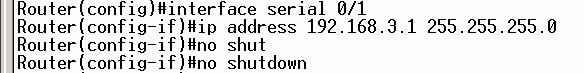
Serial (0/0) Interface Ip Adresi: 192.168.2.1 /24 Serial (0/1)Interface Ip Adresi : 192.168.3.1 /24



(Ethernet 0/0 interface’ine ip adresi verildi)



(Serial 0/0 interface’ ine ip adresi verildi)



(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

------------------------------------------------------------------------------------------------------------

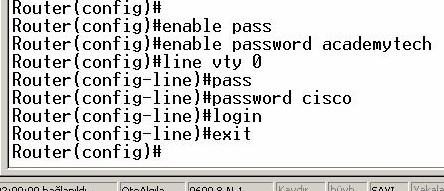
# Router’a Telnet İle Bağlanma

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

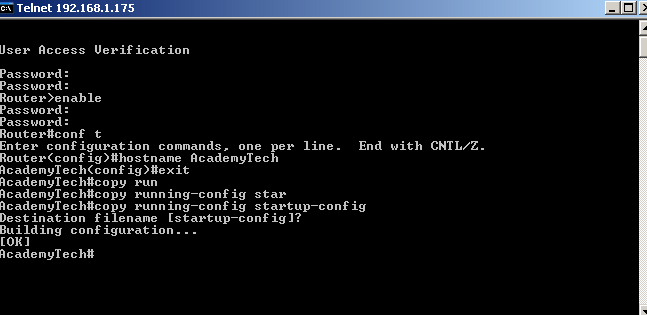
Fakat bunun için bazı şartların yerine gelmesi gerekir. Öncelikle Router’ ın ethernet interface’ i up olmalıdır ve Telnet, Enable şifreleri verilmiş olmalıdır. Telnet şifresi verilmediğinde “Password Required, but none set” şeklinde bir hata mesajı alınacak ve bağlan gerçekleştirilemeden kaybolacaktır.



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



(Telnet ve Enable Şifrelerinin Verilmesi)



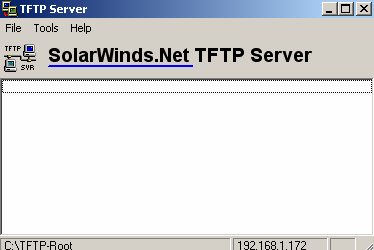
Görüldüğü gibi şifreler verildikten sonra bağlantı gerçekleştirilebilir ve her türlü konfigürasyon yapılabilir. Komutlarda herhangi bir farklılık söz konusu değildir.

# 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 sonrasını kendisi halledecektir.



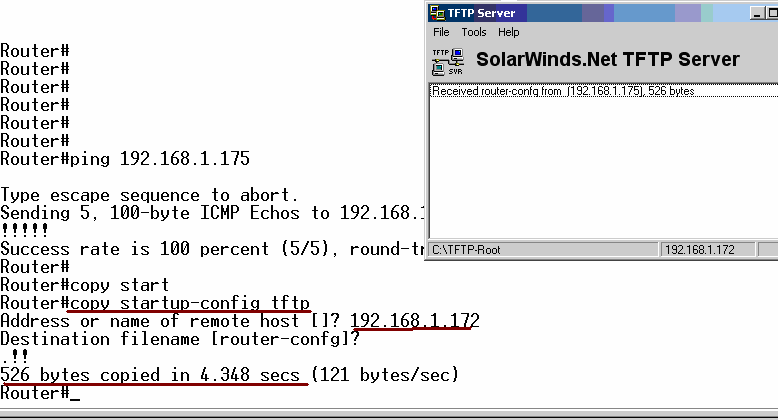
Copy startup-config tftp: Copy running-config tftp:

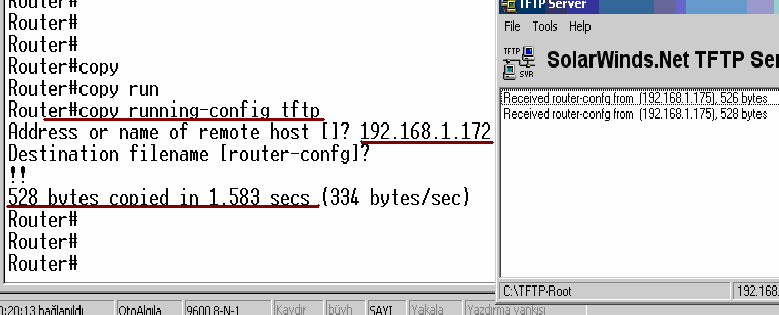
Copy flash tftp

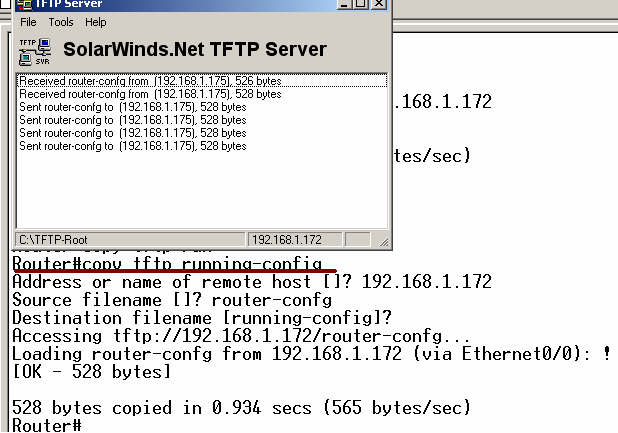
gibi bir komut yazdığımız da bize ilk olarak TFTP Server’ın ip adresi ve şayet Flaş’ ın yedeğinin alacaksak onun tam adını soracaktır. Ve bütün bunlar yapılırken TFTP Server çalışıyor durumda olmalı.

TFTP Serverdan geri yüklemelerde ise komut tam tersi yazılarak çalıştırılacaktır. Copy tftp startup-config

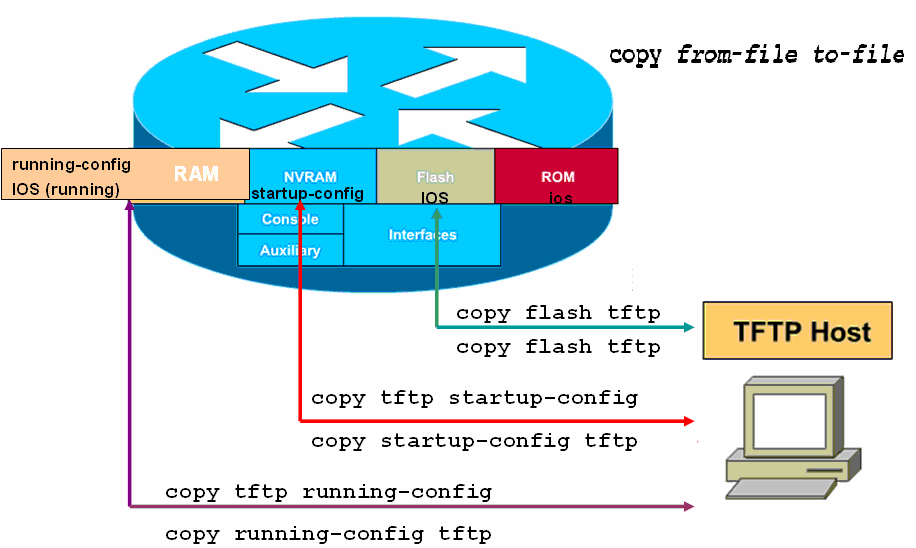
Copy tftp running-config Copy tftp flash





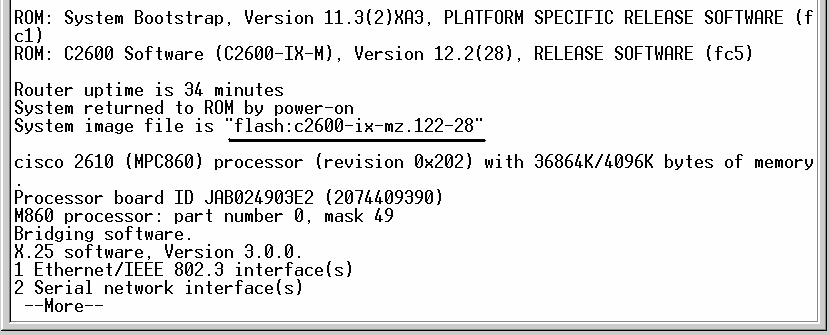


# Copy Komutlari Ozet

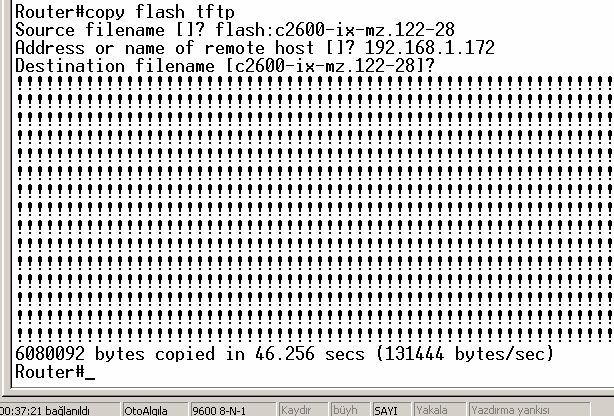


**IOS Yedek Alma ve Yükleme**

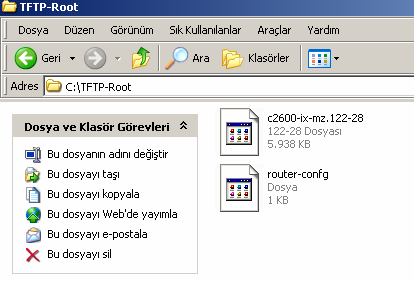
TFTP Server kullanarak IOS’ in yedeği alınabilir veya IOS yüklenebilir. Bunun için Sistem Image File’ ın tam dosya adı bilinmelidir ve bu “show version” komutu ile öğrenilebilir. Alınan bütün yedekler gibi IOS’ in yedeği de TFTP Server tarafından TFTP-Root klasörünün altına atılır.



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



(“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 basılara Rommon’ a girilir ve konsol hızı 115200 bps’ a .ıkartılır. (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 bölümleri 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**

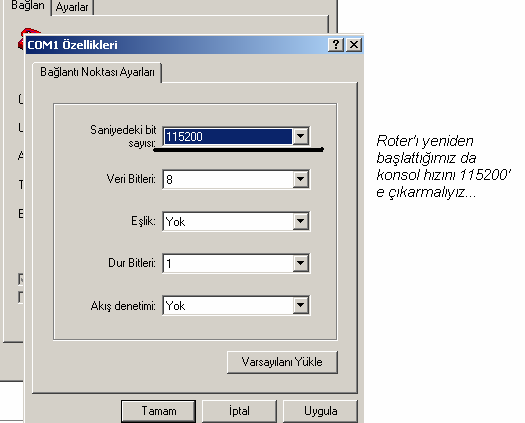
# enter rate: 0 = 9600, 1 = 4800, 2 = 1200, 3 = 2400

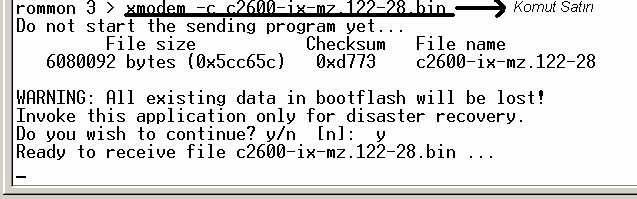
**4 = 19200, 5 = 38400, 6 = 57600, 7 = 115200 [0]: 7**

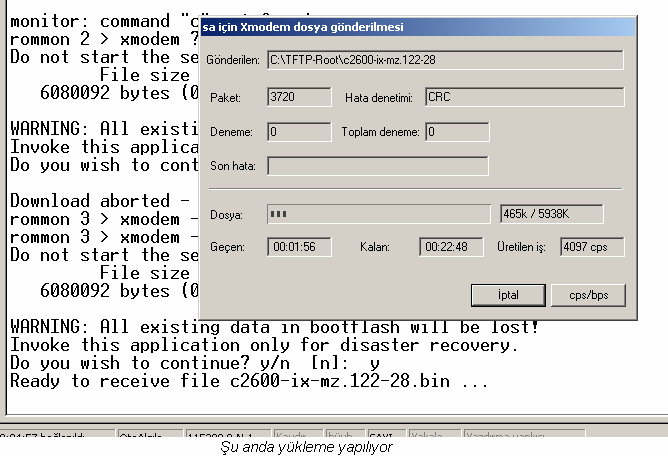
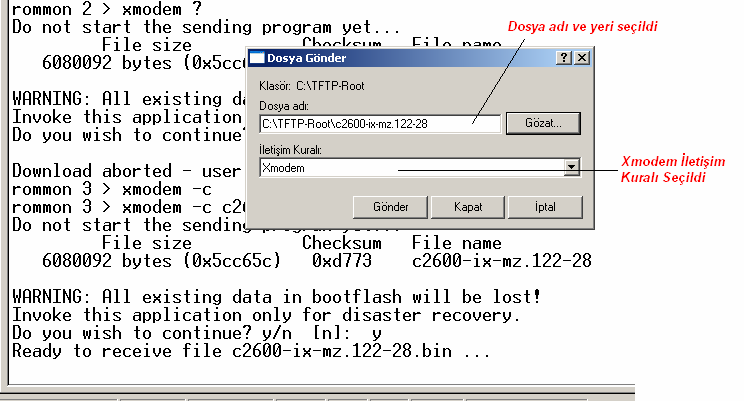
change the boot characteristics? y/n [n]: n

# 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” özelliğindenfaydalanacağız.



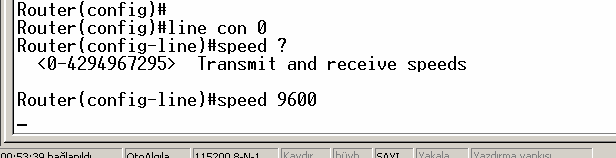




IOS yüklenmesi tamamlandıktan sonra Routerımız açabilirz. Fakat yapmamız gerekenler henüz bitmiş değil. Routerımızın konsol hızı hala 115200’ de. İlk açtıuğımızda Hyper Terminal ile bu hızı göz önüne alarak bağlanıp, standardı sağlama için konsol hızını tekrar 9600 bps olarak değiştirmemiz gerekir.

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

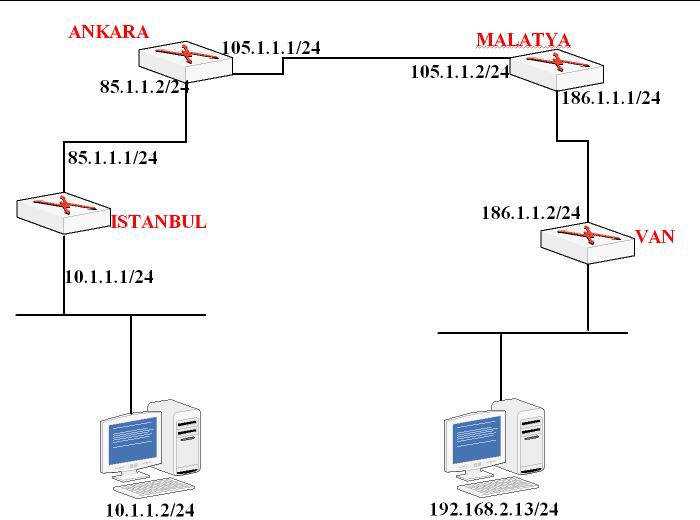
Ve bağlantımız kesildi çünkü Hyper Termina ile bağlantımızı oluştururken konsol hızı olarak 115200 bps’ ı seçmiştik. Bunu da eski haline getirmemiz gerekir.



# 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 bulunduğunda, 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’ ı 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 İstanbu Router’ ının sadece 10.1.1.0 ve

* + - 1. network’ lerini bildiğini söyleyebiliriz.

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

* + - * + Routing işlem, Bir paketin bir Networkdeki bir aygıttan 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 seçmektedir.
        + 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.

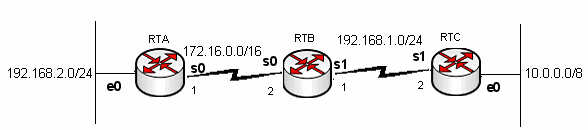
Peki İstanbul Router’ ına bütün tanımlamaları yaptıktan sonra uzak bilgisayara ping atabilir miyiz

?

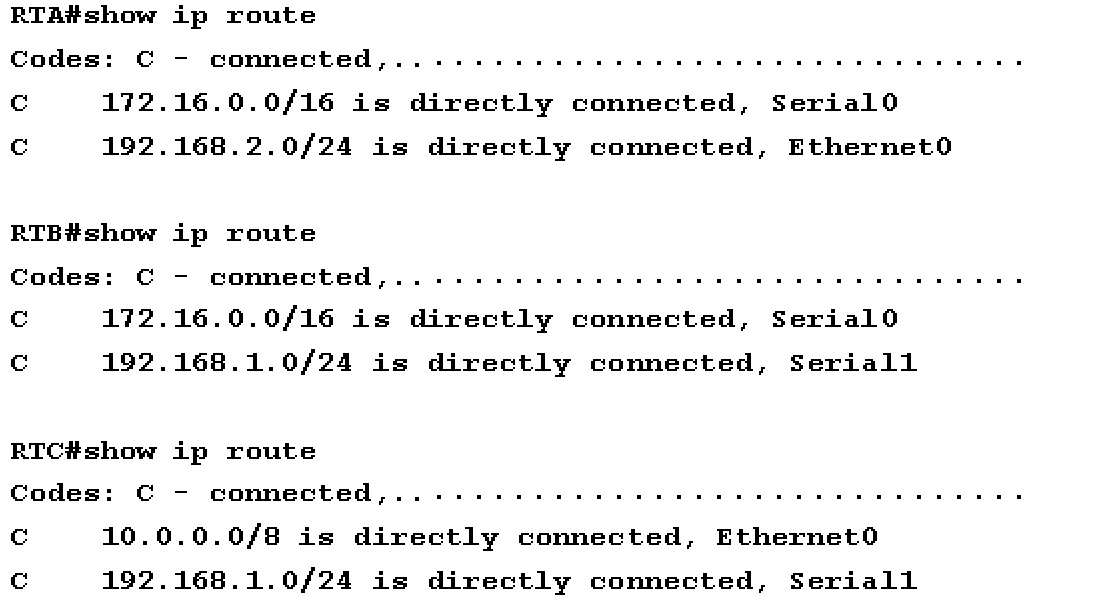
Hayır…

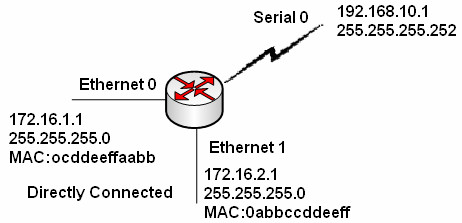
Biz sadece İstanbul Routerında Static Routing yaptık. Malatya Routerında hiçbir işlem yapmadığımızdan dolayı Malatya Routerı ping işlemine cevap vereceği ip adresine nasıl ulaşacağını bilemediği için Ping işlemi gerçekleşmeyecektir. (Ping işlemi iki yönlüdür, paket hedefe gider ve gelir.)

Daha oncede belirttigimiz gibi Routerlar icin Directly Connected networklerine herhangi bor yonlendirme yazmaya gerek yoktur. Iki Routeri birbirine bagladiginizda ve interfacelerini uygun sekilde konfigure edip up durumuna getirdiginizde Routing Table’ larda o networkler ile ilgili bilgileri goruruz.



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





# ROUTING BASICS

Routerların temel işlevi yönlendirmek yapmaktır. 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 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

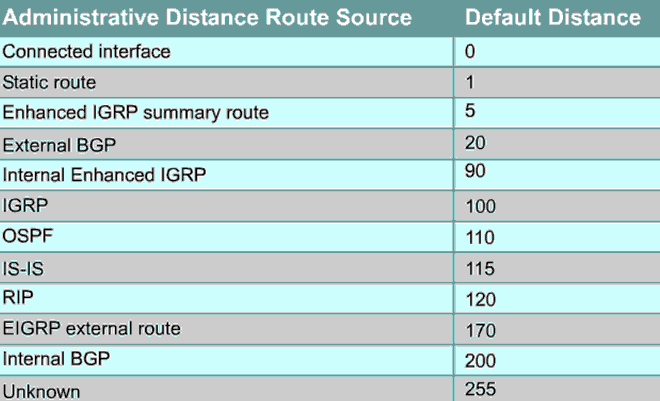
Az önce de bahsettiğimiz gibi static Routing “ip route” komutu ile Global Configuration modda yapılır ve küçük ölçekli networklerde ideal çözümdür.

Static Routing yapılırken 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.

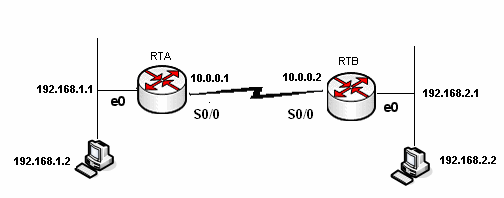
# Router(config)#ip route [hedef adres][subnet mask][Next Hop] [distance]

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

Default Administrative Distance degerleri sunlardi:



Static Routing, i ornek bir calisma ile inceleyecek olursak;

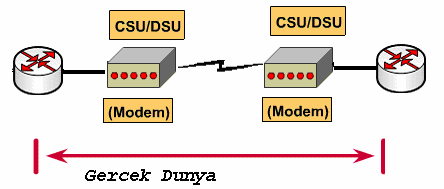


192.168.1.0 ve 192.168.2.0 10.0.0.0 networklerimiz var. Bütün subnet masklarımız 255.255.255.0 olsun. Bu durumda Static Routing işlemi her iki router için şu şekillerde gerçekleştirilmelidir. A Routerının serial 0 adresi 10.0.0.1ve DCE iken B routerının serial 0 adresi 10.0.0.2’ dir.

**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 çalışmayacaktır. Örneğin;

# RouterA(conf-if)#clock rate 64000



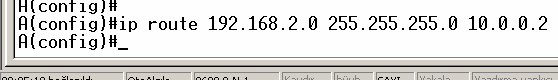




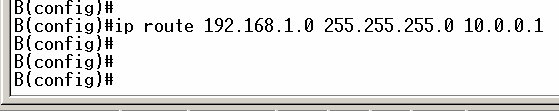
(CSU/ DSU)

# Artik konfigurasyonumuza gecebiliriz.

Router A için;



Router B için;



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 routerlardan herhangi biri secilebilir, burda onemli olan konfigurasyonlar bittigi zaman Routerimizin next hop adresine nasil ulasacagini bilip bilmedigidir.

Routerlarin konfigurasyonlari ve problem cozumu asamasinda running-config dosyalarinin 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

!

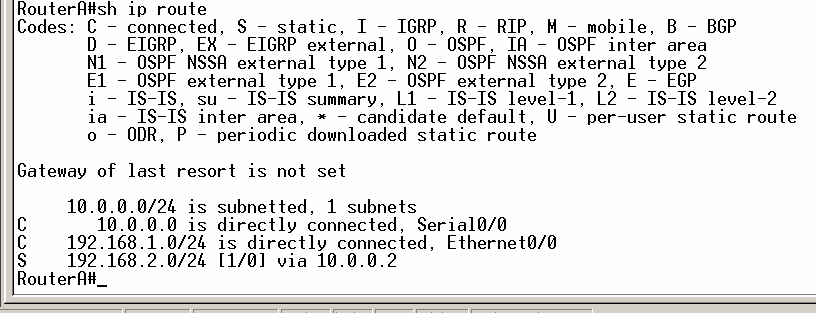
no scheduler allocate end

-------------------------------------------------------------------------------------------------------------

-

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

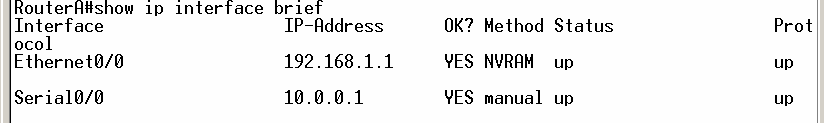
“Show ip route” komutu ile yönlendirme tablosunu görebiliriz.



(RouterA için yönlendirme tablosu)

Yönlendirme tablosunda “S” ile başlayan satırlar Statik bir yönlendirme yapıldığını ve şekilden 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 direk olarak bağlanmış networkleri gösterir ve bu networklere “Directly Connected” networkler denir. Roterlar kendi Directly Connected networklerini bilirler ve bu networkler ulaşmak için yönlendirme yapılmasına gerek yoktur.

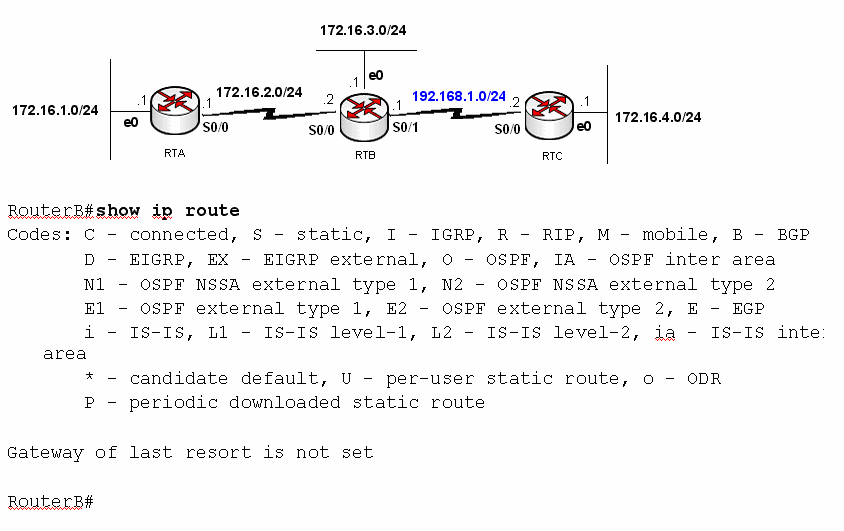
Senaryomuzda hiçbir yönlendirme yapmasaydık bile A router’ınan ethernet interface’ ine bağlı bir bilgisayardan B router’ ının serial interface’ ine ping atabilirdik. Bunun için tek yapmamız gereken şey, o bilgisayarda Default Gateway’ i (Varsayılan Ağ Geçidi) 192.168.1.1 olarak konfigüre etmektir.



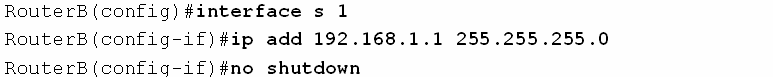
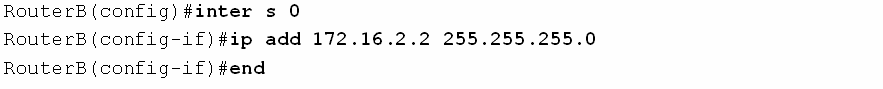
(“show ip interface brief” komutu ile interface’ leri durumunun görüntülenmesi)

# Routing Table

Routing Table konfigurasyonlarimiz ve projelerimiz sirasinda problem teshisimiz acisindan cok onemlidir. Iyi bir network yoneticisi running-config ve routing table’ a hakim olmalidir. Isimizi Routing oladugu durumda Routing Table bir numarali yardimcimiz olacaktir.

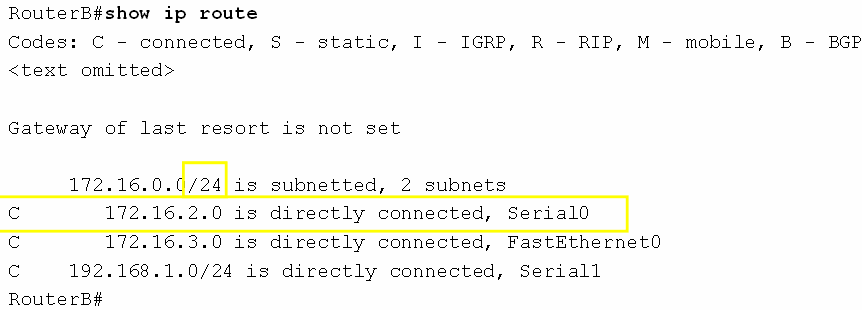


Iste ornek bir network topolojisi B routerinin Roting Table’i. Routing Table gorundugu uzere bombos. Burada Directky Connected networklerin bile gorunmemesinden dolayi interfaceler ile ilgili bir sorun oldugundan bahsedilebilir. Sorunun ne oldugu ile ilgili bilgiyi “show ip interfaces brief” komutu ile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





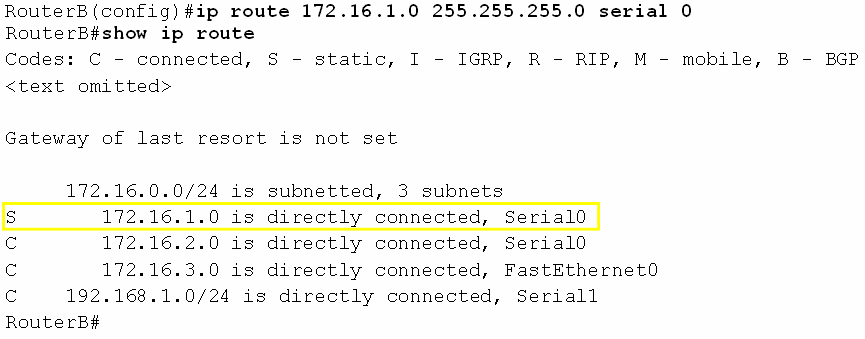
Router B icin interfaceleri up duruma getirdikten sonra artik en azindan Rouring Table’ imizda Directly Connected networklerimizi gormemiz gerekir.



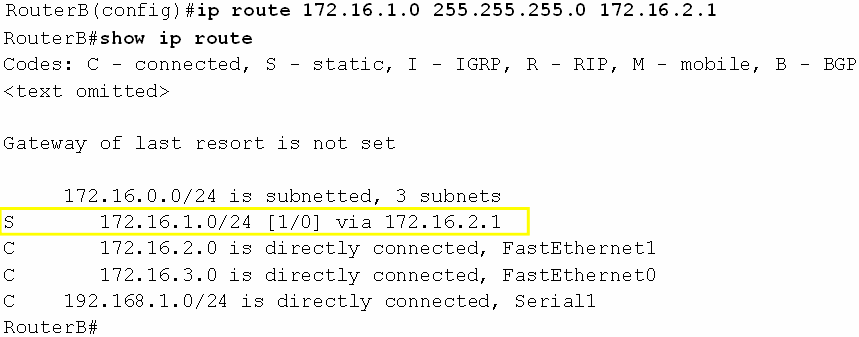
Burada 172.16.0.0 networkune Parent route ve o networkun subnetworku olan 172.16.2.0 -

* + - 1. networklerine Child Route denir.

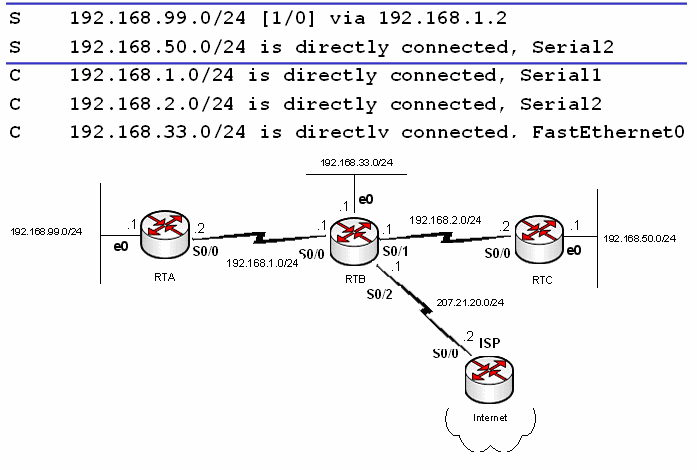
Bilindigi gibi Static Route yazilirken Routerin interface’ i yada Next Hop ip adresi kullanilabilir. Routerin interface’ i kullanildiginda o static route satiri interface ile direk bagli bir network gibi gorunecektir.



Next Hop ip adresi kullanildiginda ise Routing Table o networke verilen ip adresi ile ulasabilecegini gosteren satir yer alacaktir.



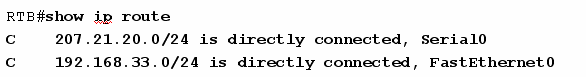
Interfaceler “up” oldgu surece Routing Tablelarda bozulma olmayacaktir.



Ornegin sekildeki yapi icerisinde RTB Routeri icin butun interfaceler up durumdayke Routing Table sorunsuz gorunuyor. 192.168.1.0 ve 192.168.2.0 networklerinin bulundugu interfacelerin bir an icin down oldugunu dusunelim.

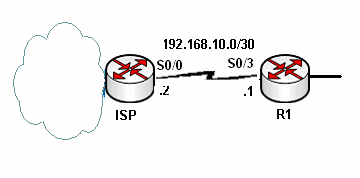


Bu durumda Routing Table RTB icin su sekilde olacaktir.



Soz konusu Interfacelere direk bagli olan networkler ve o interfaceleri kullanarak yazilan Static Route satirlari artik Routing Table’ da yoklar.

# Default Routing



Burada ISP Routeri ile baglanilan networke (Internet) R1 uzerinde default route yazilarak ulasilabiliecektir.

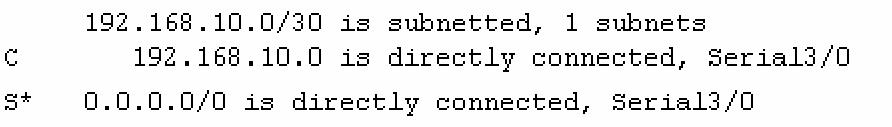
Default hedefi bilinmeyen paketleri yonlendirmek icin uazilabilecek Route satiridir seklinde tanimlanabilir.

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.

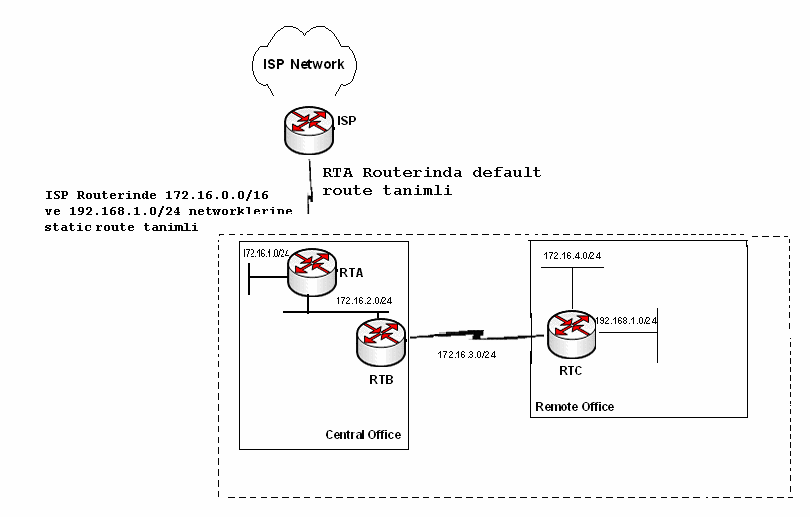


# Extra

Default tanimlamak bazen sorunlar ile birlikte gelebilir.

Cunki uzerinde Default tanimli ve bu route satiri ile paketleri internete 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 yapi icerisinde RTB ve RTC arasindaki baglantinin down oldugunu dusunursek Routing Table lar update edildikten sonra RTA ve RTB routerlari 172.16.4.0 ve 192.168.1.0 hedef networklerine giden yollari bilmedikleri icin hedefinde bu networkler bulunan paketleri default route satirindan hareketle ISP routerina gonderecekti.

ISP Routeri da kendine gelen bu paketleri uzerinde tanimli statik route satirlarindan hareketle tekrar geri gonderecek ve bu sebeple bir dongu olusmasina sebep olacaktir. Bu dongu IP basligindaki TTL (Time – to - live) alani sifirlanana kadar devam edecektir.

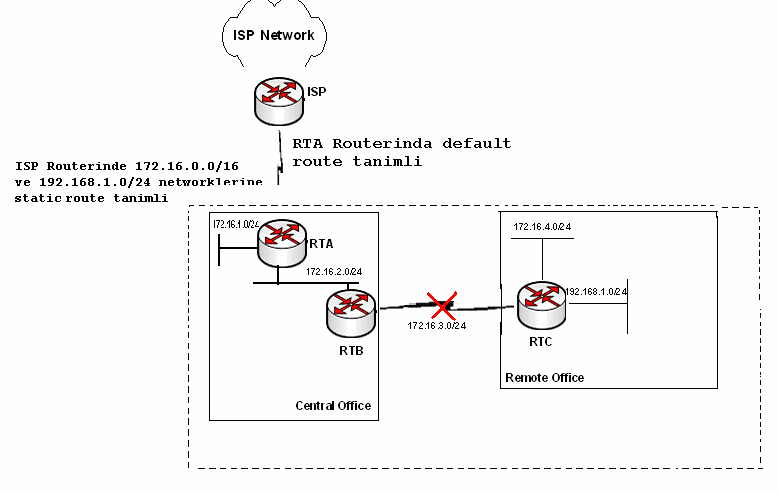
Bunun icin kullanilacak cozum RTA uzerinde Discard Route denen tanimlamayi yapmaktir.

Discar Route routing table’ da bir eslesme olmadiginda ve default route’ un isletilmesi istenmediginde kullanilir ve paketler null0 ‘ a gonderilir.

# Ornegin;

RTA(config)#ip route 172.16.0.0 255.255.0.0. null0

Satiri 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 altinda 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 arasinda 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 tarafında 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 kom şu routerlarından da aynı bilgileri alırlar. Bu döngünün sonunda her router sistemde 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.

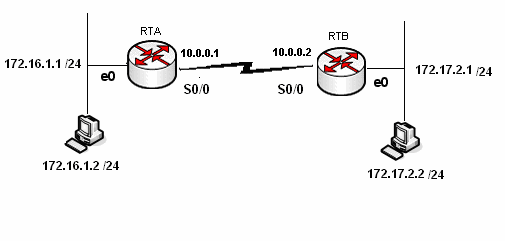
# Distance Vector Protokoller RIP (RIPv1)

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.

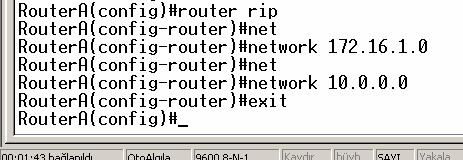
Rip ile tanımlanan routerlar her 30 saniyede bir kendisinde tanımlı olan networkleri komşu routerlarına iletirler. Burada dikkat edilmesi gereken bir konu, RIP ile tanımlanan bir networkün bağlı bulunduğu interface’ i, aynı zaman da routing update gönderilecek bir interface olarak seçiyor olmamızdır.

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

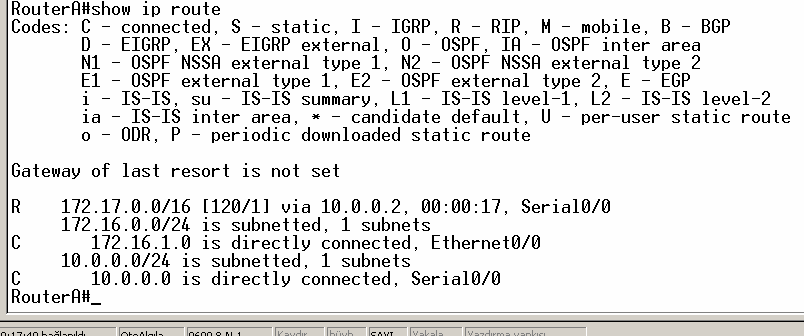
Rip konfigürasyonu diğer bürün routing protokoller de olduğu gibi oldukça basittir. (Bütün subnet maslar 255.255.255.0)



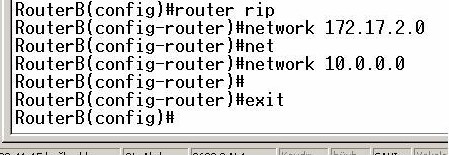
Bu senaryoyu Rip ile konfigüre edecek olursak;



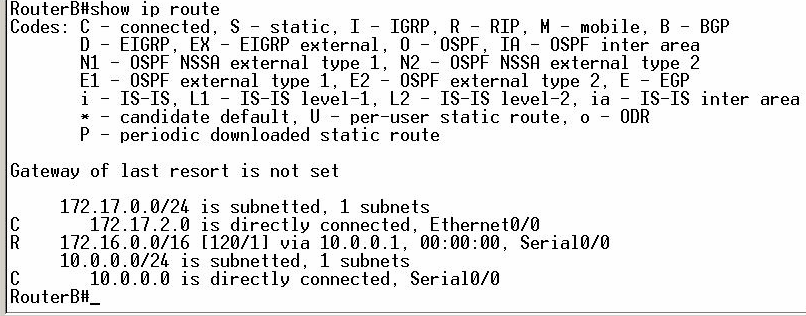
(RouterA Rip Konfigürasyonu)



(RouterA Yönlendirme Tablosu)



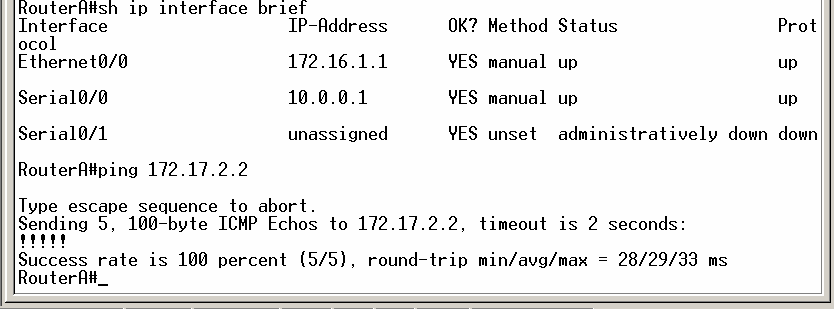
(RouterB Rip Konfigürasyonu)



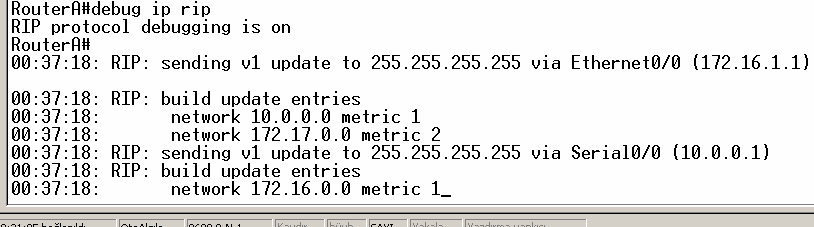
(RouterB Yönlendirme tablosu)

Routing Table’ ımıza “Show ip route” komutu ile baktığımız da 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 aşılacak hop sayısıdır.

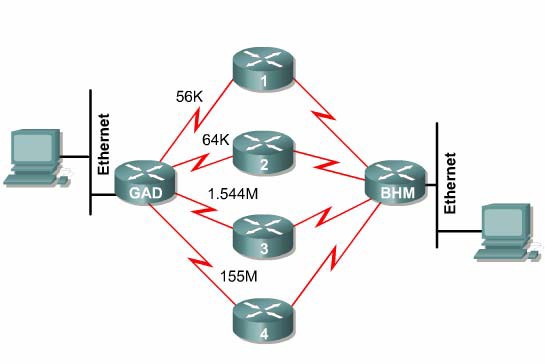


Rip protokolü updatelerini broadcast adresi olan 255.255.255.255 ip’ sinden yapar. “debug ip rip” komutunu verdiğimizde bunu açıkca görürüz.



# Rip Load Balancing

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



(Routerlar metricler eşit olduğu için load balancing yapar)

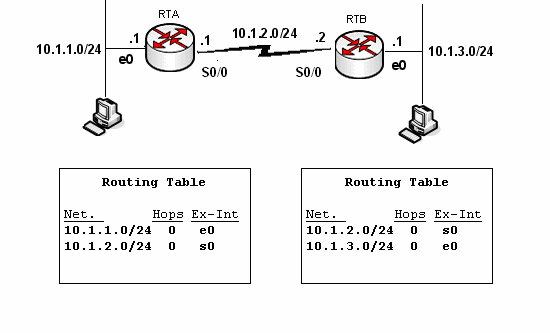
Mantıksal olarak düşündüğümüz de Rip’ in load balancing yapma ihtimali her zaman vardır. Çünkü referans olarak bir tek hop sayısına bakar. Oysa diğer protokoller de load balancing ihtimali en iyi yol seçimi sırasında bir çok kriter göz önüne alındığı için mucize derecesinde zayıf bir ihtimaldir. Fakat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

# Split Horizon

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

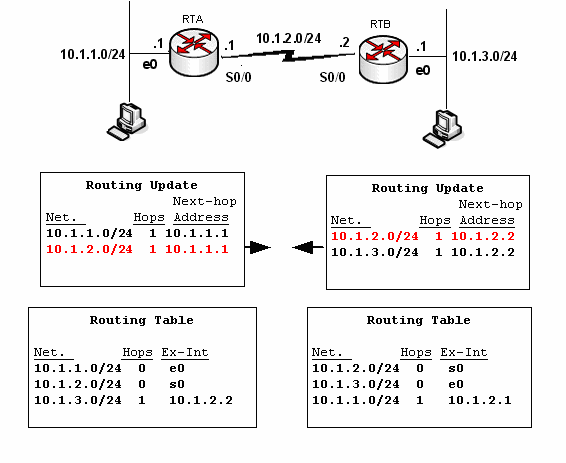
Ayrıca router’ın ağ üzerinde herhangi bir değişiklik olduğunu anladığında bu değişikliği, öğrendiği interface haricindeki interface’lerden yayınlamasını sağlar. Böylece router’lar değişikliği sadece bir yönde yayınlarlar.

Asagidaki ornek ile Split Horizon kuralini detayli anlayabiliriz.



Iki adet Routerimiz var ve baslangica Routing Table’ lar sekildeki gibi olusmus durumda yani sadece Directly Connected networkleri biliyorlar.

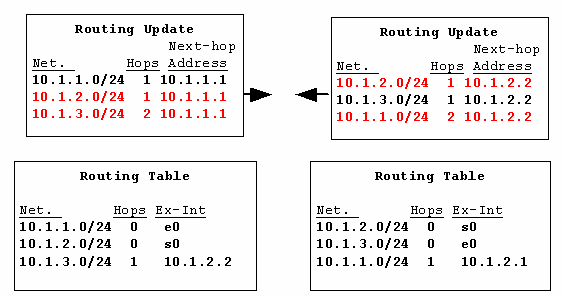
Split Horizon Disable edildigi zaman Routerlar Routing Table’ larinda ki butun networkleri ve herhangi bir interfacelerinden ogrendikleri butun networkleri update edeceklerdir.



Routerlar sekilde gosterilen updateleri komsu Routerlarina yapacaklar. Burada kirmizi ile gosterilmis Directly Connected networklerinde update edildigine dikkat edin. Bu update Split Horizon, un disable olmasinin sonucudur.

Routing Table incelendiginde bir sorun yok gibi gorunuyor. Gercekten de yok, cunki split horizonun disable olmasindan kaynaklanan updateler daha yuksek metrige sahip oldugu icin Routing table’ larda yer almadi.

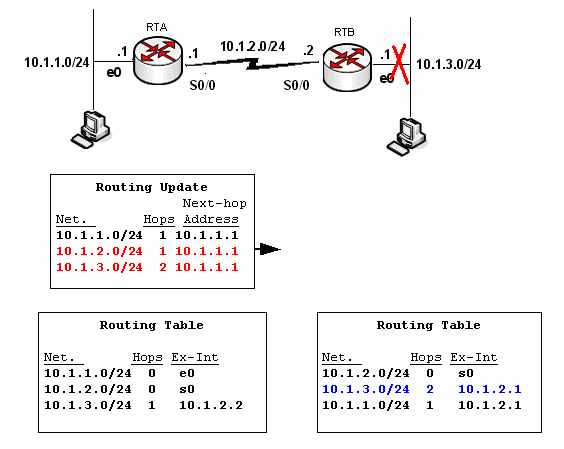
Simdi bir sonraki updatelere bakalim.



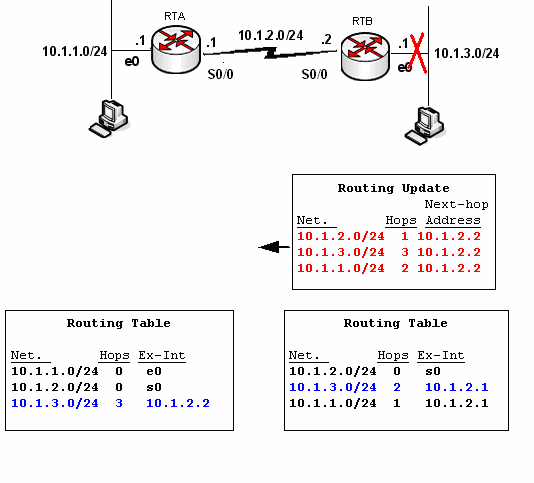
Burada daaslinda update edilmemesi gereken networkler update edilmis olmasina ragmen bir sorun yok cunki o networkler daha buyuk metric ile update ediliyor. Ornegin RTA Routeri

* + - 1. networkunu serial 0 interfaceinden aldigi icin split horizon disable edilmemis olsaydi o interfaceden geriye update etmeyecekti.

Bu ana kadar bir sorun olmadi ama bir an icin RTB Routerina bagli olan 10.1.3.0 networkunun down oldugunu varsayalim.

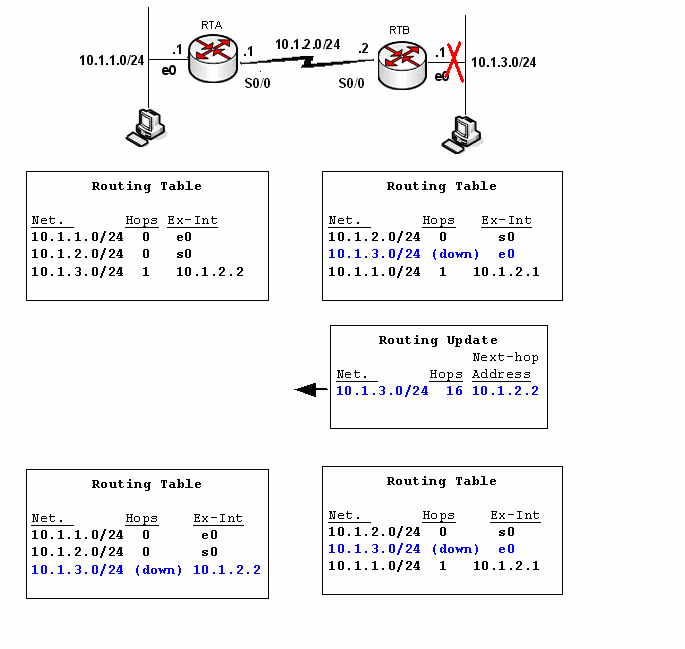


Bu durumda RTB routerina RTA routerindan 10.1.3.0 netwroku kendisine baglai olan network down oldugu icin daha kucuk metric ile geliyormus gibi olacak ve RTB routerinin Routing Table’ inda sekildeki gibi yer alacak. Su anda RTB Routeri bir sure once kendisine direk bagli olan networke diger router uzerinden 2 hop gecerek gidebilecegini saniyor. RTB updatelerini ustelik yanlis olan Routing Table,’ ina dayanarak yapacaktir.



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 edecektir. (Rip maximum 16 ho 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.

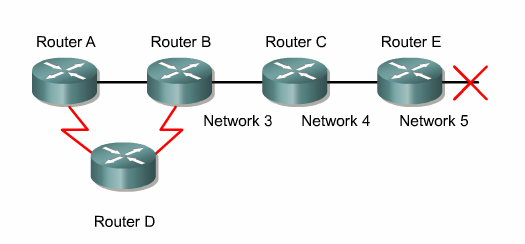
Dolayisiyla her ikir router da 10.2.3.0 networku icin Hol Down Timer’ i baslatirlar.

Bu calisma yapisina Split Horizon with Poisen Reverse denir ki Routerlar, da default olarak enable durumdadir. Disable edilme gereken zamanlar da ki CCNA 4 icerinde bu konudan bahsedegiz, asagidaki komut kullanilabilir.

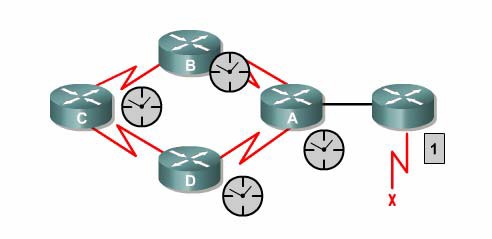


# Route Poisoning

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 sayıcılar 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 degisiklik oldugu anda Routerlar tarafinda gonderilen updatelerdir. Topoloji degistigi anda bunu farkeden Router periodic update suresini beklemeden degisikligi komsu Routerlarina bildirir.

Triggered Updateler Route Poisoning ile tumlesik calisirlar.

# Extralar

Timers Basic ve uptede timer komutlari ile Rip update, holdwoen v.s sureleri degistirilebilir. Router(config-router)#**timers basic** *update invalid holddown flush*

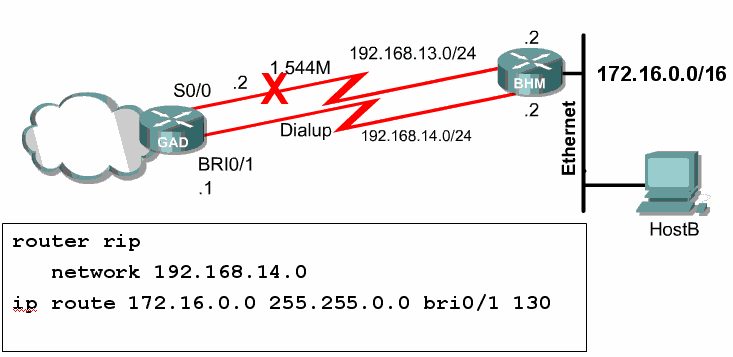
Router(config-router)#**update-timer** *seconds*

# Rip ve Floating Static Route

Floating Static Routelar backup route olarak tanimlanmis route’ lardir. Bu Route’ lar reel olarak calisan route lara gore daha yuksek bir Administrative Distance ile konfigure edilmelidir. Rip ile calisan bir Floating Static Route icin Administrative Distance degeri 120’ den buyuk olmalidir.

Bu durumda Rip sorunsuz calistigi surece Floating Static Route Routing Table’ da gorunmeyecek ancak Rip devre disi kaldiginda calismaya baslayacaktir.

Ve Rip tekrar aktif olarak calismaya baslarsa devre disi kalacaktir. WAN baglantisinin surekli up olmasini isteyen musterile icin ideal cozumdur. Alternatifi olarak ornegin ISDN baglantilari Floating Static Route ile backup icin onerilir.



Ornegimiz de iki nokta arasinda 1,5 Mbitlik bir baglanti var ve bu baglanti Rip ile konfihure edilmis. Ayni iki nokta arasinda dila-up bir baglanti var bu da Floating Static Route ile konfigure edilmisve Administrative Distance icin Ripinkinden daha buyuk olan “130” secilmis.

(Burada ki bri 0/1 portu ISDN baglantilari icin kullanilan porttur, CCNA 4 icerisinde detayli olarak anlatilacaktir.)

Dolayisiyla Rip ile calisan hat down oldugunda dial-up baglanti devreye girecek ve hat tekrar ajtif oldugunda devreden cikacaktir.

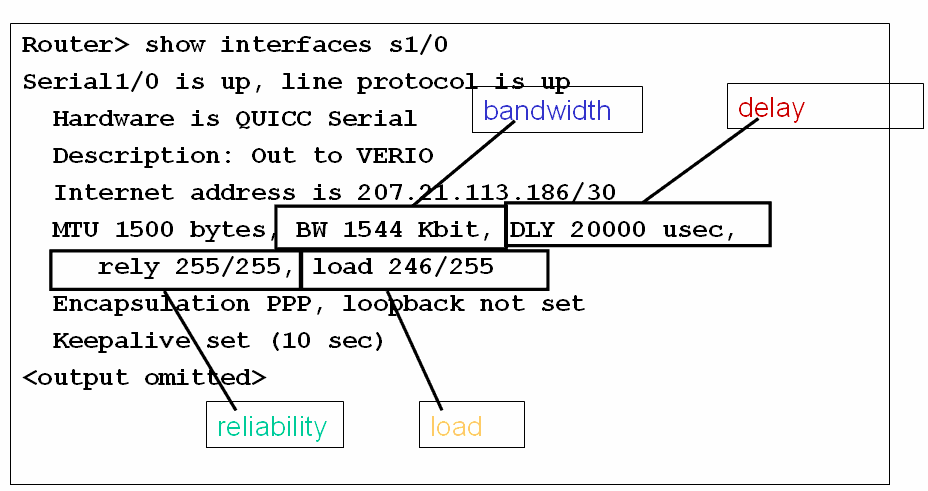
# IGRP (Interior Gateway Routing Protocol)

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: Guvenilirlik

K5: MTU (Maximum Transmission Unit)



Burada buyuk cogunlukla etki eden deger bant genisligi degeridir. Routerlar bant seri interface’lerindeki genisliklerini anlayamazlar bu yuzden bizim verdigimiz yada default olan degerleri kullanirlar. 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 anlamli olmasi icin gercek bant genisligi interfacelere atanmalidir. Bunun icin “bantwith *bantgenisligi(kbit)*” komutu kullanilir.



# IGRP AD ve Timers

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

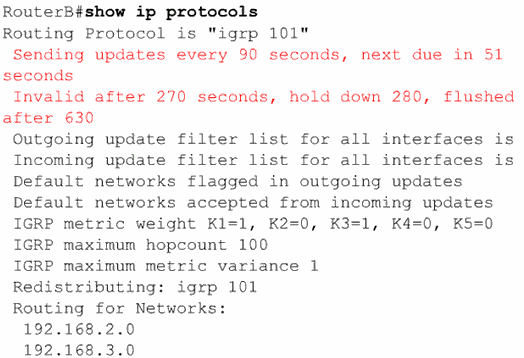
IGRP default olarak 90 saniyede Routing Table’ ini komsu Routerlarina 255.255.255.255 broadcast adresi uzerinden update eder.

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

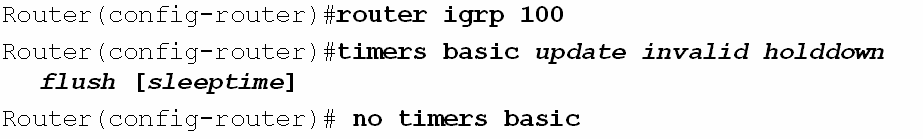
Daha buyuk metrige sahip updateleri ancak Holddown Timer suresinin sonunda kabul eder ki bu sure 280 saniyedir. Artik bu noktadan sonra IGRP ile kofigure edilmis Router kaybettigi network bilgisini silmese bile daha buyuk metric ile gelebilecek updateleri kabul edecektir.

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

“show ip protocols” komutu ile bu sureler goruntulenebilir.



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.



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

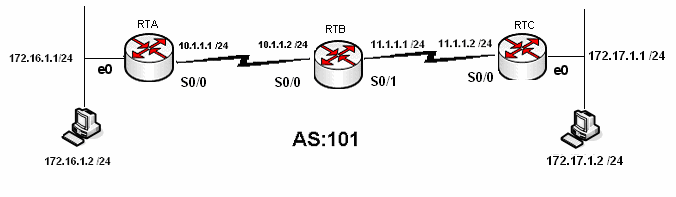
Burada Router variance ile belirtilmis sayisi alip en kucuk metric degeri ile carpar ve o degerin altinda metrige sahip yollar arasinda load balancing yapar.

# IGRP Konfigurasyonu

IGRP’ de tıpkı Rip gibi classfull bir routing protokoldür.

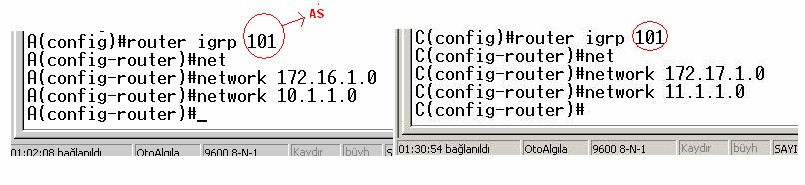
IGRP konfigürasyonu Rip’ in ki ile büyük ölçüde aynıdır. Burada tek fark aynı sistemde çalıştığımızı belirtmek için kullanacağımız Autonomus System numarasıdır. Kısaca AS denebilir. Bütün Routerlarda aynı AS kullanılmaz ise routerlar arasında iletişim olmaz.

Ornek bir senaryo ile konfigürasyonu yapmak gerekirse;



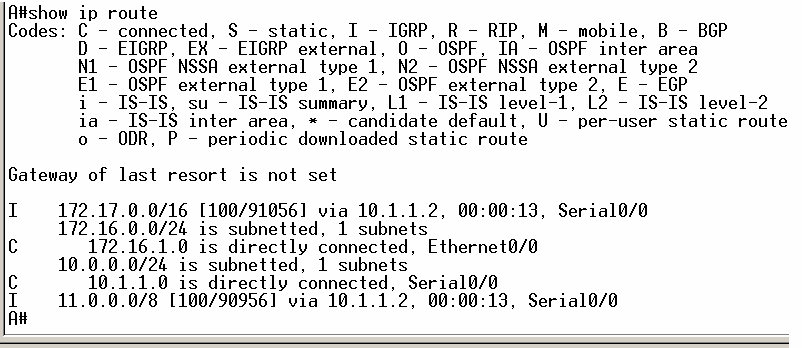
Konfigürasyon yapılırken DCE ve DTE uçlar düzgün belirlenmeli ve gereken yerlere “clock rate”

verilmelidir.



(A ve C router’ larının konfigürasyonu)

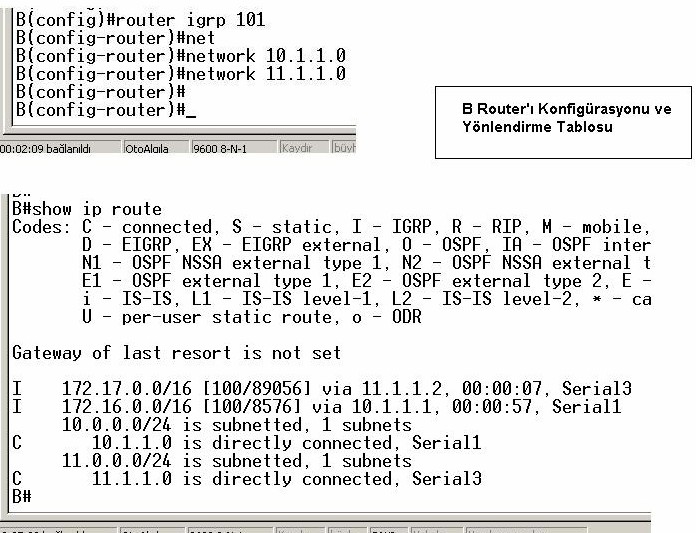
Routing Table incelendiğin administatice distinct’ in 100 olduğu görünecektir. Yine dikkat edilirse metric değerlerinin Rip’ inkinden çok faklı ve yüksek değerlerde olduğu gözden kaçmaz. İşte bu metrik değerleri bahsettiğimiz K1’ den K5’ e kadar değerlerle hesaplanmıştır.



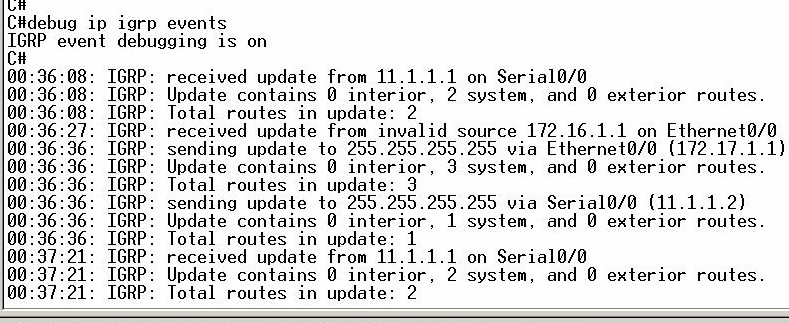
(A router’ ı Yönlendirme Tablosu)



(C Router’ı yönlendirme tablosu)



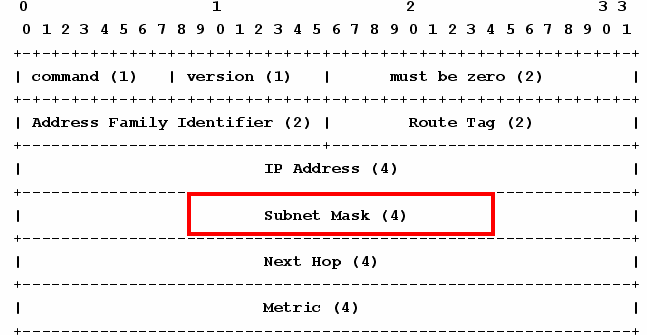
IGRP’ de tıpkı Rip gibi updatelerini 255.255.255.255 broadcast adresinden yapar.



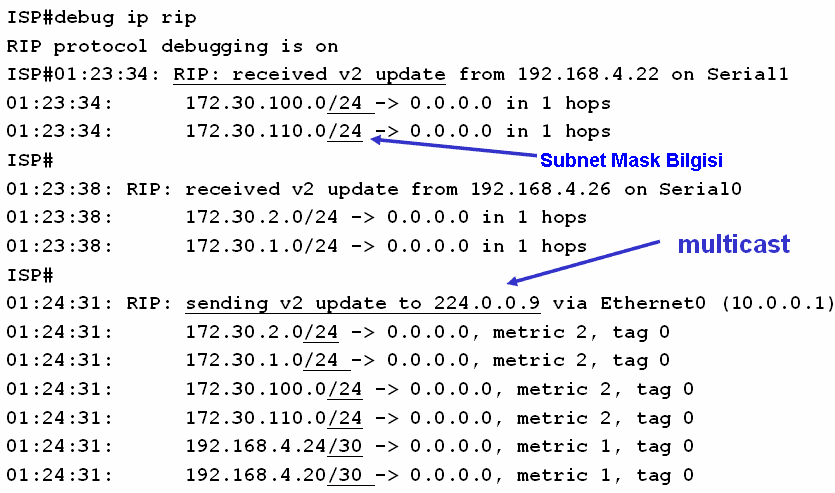
# RIPv2

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

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

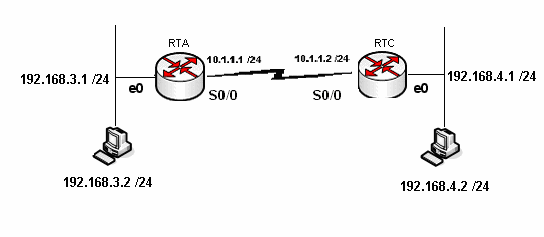


Burada ki subnet mask bilgisi protokolun Classless calismasini saglar.

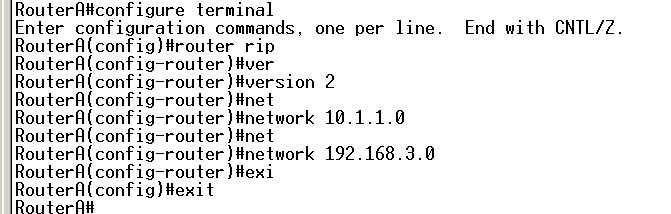


Konfigürasyonda ise tek küçük fark “version 2” komutunun verilecek olmasıdır. Bu komut router konfigürasyon alt moduna geçildiğinde verilmelidir.

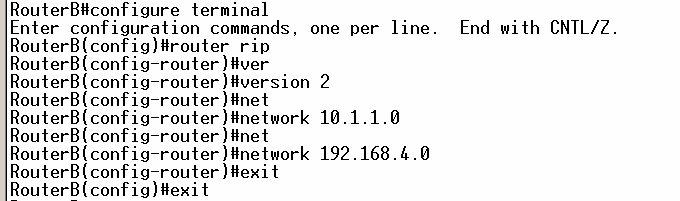
Rip protokolü sırasında üzerinde çalıştığımız senaryoyu burada da uyguladığımız da arada ki farkları daha iyi anlayacağız.



Bu topoloji de ip adreslerimizi ilgili interface’ lere atadıktan sonra konfigürasyon Ripv2 için şu şekilde olacaktır.



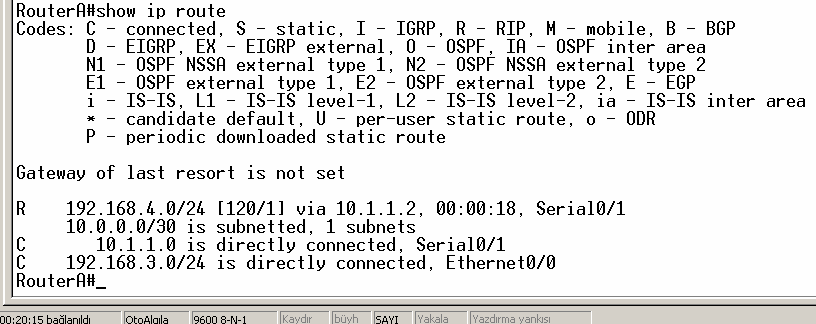
(Router A için Konfigürasyon)



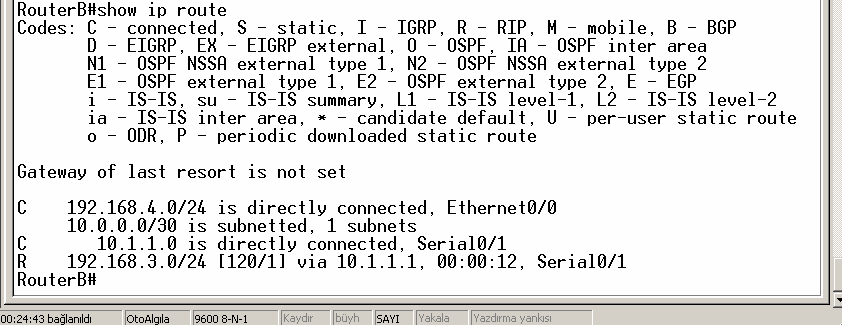
(Router B için Konfigürasyon)

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.

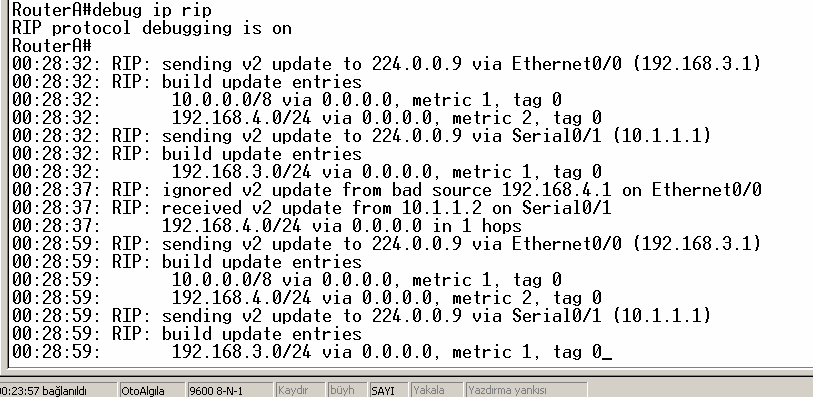


(RouterA için Routing Table)



(RouterB için Routing Table)

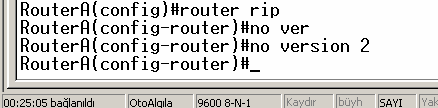
Routign Table’ lar dikkatle incelendiğinde uzak networklere giderken kullanılacak yollar metric ifadeleriyle birlikte görüntülenebiliyor. Ripv2’ de tıpkı Ripv1 gibi metric hesabında hop sayısını kullandığı için buradaki metricler aynı zaman da hop sayısınıa eşittir.

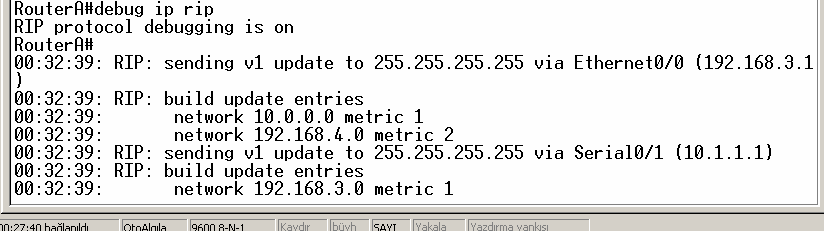


(RouterA için debug)

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

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:

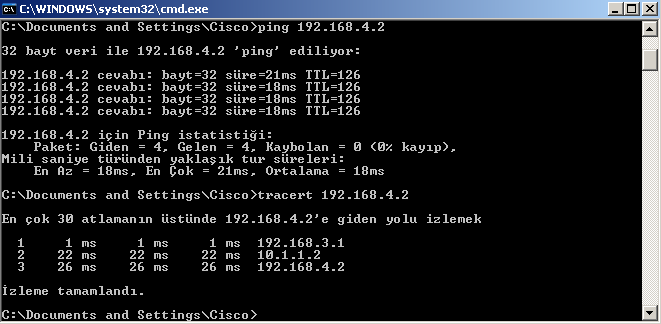




(RouterA için dedbug)

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

Networkler arasıdan ki iletişimi komut satırında kullanabileceğimiz “tracert” komutu ile inceleyebiliriz.



Ben bu komutu kullanırken 192.168.3.2 ip adresine sahip bilgisayarı kullandım. 1 adımda ping paketim varsayılan ağ geçidi olarak tanımladığım RouterA’ nin Ethernet interface’ ine, ikinci adım de 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

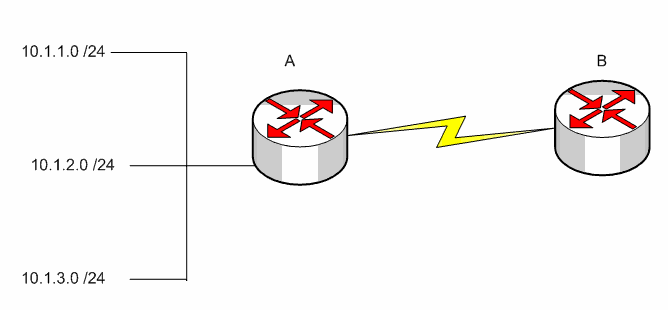
line vty 0 4

!

end

# RipV2 Auto Summary

Ripv2’ nin Auto Summarization ozelligi vardir ve default olarak acik durumdadir.



Ornegin sekildeki yapi icerisinde Ripv2 ile konfigure edilmis A routeri yine Ripv2 ile konfigure edilmis B routerina 10.0.0.0 netrworkunu update edecektir.

Bu calisma mantigi icerisinde default olarak acik olan auto summarization ozelligi kapatilmadigi takdir de Ripv2 ninde sanki Classfull mus gibi calsitigi soylenir.

Auto Summarizxation ozelligi “no auto-summary” komutu ile kaldirilabilir. 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

Konfigurayonun bu hali ile artik A routeri summary update yerine butun networkleri update edecektir ve B Routeri Routing Table’ inde butun networkler yer alacaktir.

# Extra

Split Horizon kuralinin enable olmadigi durumlarda ripv2 ile konfigure edilmis ve update edilecek networklerin uzerine yazacak ve interface’ e uygulanacak “ip summary-address” komutu kullanilabilir.

int s1

ip address 10.1.1.1 255.255.255.0

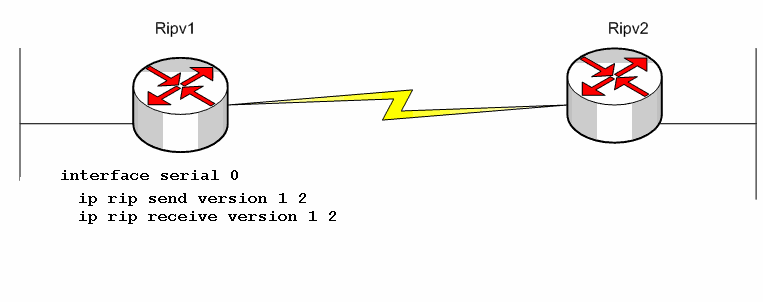
ip summary-address rip 10.2.0.0 255.255.0.0 no ip split-horizon

router rip network 10.0.0.0

Ornegin bu uygulamada 10.2.0.0 update’ i rip tarafindan ozetlenen 10.0.0.0 update’ inin uzerine yazacaktir.

# Ripv1 ve RipV2 Haberlesmesi

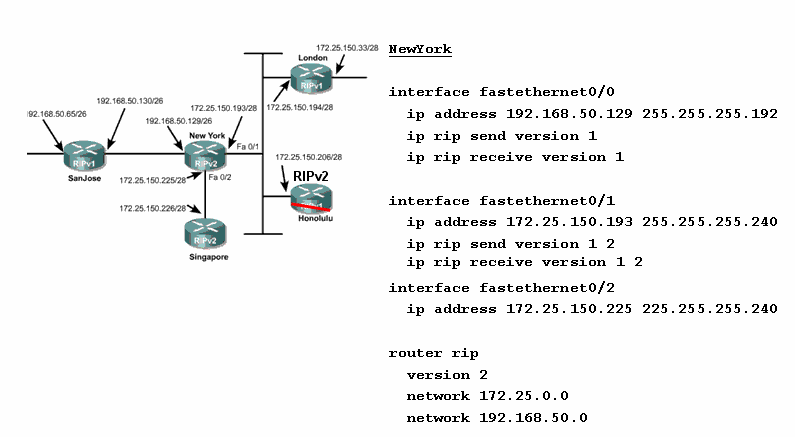
Ripv1 ve Ripv2 konfgirasyonlar sistem de bulnan routerlar arasinda sadece Ripv1, Ripv2 paketlerini alip gondermek yada her ikisinde alip gondermek uzere konfigure edilebilir.



Burada istenirse farkli uygulamalar yapilabilir. Ornegin;

ip rip receive version 1

komut satiri ile soz konusu Routerin sadece Version 1 updatelerini almasi saglanabilir.



CNAP Slaytlarindan alinan bu sekilde durum daha iyi anlasilacaktir. Burada Newyork Router i Ripv2 ile konfigure edilmis. Ve interfacelerine sirasiyla su sekilde konfigure edilmis diger routerlar bagli;

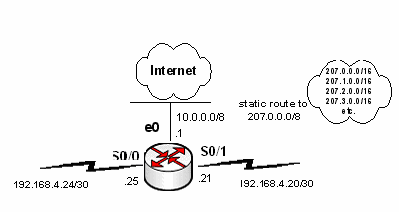
Fa0/0: Ripv1 Fa0/2:Ripv2 Fa0/1:Ripv1 ve Ripv2

Bu durumda Fa0/0 interface ine hem London Routerindan aldigi v1 updatelerini gondermeli hemde ondan v1 updatelerini almalidir.

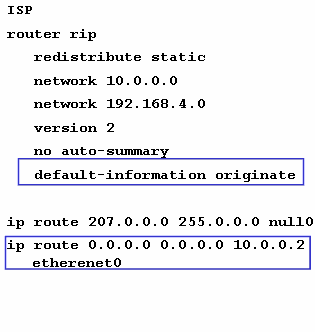
Fa0/1 interface inde ise konfigurasyona bakildiginda hem v1 hem de v2 updatelerini gondermek uzere hem de almak uzere konfigure edildigi anlasilmistir. Cunki bu interface’ e hem Ripv1 ile hem de Ripv2 ile konfigure edilmis Routerlar baglanmistir. (Multiaccess)

Fa0/2 icin zaten ozel bir konfigurasyona gerek yoktur.

# Ripv2 ve Default Routing



ISP Routeri uzerinde Default Route tanimlasinmasi asagidaki gibi olacaktir.



# Ripv2 Authentication

Ripv2 konusan Routerlarin updateleri sirasinda authentication saglanabilir. Bunun icin Global Konfigurasyon modunda “key” komutu kullanilmalidir.

Router(config)#key chain Ozcan Router(config-keychain)#key 1

Router(config-keychain-key)#key-string Yildiz

Authentication saglanacak Routerlar icin password ayni olmalidir ancak key adi degisitirilebilir. Key olusturulduktan sonra interface’e uygulanmalidir.

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 kullanimi opsiyoneldir. Authentication bilgilerinin encrypted halde gonderilmesini saglayan bu komut kullanilmadiginda da interface default olarak text halinde authentication bilgilerini gonderecektir.

# 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 çıkan trafiği kontrol edecek şekilde uygulanabilir. Eğer herhangi bir interface’e bir access list atanmışsa router bu interface’den gelen her paketi alıp inceleyecek ve access list’te belirtilen işlevi yerine getirecektir. Yani ya o paketi uygun yöne iletecek ya da paketi yönlendirmeden yok edecektir.

* + - * + Access List’ lerde kriteler satır satır belirtilmiştir. Gelen isteklerin kriterlere uyup uymadıkları sırayla belirlenir.
        + İlk eşleşen kriterin bulunduğu satıra gelindiğinde 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
4. başlık dememize rağmen aslında iki başlık gibi düşünülmelidir. Çünkü Named Accces Listler hem standart hem de Extended olarak kullanılabilirler.

Access Listler arasıdan 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 acess list |

Access List’ ler oluşturulurken dikkat edilmesi gerekenler şunlardır;

* + Oluşturulduktan sonra mutlaka bir interface ile ilişkilendirilmedlidir aksi takdir de aktif olmayacaktır.
  + Kriterler satır satır uygulanacağı için listeler oluşturulurken en belirgin kriterden en genel kritere doğru yukarıdan başlayarak organize edilmelidir.
  + Listeden 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 oluştuğunu ve bu satırında diğer satırlardaki herhangi bir kritere uymayan istekleri yok ettiğini söylemiştik. Dolayısıyla mutlaka ve mutlaka bir Access List grubunda “permit” aksiyonu olmalıdır.
* Access Listler sadece Router üzerinden giden veya gelen trafiği düzünlemek için kullanılabilirler. Router’ ın sebep olduğu trafik için kullanılamazlar.
* Access Listler’ den satır çıkaramasanız ve satır eklediğinizde de o satır en son satır olarak yerini alır. Dolayısıyla kritelerinizi 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 edlien 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 kümesi için geçerlidir.

Router(config)#acces-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 ilişiklendirilir. Burada ki in ve out komutlara isteğe göre içeriden dışarıya (in) ve dışarıdan içeriye (out) olan trafiği kısıtlamak için kullanılır.

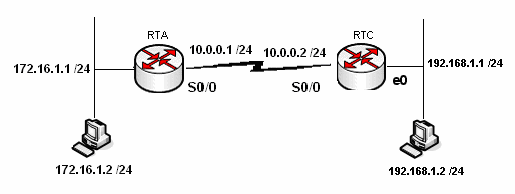
Örneğin networkümüz de bulunan 192.168.1.100 ip adresine sahip bilgisayarın dışarıya çıkışını önlemek istersek komut satırında;

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

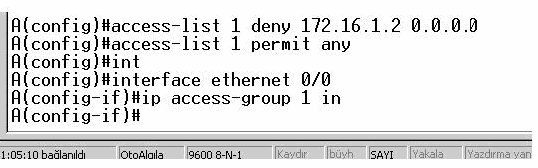
Yazmalıyız. Burada 1. satırda ilgili hosta “deny” uygulandı, 2.satırda diğer hostların “implicit deny all” kuralı ile yok edilmemeleri için kalan hostlara “permit uygulandı, 3 ve 4.satırlarda ise oluşturulan Access list Ethernet interface’ ile ilişkilendirildi. ”

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 interface2 inden dışarı çıkmasını engelleyelim fakat diğer bilgisayarlar bundan hiçbir şekilde etkilenmesin.

Bu durumda A Router’ ın şu konfigürasyon yapılmalıdır;



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 erişememsini, 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

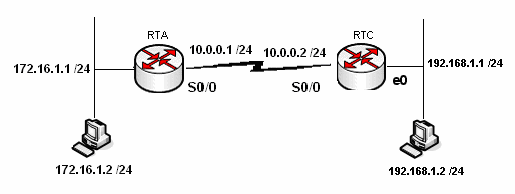
Router(config)#access-list 101 permit tcp host 192.168.1.100 host 212.1.1.8 eq 25 Router(config)#access-list 101 permit ip any any

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 erişememsini fakat 25. porttan erişmesini sağlamış oluyoruz. 3. satır ile de diğer bilgisayarların “implicit deny all” kuralı ile yok edilmelerini önlemiş olduk.

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

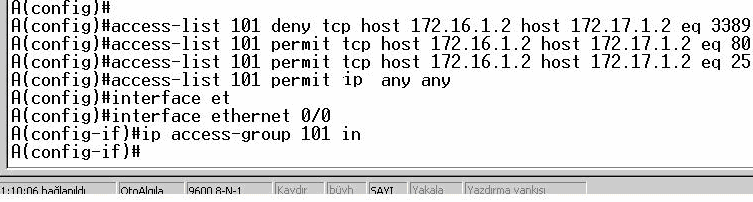
Bir senarayo üzerinde çalışmak gerekirse;

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



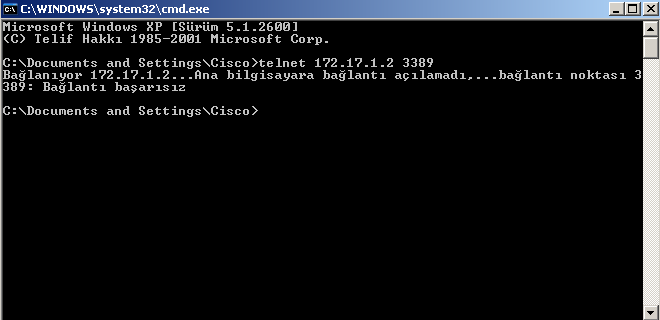
1. 172.16.1.2 bilgisayarı 172.17.1.2 bilgisayarına 3389. porttan erişemesin.
2. 172.16.1.2 bilgisayarı 172.17.1.2 bilgisayarına 80. porttan erişebilsin.
3. 172.16.1.2 bilgisayarı 172.17.1.2 bilgisayarına 80. porttan erişebilsin.
4. 172.16.1.0 networkünde bulunan diğer bilgisayarlar uzak networkteki diğer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

Böyle bir durumda A Router’ ı üzerinde yapılacak konfigürasyon şu şekilde yapılmalıdır:

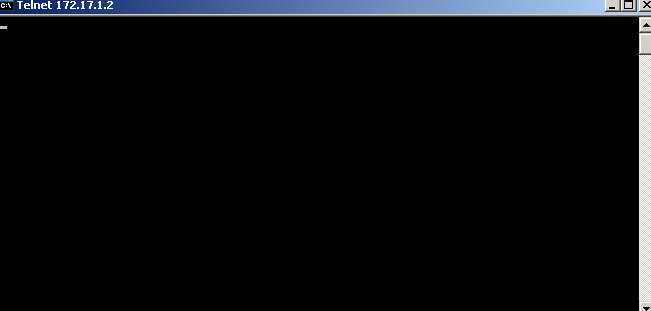


Senaryo için belirlediğimiz istekleri satır satır konfigüre ettik. Access List’ in 4. satırındaki komut ile kalan bilgisayarların çıkmasına izin verilirken 1,2 ve 3. satırlarda 172.16.1.2 bilgisayarının uzak networkte ki 172.17.1.2 bilgisayarına doğru olan trafiğinde çeşitli kısıtlamalar ve izinler uygulandı. 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.



(172.17.1.2 uzak bilgisayarına 3389. porttan bağlanılamıyor.)



(172.17.1.2 bilgisayarına 80. porttan Telnet ile bağlanılması)

# Named Access Listler

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

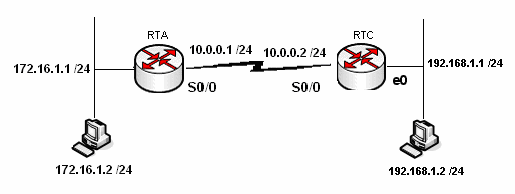
Extended Access List’ te üzerinde çalıştığımız aynı senaryoyu Named Access List ile konfigüre etmek istersek komut satırına;

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

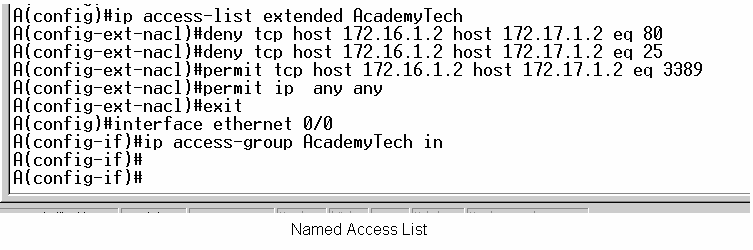
Router(config-ext-nacl)# permit tcp host 192.168.1.100 host 212.1.1.8 eq 25 Router(config-ext-nacl)# permit tcp any any

Yazmamız yeterli olacaktır. Burada 1.satırda belirtilen AcademyTech bizim belirleyeceğimiz bir isimdir 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 tanımlanmalı;

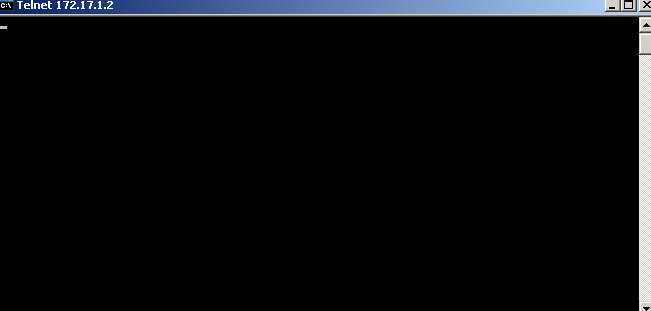


Burada Access List’ in1. satırında extenden Acces List kullanılacağı ve bu Access List’ in isminin AcademyTech olacağı belirtildi, 2. ve 3. satırları ile uzak networkteki 172.17.1.2 bilgisayarına 80 ve 25. portlardan erişilmesi, 172.16.1.2 bilgisayarı için yasaklanmış oldu.

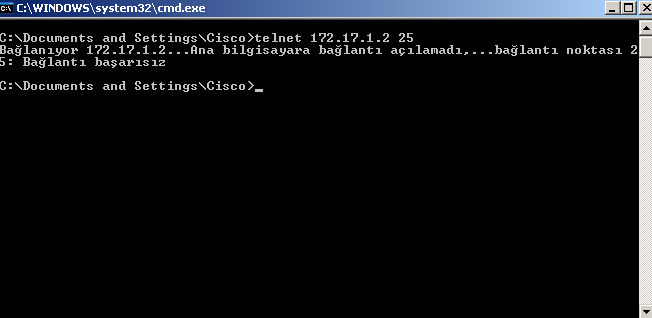
1. ve 5. satırlarda ise gerekli izinler verildi. 4. satırdaki komut örnek olması için komut satırına yerleştirildi. Normal şartlarda bu satır kullanılmayabilir, çünkü 5. satırda ki ifade ile zaten

172.16.1.2 bilgisayarı da diğer izinleri elde ediyor.

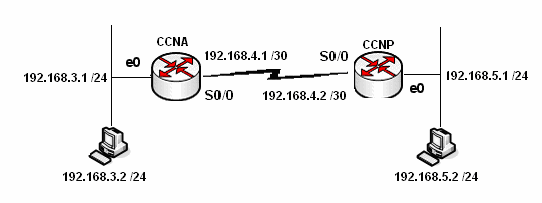
Telnet ile Acces List’ leri test edersek;



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



(25. porttan bağlanma denemsi başarısız.)



**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**

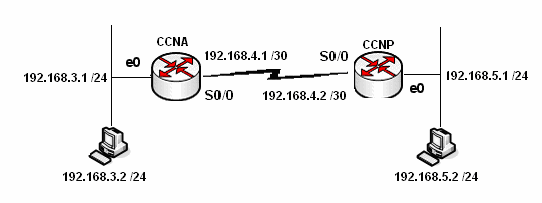
CCNA(config-if)#**ip access-group 1 in**

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**



**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**

**HostA da bulunan FTP Server ve Web Server’a 192.168.5.2 bilgisayarinin erismesini engellemek. (Kalan trafik akisi 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**

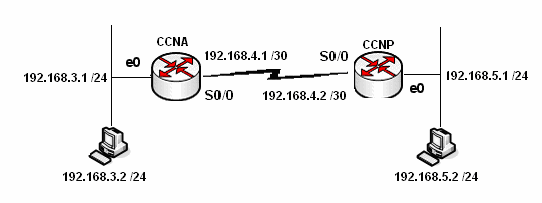
CCNP(config)#**access-list 101 permit ip any any**

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 trafik 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**

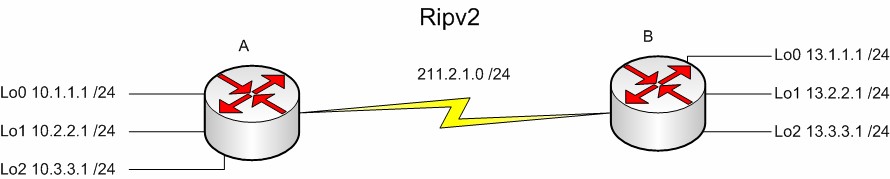
CCNA(config-if)#**ip access-group Yildiz in**

CCNA(config-if)#

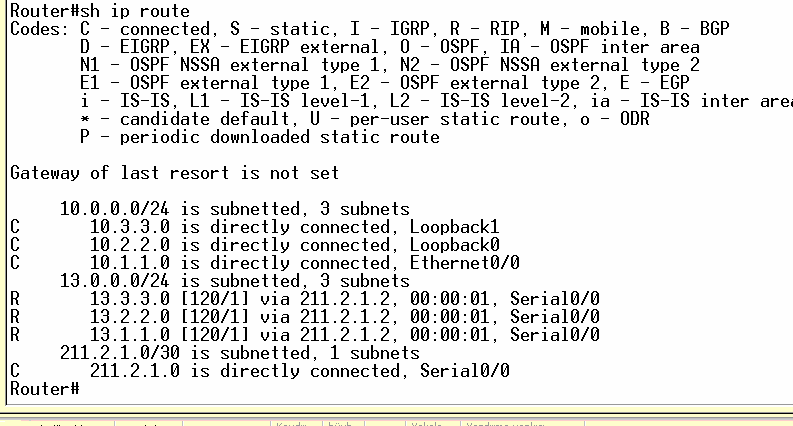
# Access Lists ve Distribute List

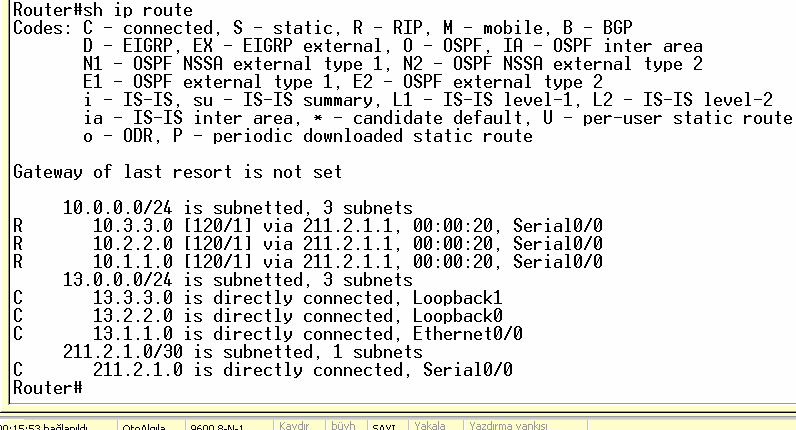
Routing protokoller ile calisirken bazi networklerin update edilmemesini isteyebiliriz. Bunun icin passive interface komutu bir cozumdur ancak burada o interface’ den hic bir update yapilmayacaktir. Oysa Access Listler ile birlikte olusturulacak Distribute List’ ler ile hangi networklerin update edilecegine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

Durumu ornek calisma ile ozetleyecegim.



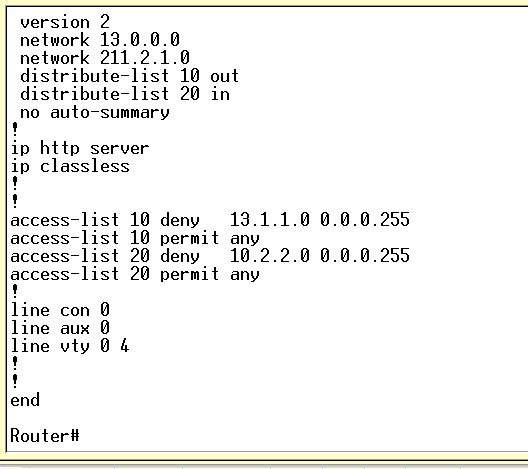
Ornek topolojide her router icin 3’er adet loopback interface olusturdum ve konfigurasyon icinde bu loopbacklaride Ripv2 icerisinde tanittim. Baslangicta Routing Table’ lar A ve B icin sirasiyla su sekilde olustu.





Her iki Router da Loopback adresler Directly Connected ve Ripv2 ile update edilmis olarak gorunmekteydi.

A routerinda iki adet acces list yazdim ve bunlar Ripv2 konfigurasyonuna Distribute list komutu ile bagladim.

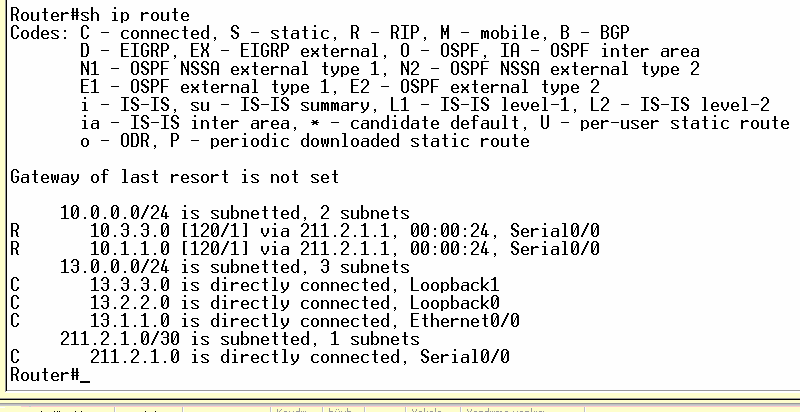
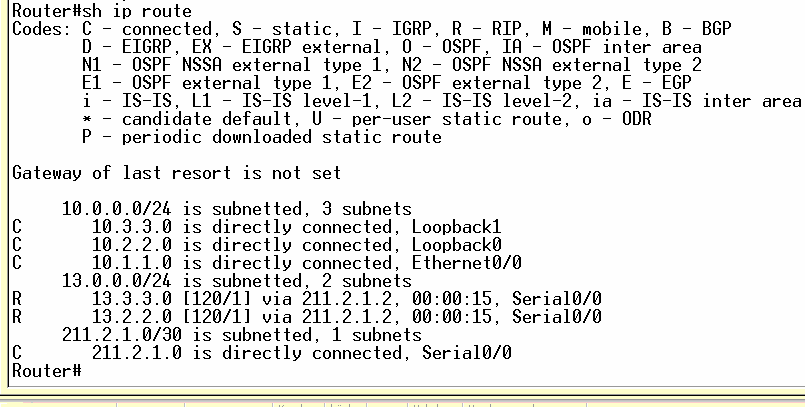


10 numarali access list ile 13.1.1.0 networkunun, 20 numarali access list ile 10.2.2.0 networkunu yasaklamak icin gereken satirlari yazdiktan sonra “in” ve “out” olarak Rip’e uyguladim.

Yazilan satirlarin tam turkcesi su sekildedir: 10.2.2.0 networkunu iceriden disariya gonderme,

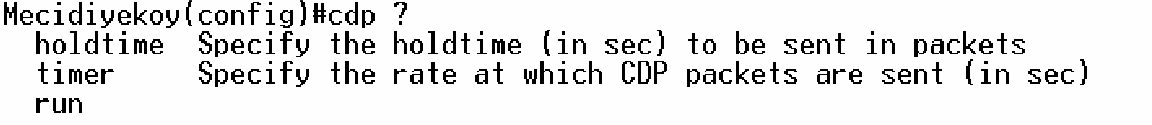
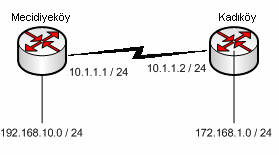
* + - 1. netwrokune ait update’ i disaridan iceriye alma.

Bu durumda Routing Table’lar su sekillerde degisti.



# CDP (Cisco Discovery Protocol )

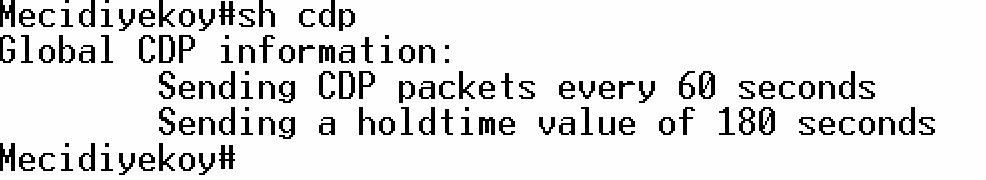
CDP ( Cisco Discovery Protocol ) Cisco tarafından geliştirilmiş bir protokoldür. CDP ortamdaki Cisco cihazları hakkında bilgi almak için kullanılmaktadır. IOS 10.3 ve üzerinde CDP default olarak çalışmaktadır. Aşağıdaki örnekte routing işlemi gerçekleştirilmiş, çalışan bir sistem bulunmaktadır.



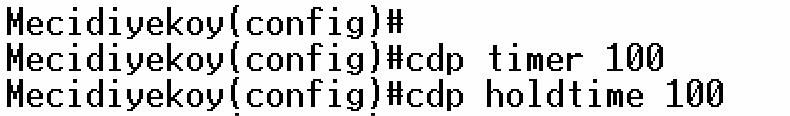
Mecidiyeköy router’ ında cdp yazdıktan sonra soru işareti kullandık. Eğer IOS versiyonumuz 10.3 den düşük ise “cdp run” komutunu kullanarak CDP protokolünü çalıştırabiliriz.



CDP ile ilgili özellikleri görüntülemek için “sh cdp” komutu kullanılır.

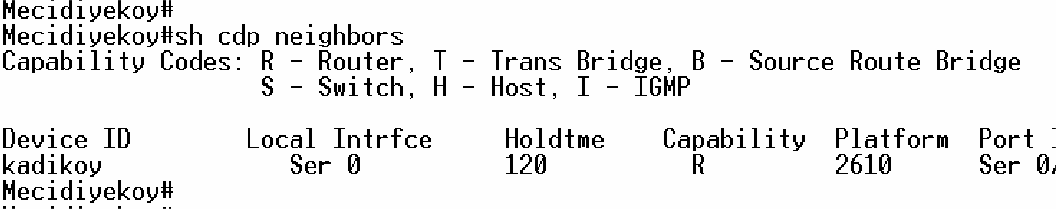


Görüldüğü gibi 60 saniyede bir CDP paketleri gönderilir. Bu paketler sayesinde ortamdaki Cisco cihazları hakkında bilgi toplanır. Eğer 180 saniye haber alınamaz ise cihaz database’ den silinir. Bunlar default değerlerdir. Değiştirmek için aşağıdaki komutlar kullanılır.



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

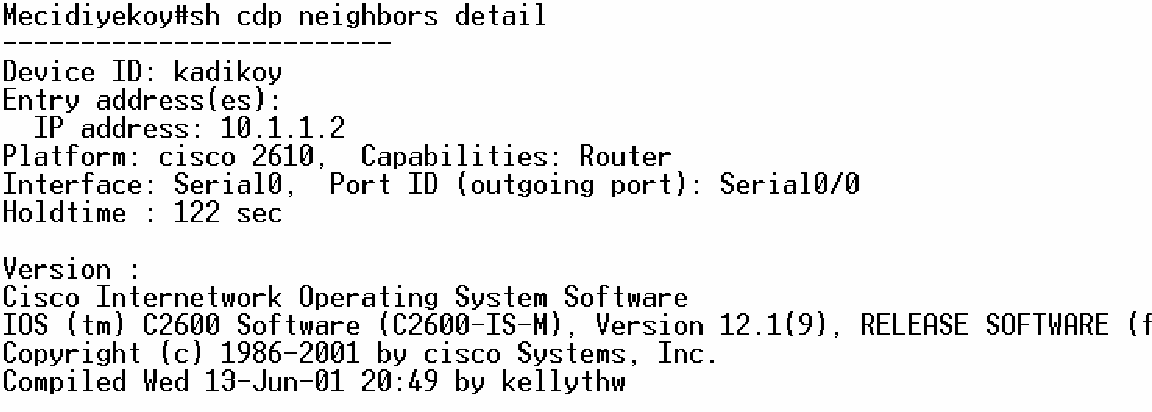
CDP kullanılarak öğrenilen bilgiler “show cdp neighbors” komutu ile görülebilir.



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.

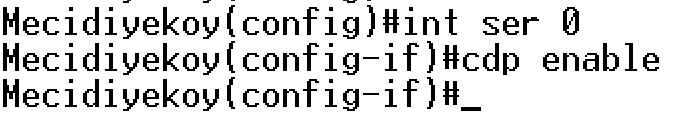
Daha detaylı bilgi alabilmek için “sh cdp neighbor detail” komutu kullanılır.



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



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



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



# EIGRP(Enhanced Interrior 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 routerlardan 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 dâhil 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 uzerinden gonderir. T1 ve uzeri bant genisliklerinde 5 saniye de bir gonderilen bu paketler T1 den daha dusuk bant genisliklerinde 60 saniyede bir gonderili. (Hold Time=3 X hello interval)

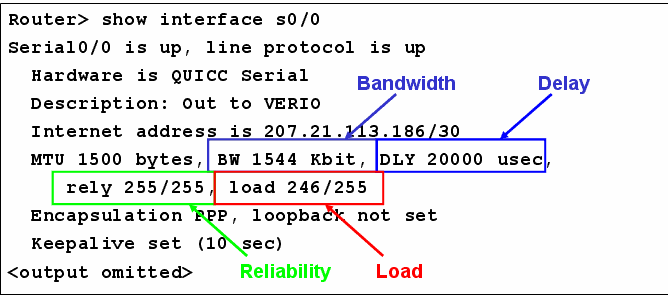
Acknowledgement paketleri data icermeyen paketlerdir ve guvenli iletisimi saglar. Hello paketlerinin multicast olmasina karsin Acknowledgement paketleri unicast calisirlar.

Update paketleri sistemdeki bir router yeni bir network buldugunda ya da kaybettiginde, metric hesabinda bir degisiklik oldugunda ve Successor degistiginde gonderilir. Bu aksiyonlardan biri gerceklestiginde EIGRP konusan bir Router butun komsularini multucastupdate gonderir.

Query paketleri bi router herhangi bir sekilde yeni, ozel bir bilgiye ihtiyac duyuldugunda gonderilir. Sozgelimi Successor’ i down olan ve Feasible Succesr’ i bulunmayan bir router Query paketleri gonderir ve cevaplar Reply paketleri ile doner. Query paketleri multicast iken Reply paketleri unicasttir.

# EIGRP Metric Hesabi

EIGRP metric hesabinda K1 vw K3 degerlerini kullanir. (Banswidth ve Delay)



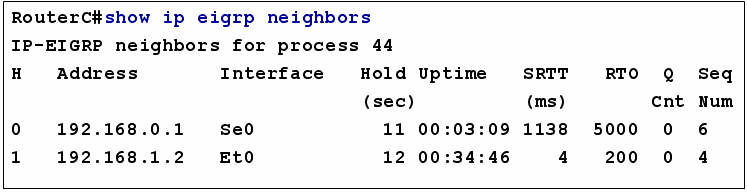
EIGRP ve IGRP bant genisliklerini ayni formul ile hesaplarlar.



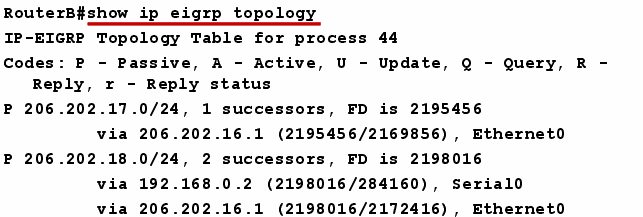
Fakat EIGRP icin K2, K4 ve K5 default olarak 0 sayilir.

# EIGRP Table’lari

EIGRP calisma mantigi icerisinde butun komsularini Neighbpr Table’ da ve hedef networke olan butun yollari da Topology Table’ da tutar. Bu bilgiler isiginda en iyi yol secimini yapar.

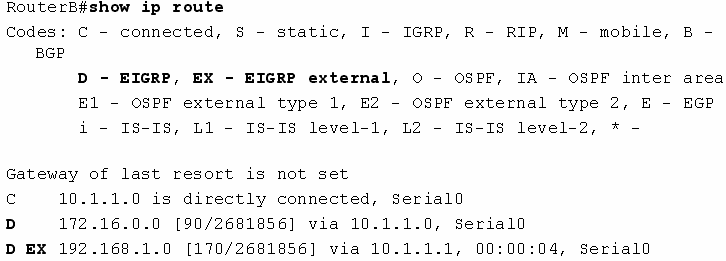


Neighbor Table da komsu routerlarin network katmani adresleri (ip adresleri), Q ile gosterilen ve sirada gonderilmeyen bekleyen paket sayisini ifade eden bir deger (ki bu deger 0 dan buyuk ise router da olasi bir problemden bahsedilebilir), SRTT ile gosterilen ve komsu routerlara gonderilen ve alinan paketler icin gecen ortalama sureyi gosteren bir deger ve Hold Time degeri bulunur.



EIGRP hedef networklere gitmek icin kullanacagi yollarin bilgisini ise Topology Table’ inda saklar. Bu table da bulunan bilgilere dayanara Successor ve Feasible Successor’ u secer.

Routing Table ise Successor (best route) olarak secilen yolun bulundugu yerdir.



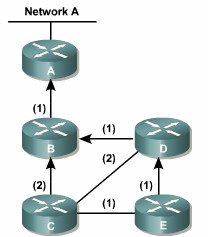
EIGRP harici bir protokolden gelen update bilgileri Routing Table’inda EX (external) olarak isaretler.

EIGRP topolojini oluştururken Dual Algoritmasını kullanır. Bu algoritma ile kendisine bir en iyi yol (Successor) bir de yedek sayılabilecek 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 anlaşılacağı gibi bir sonraki router için geçerli olan Feasible Distance’ dır.

Burada bir önemli kuralda, Feasible Successor seçilen yola ait Reported Distance değeri, Successor seçilen yolun 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 B | FD 3 | RD | Topoloji 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 olduğuna dikkat edin)

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

|  |  |  |  |
| --- | --- | --- | --- |
| Next Hop B | FD 2 | RD | Topoloji 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.)

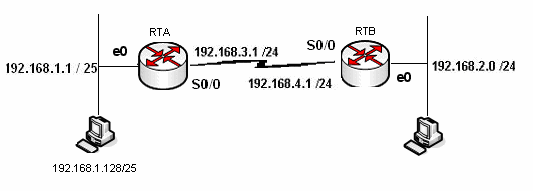
Not: EIGRP IPX ve AppleTalk networklerini de destekler ve bu networklere ait Neigbor, Topology ve Routing table’ lari ayiri ayri tutar.

# Auto Summarization

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

Sözgelimi 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 routerına Auto Summarization yapacak ve 10.1.1.0 / 24 networkü bilgisini update edecektir. Bu istenmeyen bir durum ise “no auto- Summarization” komutu ile özellik kaldırılabilir.

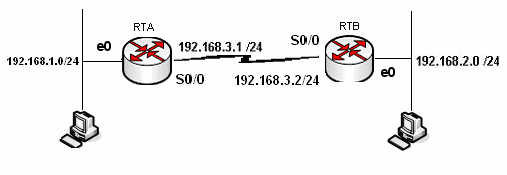


Auto Summarization ozelligi “no auto-summary” komutu ile kaldirilabilir.

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

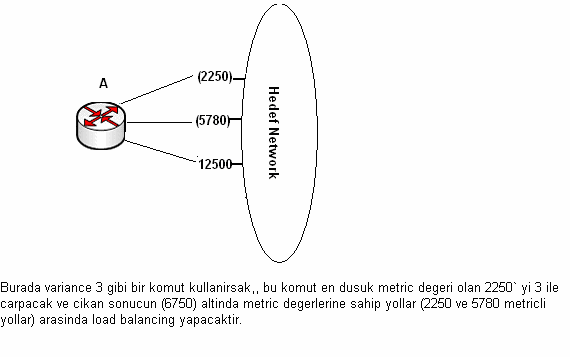
RouterB(config-router)#network 192.168.3.0 RouterB(config-router)#no auto-summary

# 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 ın load Balancing yapması ihtimaller arasındadır. Fakat EGRP’ 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.



Ornek Konfigurasyon; Router(config)#router eigrp 14

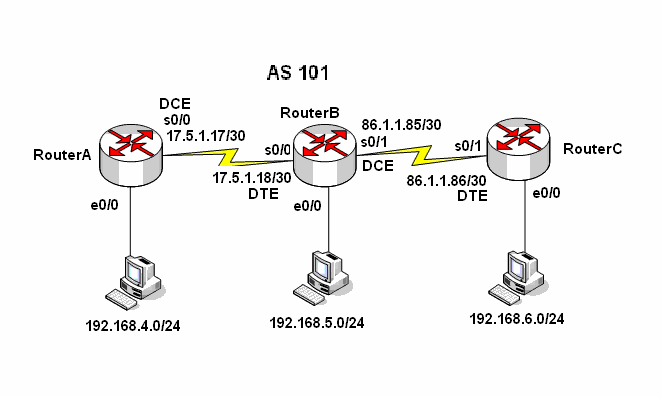
Router(config-router)#network 10.1.1.0

Router(config-router)#network 10.2.1.0

Router(config-router)#network 10.3.1.0

Router(config-router)#variance 2

# EIGRP Laboratuar Calismasi



Burada yapilan calismada AS olarak 101 secilmistir.

Laboratuar ortaminda clock uretimini saglayacak DCE kablolarin takildigi interfacelere uygulama icerisinde clock rate komutu verilmistir.

Auto Summarization ozelligi kapatilmistir.

Her bir Router, dan Runnin-config dosyalari. Routing Table’lari, Neighbor Table’lari ve Topology Table’ lari alinistir.

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

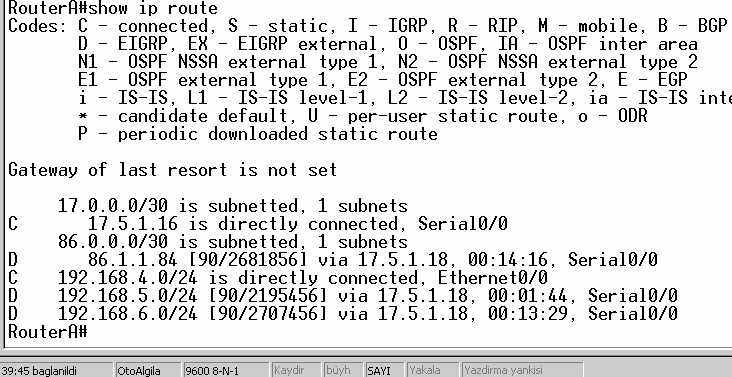
line aux 0

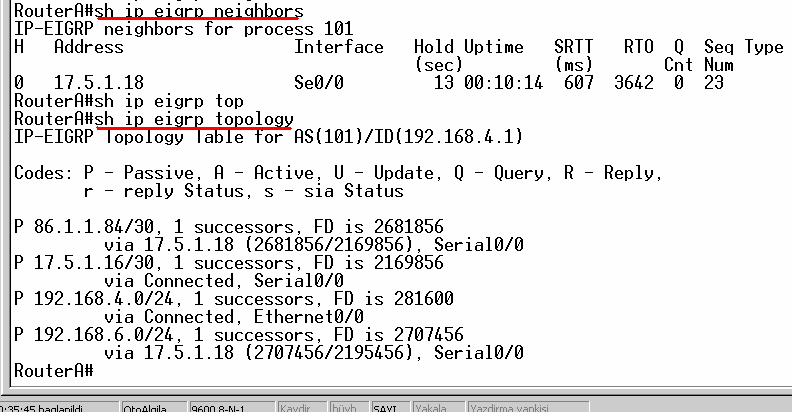
line vty 0 4

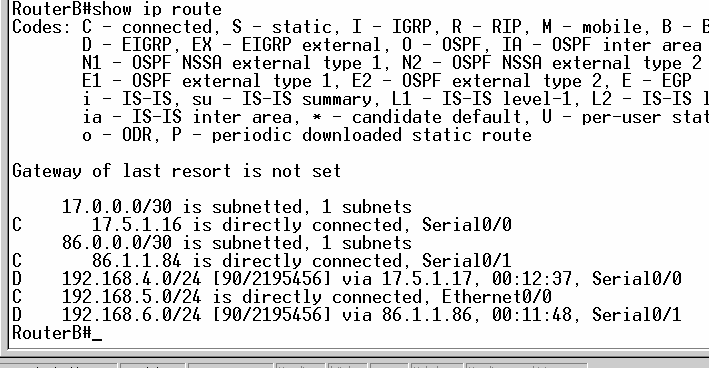
!

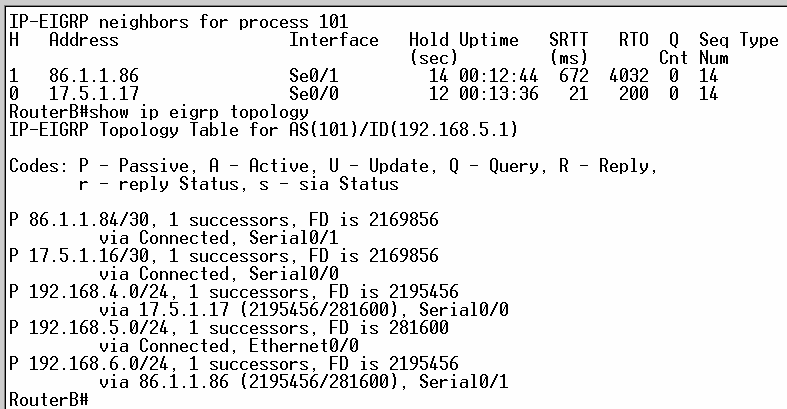
no scheduler allocate end

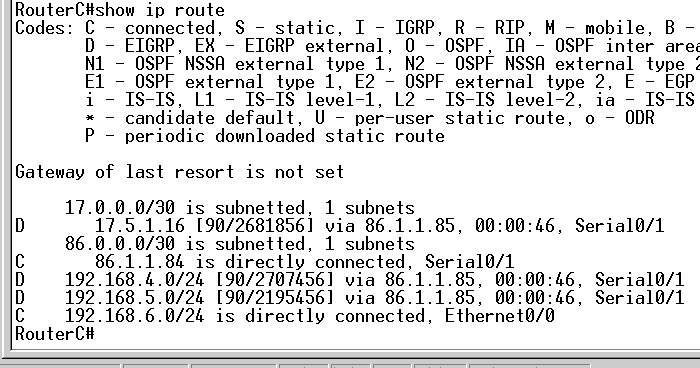
RouterC#

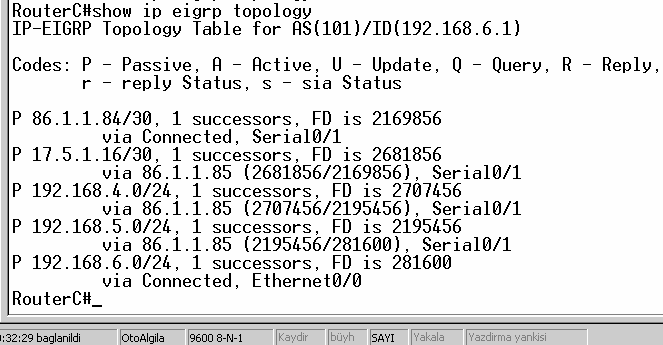
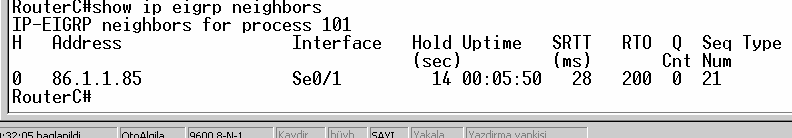




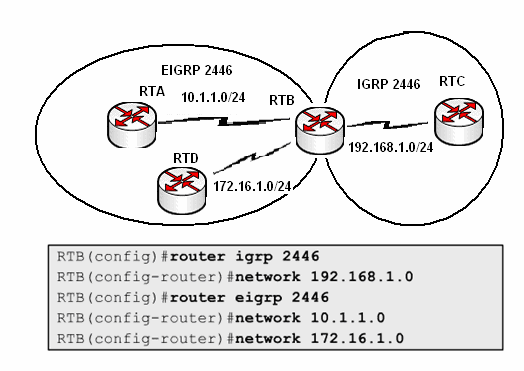




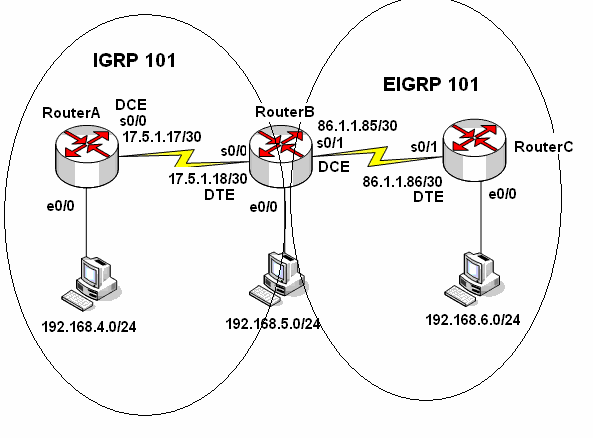




# EIGRP ve IGRP Birlikte Calismasi



IGRP ve EIGRP ayno AS icerisinde birbirleriyle haberlesirler. Burada ozel olarak dikkat edilecek tek nokta EIGRP konusan Routerlarin Routing Tebler’ larinda IGRP konusan Routerlara giden yollari External olarak etiketlemis olmasidir.



Hem IGRP hem de EIGRP icin AS numarasi 101 secilmistir.

Router B uzerinde hem IGRP hem EIGRO konfigurasyonlari yapilmistir.

Butun Routerlarin Routing Teble’ lari ve B routerinin running-config dosyasi incelenmek uzere alinmistir.

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

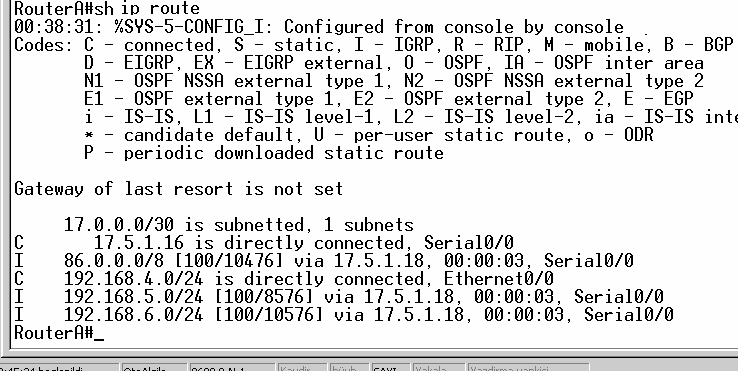
line aux 0

line vty 0 4

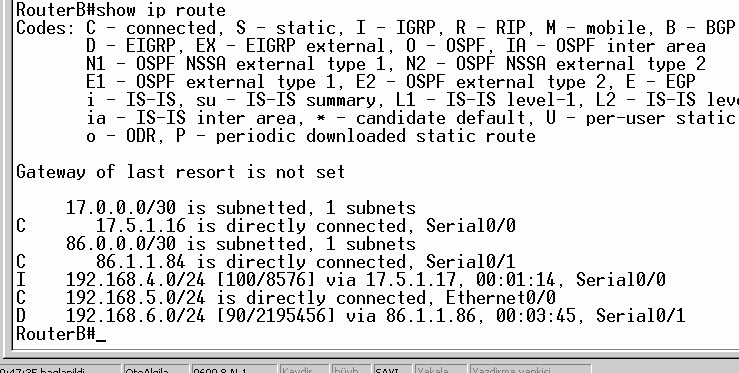
!

end RouterB#

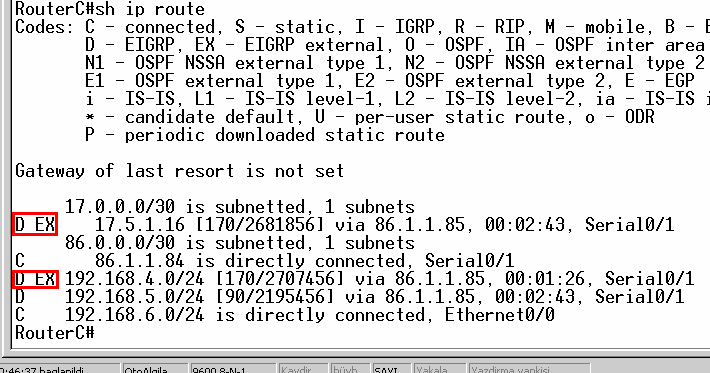
# RouterA Routing Table’i



**RouterB Routing Table’i**



**RouterC Routing Table’i**



**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 yükse 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 belirleyecegimiz birimim saniye olan bir degerdir. Burada dikkat edilmesi gereken bir konu ise birbirine bagla olan iki interface ‘ inde hello zaman araliginin esit olmasi gerektigidir. Aksi takdirde komsuluk iliskisi 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 Routerlarin birbirlerine gonderdikler 4 ayri paket sekli daha vardir. Bunlar;

**Type2:** DBD yani Database Descriptiin paketleri olarak bilinir ve Routerlarin Link durumlari hakkinda ozet bilgiler icerir.

**Type3:** LSR yani Link State Request paketleri olarak bilinir. Routerlar DBD paketleri ile ogrendikleri bilgilerin detayi icin diger Routerlara LSR paketleri gonderirler.

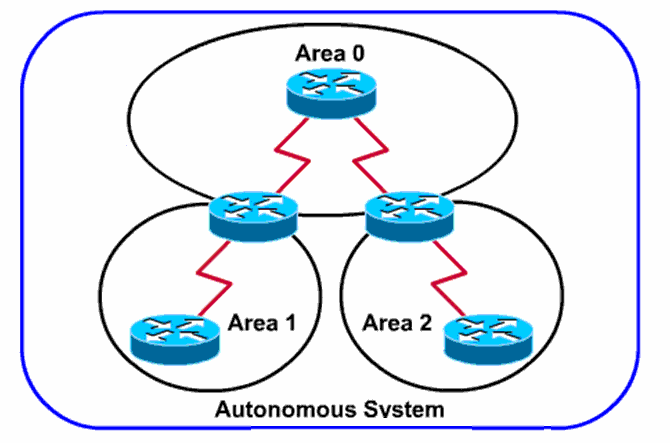
**Type4:** LSU yani Link State Update paketleri olarak bilinir. LSR ile istenen Link State Advertisements (LSAs)paketlerini tasir.

**Type5:** LSA yani Link State Acknowledgement paketleridir ve routerlar arasinda paketlerin alidigi onay bilgisini tasir.

# OSPF Area

Ospf calisma mantigi arealar uzerine kurulmustur ve bu sayede bir dizayn hiyerarsisi saglanabilmektedir. Bu hiyerarsik yapinin convergence; i hizlandirdigi da soylenebilir.

Ospf in merkezi area 0’ dir. Area 0 backbone area olarak adlandirilir ve farkli arealar oldugunda o arealar icinde area 0 ile konusan interface’ e sahip routerlar olmalidir.

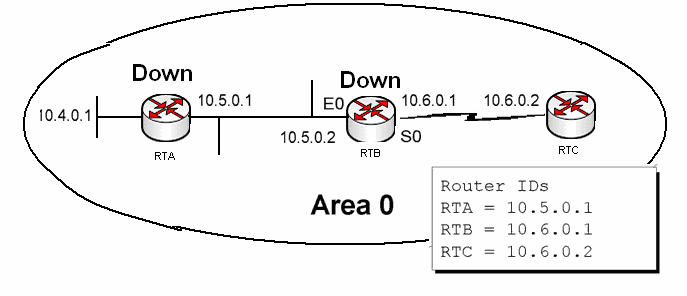


# OSPF Komşuluğu

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

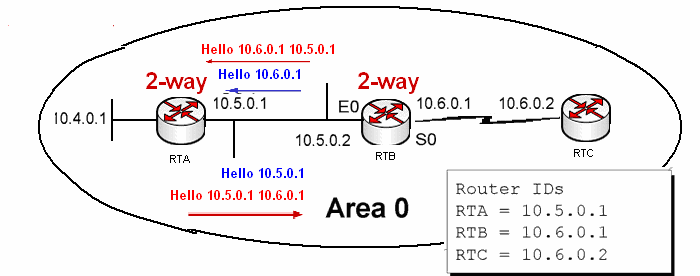
şunlardır;

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



**Init**, Diğer routerlardan cevap bekleme adımıdır.

**Two-Way,** Diğer routerların gönderdikleri Hello mesajlarının Komşu Router ID alanında kendi İP 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 iletişimi başlatacak routerı belirlemek için kullanılır, bu seçim herhangi birine bir üstünlük sağlamaz.

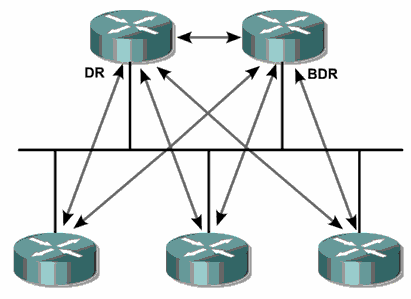
**Exchange**, Routerların bilgi alışverişi yaptıkları adımdır.

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

**Full**, Yönlendirme bilgilerinin senkron hale getirilmesi durumudur.

# DR ve BDR Secimi

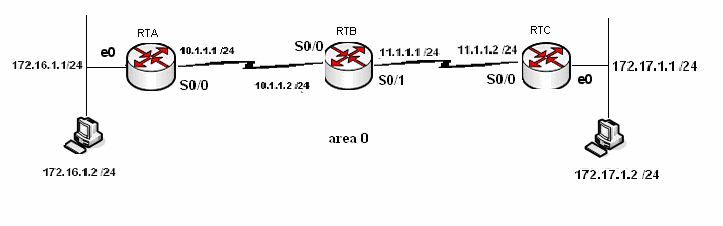
Multi-acces networklerde isler biraz daha farkli yurur. Bu networklerde Two Way halindeyken ortamda butun trafigi yonetecek bir router secilir ki buna Designated Router (DR) denir. Ve yine Bacup Designated Router (BDR) denen ve DR ‘ in yedegi olan bir router daha secilir.



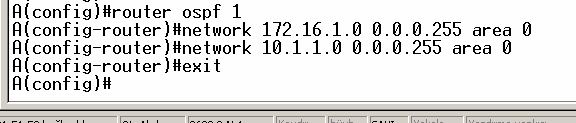
DR ve BDR secimleri Router ID’ ler ile yapilir. En yuksek Router ID’ ye sahip router DR ve ikinci en yuksek ID, ye sahip router BDR secilir.

Router ID bir routerin aktif olan interfacelerindeki en yuksek ip adresidir. Burada loopback adreslerin bir ayricaligi vardir. Eger bir Routerda loopback adresi tanimlanmissa o routerin ID’ si loopback ip’sidir. Ip adrsinin kucuk veya buyuk olmasi durumu degistirmez.

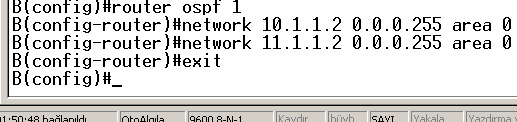
# Single Area OSPF Konfigürasyonu



Router A Konfigürasyonu;



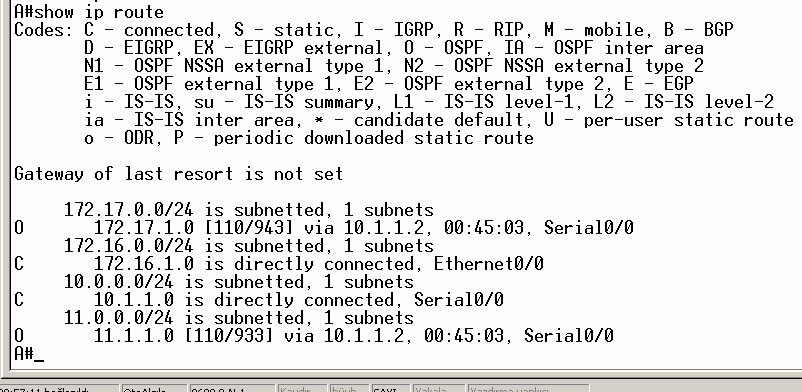
Router B Konfigürasyonu;



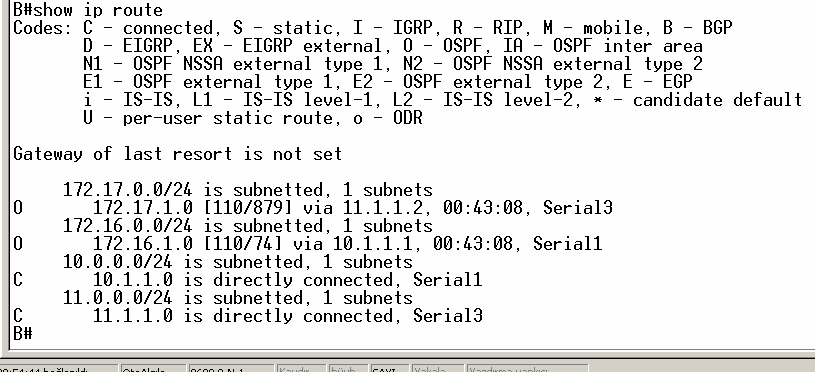
Router C Konfigürasyonu;



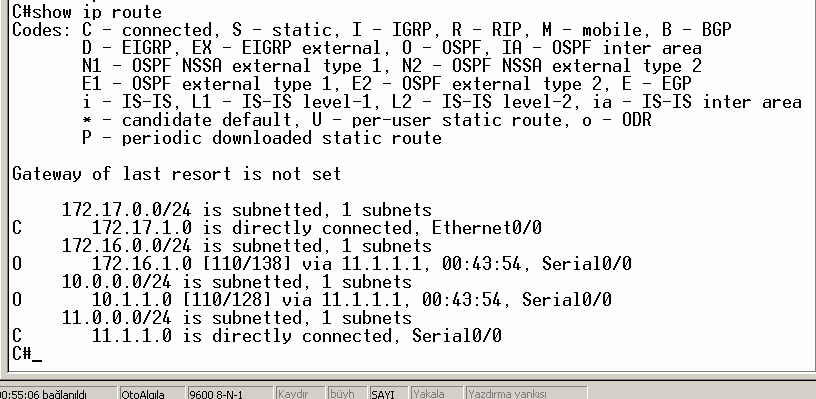
(A Router’ ının Routing Table’ ı)



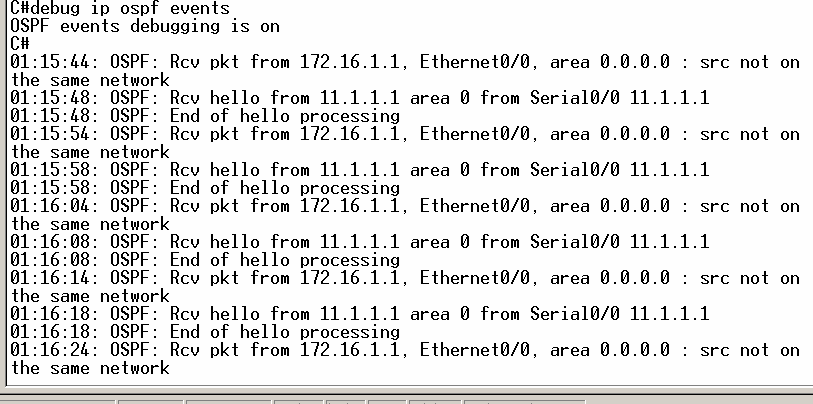
(B Router’ ının Routing Table’ ı)



(C Router’ ının Routing Table’ ı)

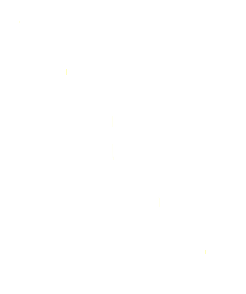
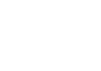
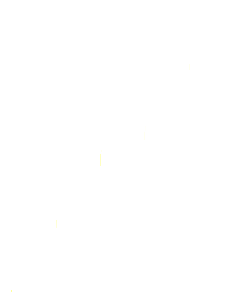
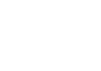
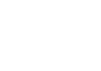


(Hello Paketleri)



# OSPF Laboratuar Calismalari

**192.168.6.0 /24**



**ankara**

**18.1.1.17 /30**

**17.1.1.18 /30**

**istanbul**

**17.1.1.17 /30**

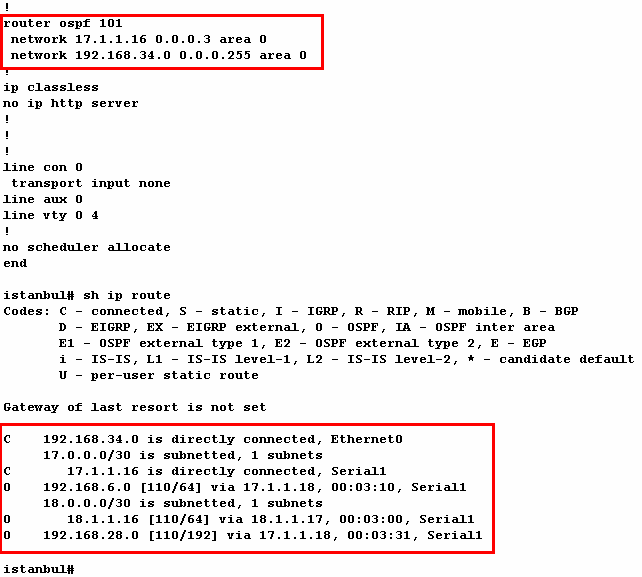
**Area 0**

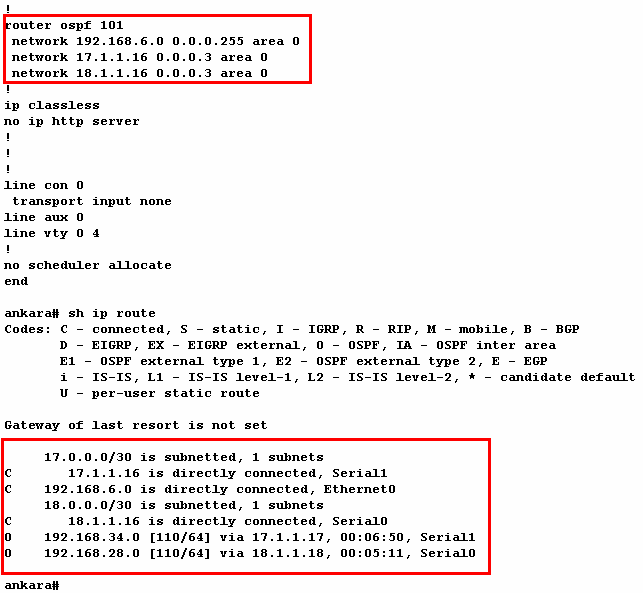
# 18.1.1.18 /30

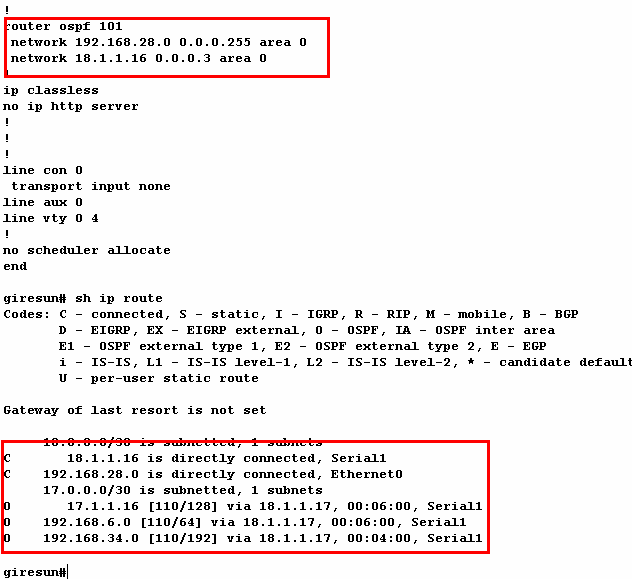
**giresun**

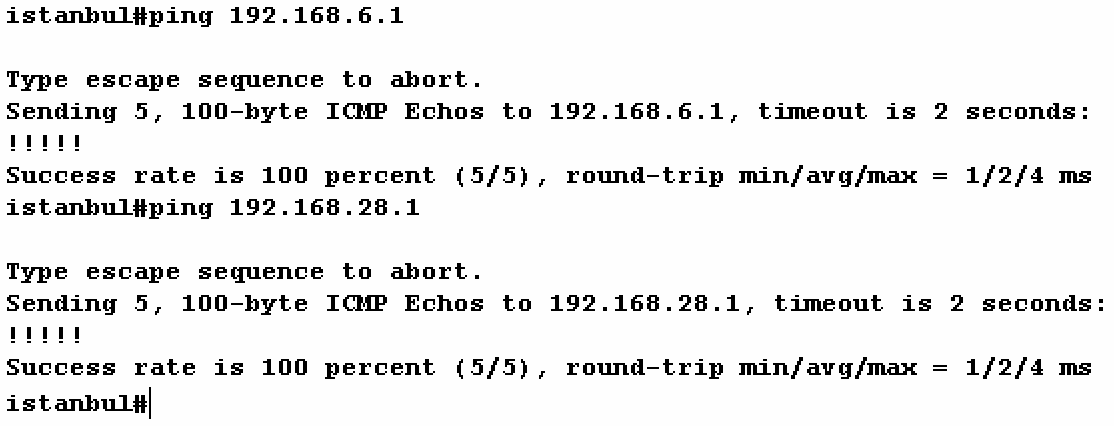
**192.168.34.0 /24**

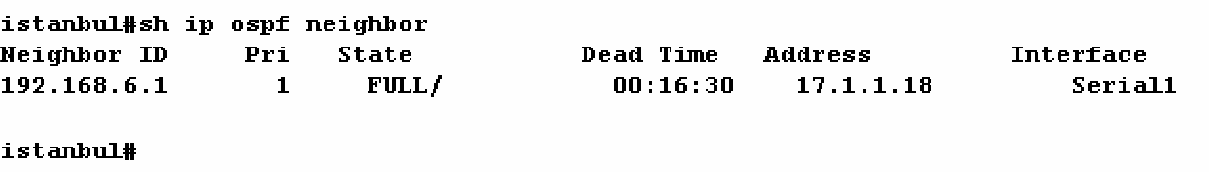
**192.168.28.0 /24**

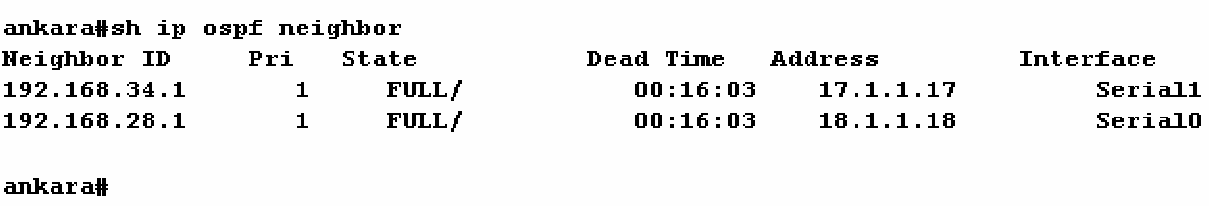


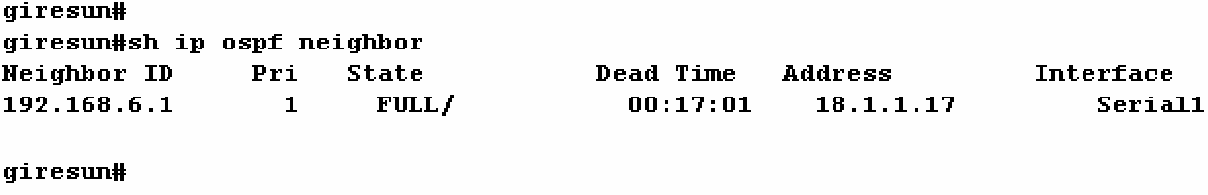


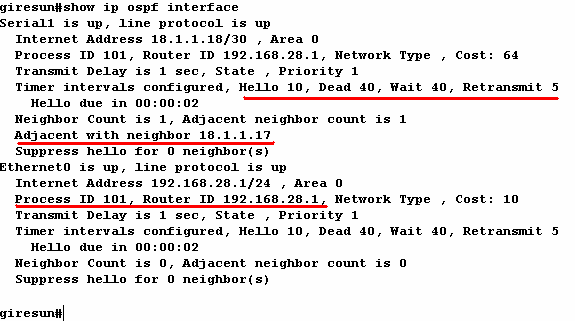










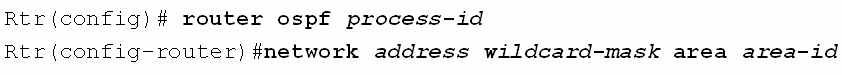


**OSPF Ozet**

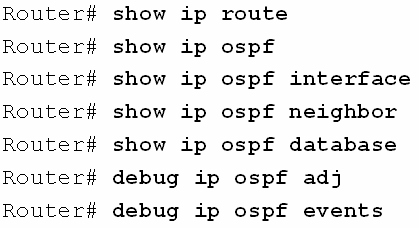
* + - * + Link State bir protokodur.
* Hizli yayilma ozelligine sahiptpr.
* VLSM (Variable Length Subnet Mask) ve CIDR (Classless Inter Domain Routing) destegi vardir.
* Metric hesabi tamamen bant genisligi genisligi uzerine kuruludur.
* Distance Vector protokollerin aksine periyodik updateler yapmaz, 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. Routerlarin komsuluk iliskisi kurabilmeleri icin Hello ve Dead Intervallarinin ayni olmasi gerekir. Hello ve Dead Interval araliklari degistirilebilir.



* Broadcast Multi Access ve Non-Broadcast Multi Access networklerde butun trafigi DR denen router yonetir, BDR ile yedeklenmistir. Bu networklerde Routerlar sadece birbirlerine Hello paketleri gonderirken diger butun paketler DR uzerinden gerceklesir.
* Konfigurasyonu oldukca basittir.



* Asagidaki show komutlari ile olaylar goruntulenebilir.



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

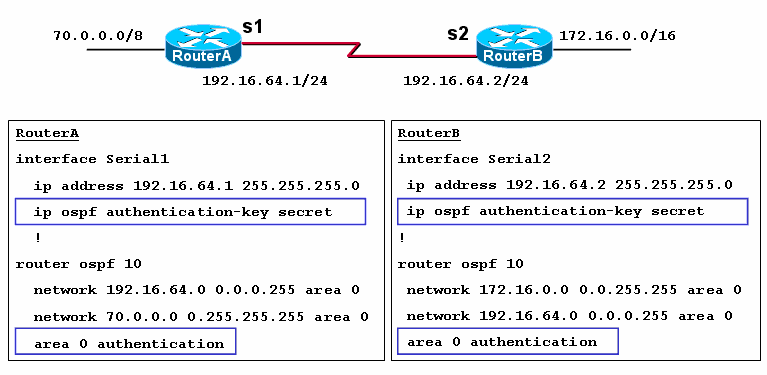
<text omitted>

04:20:23: OSPF: Synchronized with 201.0.0.1 on FastEthernet0, state FULL

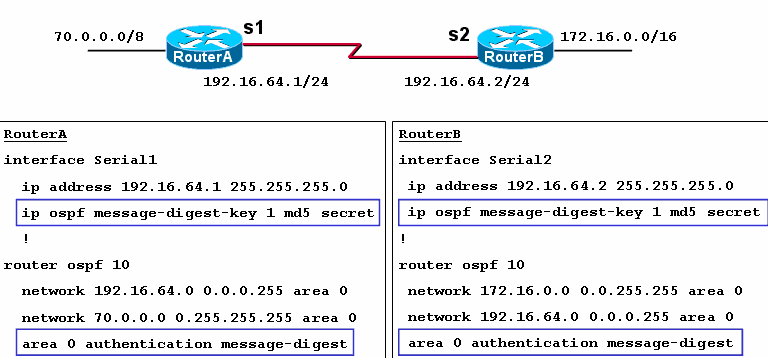
# Extralar

**Authentication Konfigurasyonu yapilabilir.**

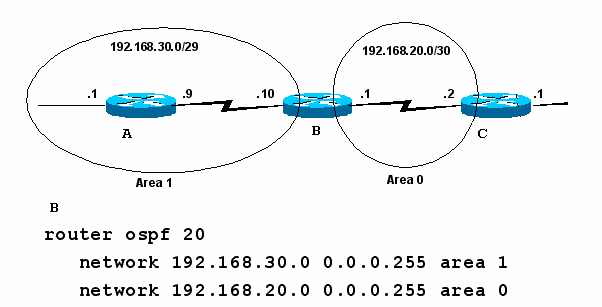
(Basit)



(MD5)



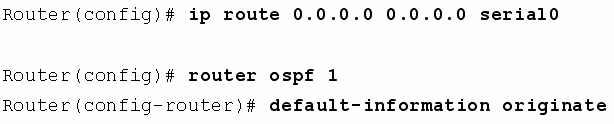
# Iki farkli Area Area Borde Router (ABR) denen Routerlar ile haberlesebilirler.



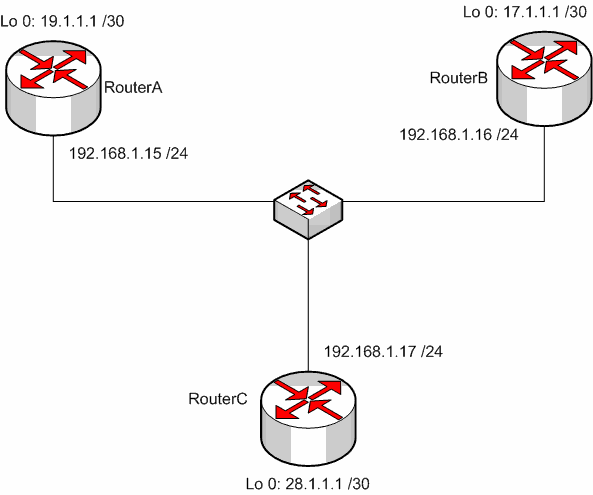
Burada B Routeri Area Border Router dir ve interfacelerinden biri area 0’ da bir digeri area 1’ dedir.

# Default Route yapilabilir.

Bunun icin static default route OSPF konfigurasyonu icine gomulmelidir.



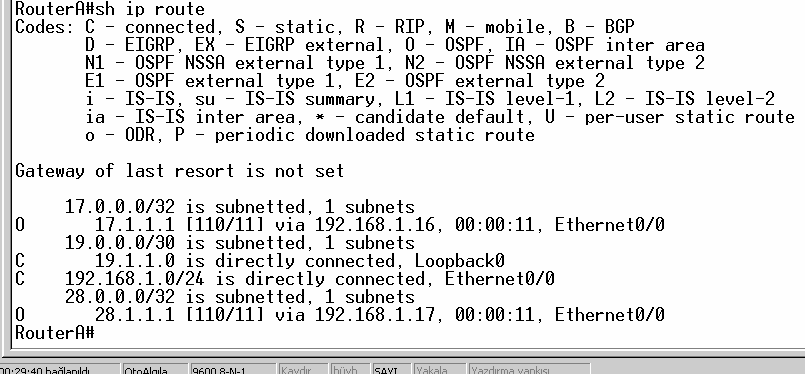
# OSPF DR-BDR Secimi Lab. Calismasi

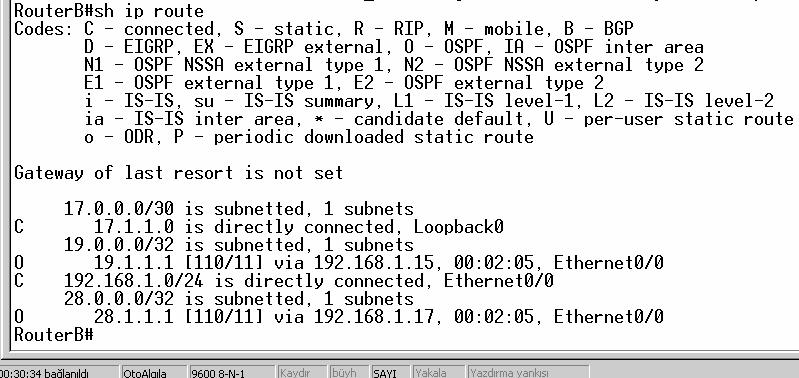


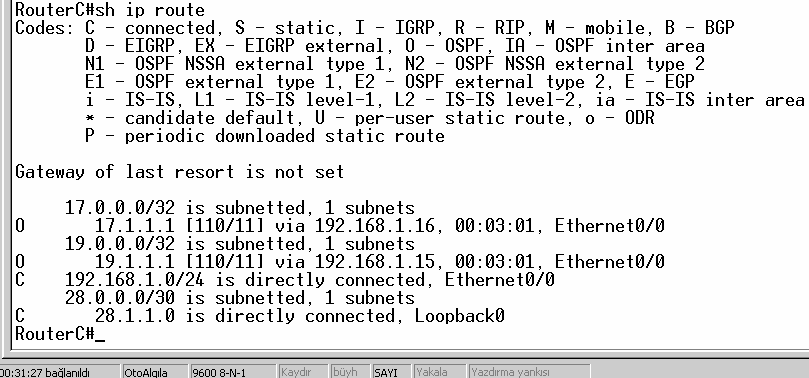
Bu calisma icerisinde DR ve BDR secimlerinin anlasilmasi amaclanmistir. Laboratuar imkanlirinin elverdigi olcude tasarlanan senaryo daher router ayni ethernet networkune baglanmis ve her Router uzerinde Loopback adresleri tanimlanmistir.

Routerlarda OSPF konfigurasyonu yapilirken Loopback betworklerde tanitilmistir.

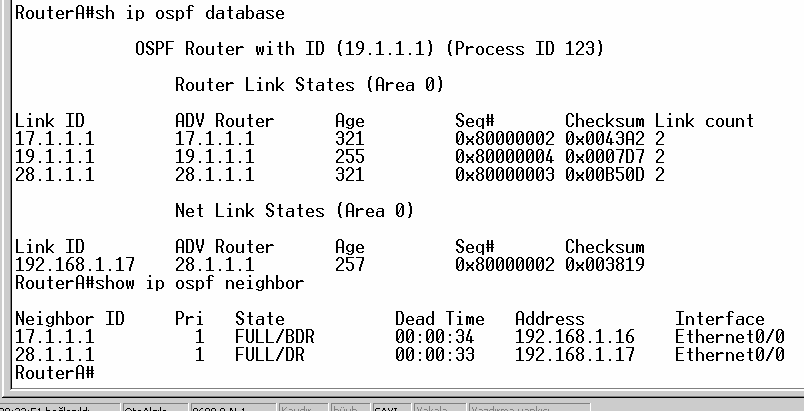
Konfigurasyon ve convergence tamamlandiktan sonra Routing Table’ lar sagidaki gibi olusmutur.



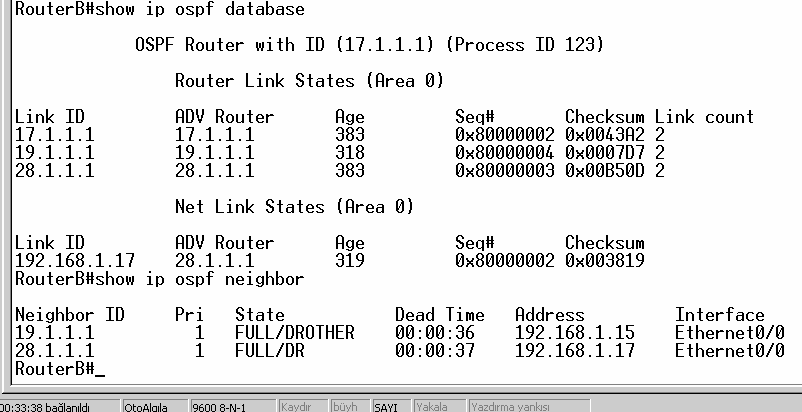


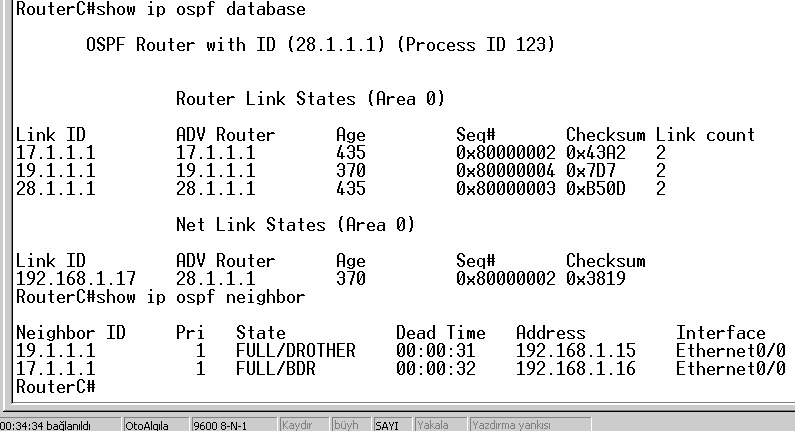


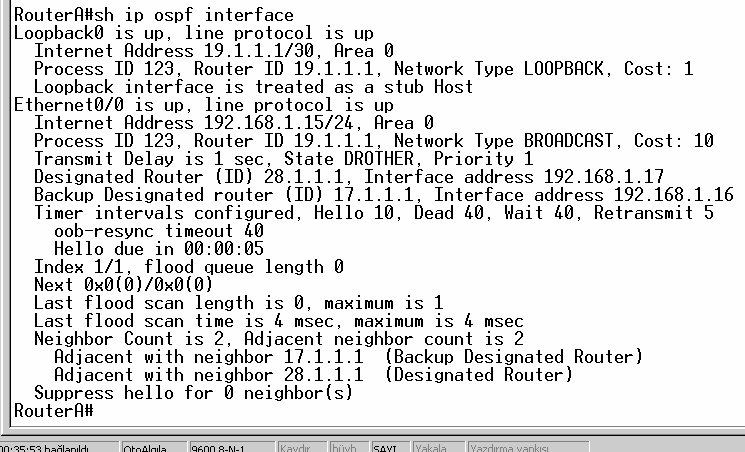
Routing Table’ larin ardindan OSPF database’i ve Ospf komsulari incelenmistir. Bu incelemede DR ve BDR’ lar detayli gorulebilmektedir.

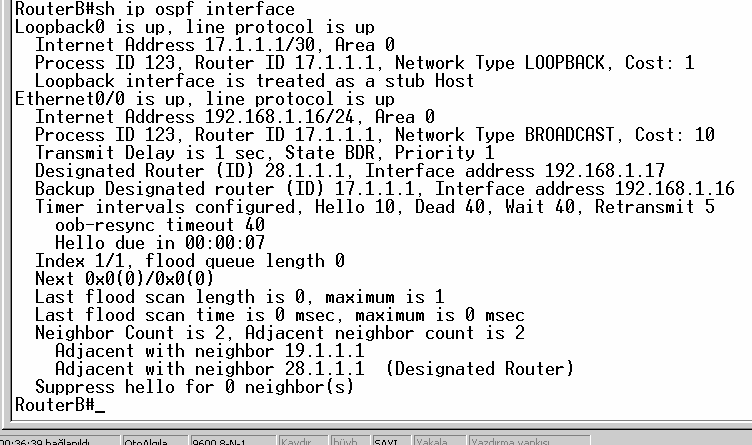


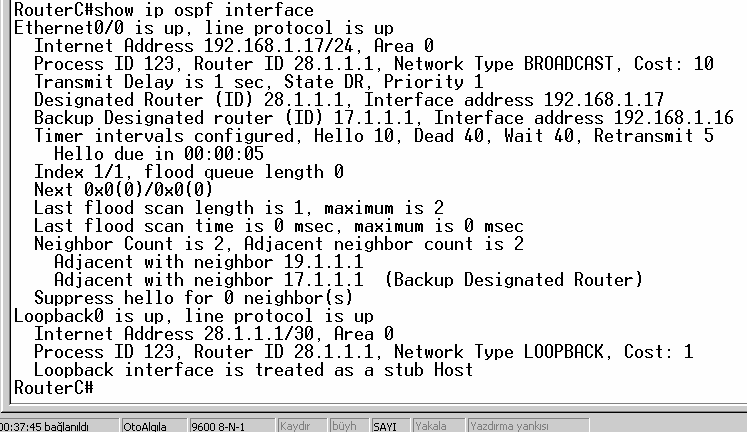
RouterA’ dan alinan bu goruntude komsu routerlar ve bu routerlar ile olan iliski tespit edilebilmektedir. Ornegin 28.1.1.1 ID’ sine sahip Router ile Full komsuluk iliskisi kurulmus ve DR olarak kabul edilmistir. (Bunun boyle olacagini zaten biliyorduk zira 28.1.1.1 ortamdaki en yuksek ID)











Show ip ospf interface komutu ile aldigimiz goruntulere baktigimiz da 17.1.1.1 ID’ li routerin BDR secildigini oysa daha yuksek ID’ ye sahip 19.1.1.1 ID’ li routerin DROther olarak kaldigini goruyoruz ki bu karmasik bir durum.

OSPF konusan routerlar ortama daha yuksek ID’ ye sahip bir Router katildiginda onu Drother olarak alirlar, yeniden bir DR – BDR secimine gitmezler. Bu Cisco’ nun bir bug’ idir. Anlasilan o ki ornegimiz de 17.1.1.1 ID’li router ortama daha once katilmis ve 19.1.1.1 ID’ li router up olmadan BDR secmimi tamamlanmis.

Cisco’ nun bu bug’ ini asmak icin interfacelerden en azindan birini down – up yapmamiz 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)#

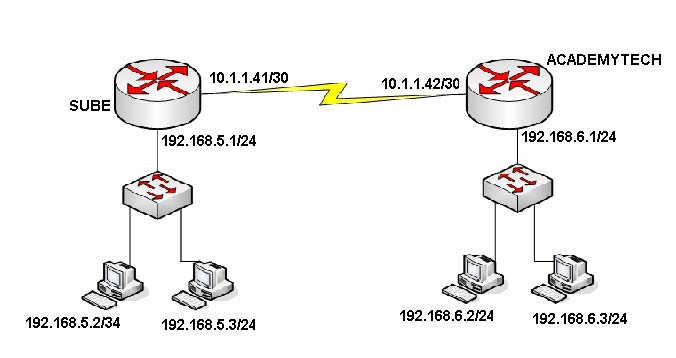
# Routing Protokollere Genel Bakış

Bu bölümde Routing Protokolleri genel olarak inceleyeceğiz. Routing Protokolleri genel olarak üç grup halinde inceleyebiliriz.



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 sistemdeki bütün networkleri öğrenmiş olarak Routing Table’ larını son haliyle oluştururlar.



Routing Protokollerin karşılaştırılması sırasında şekildeki topolojiden hareketle konfigürasyonlar yapılacak ve karşılaştırmalar gerçekleştirilecektir.

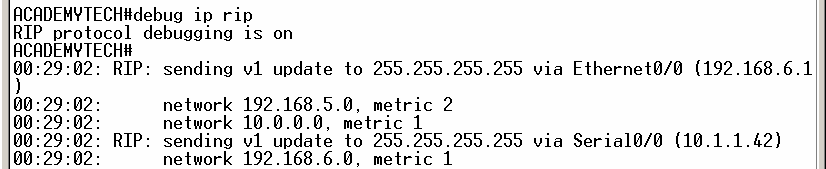
Routerlar en iyi yol seçimi yaparken (Best Path Determination) referans olarak Routing Table’ larında ki bilgileri alırlar. Dolayısıyla Routing Protokoller kullanarak oluşturulan Routing Table’ ların sistem başladıktan belirli bir zaman sonra son halini alacak olması bir dezavantaj olarak görülebilir. 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.

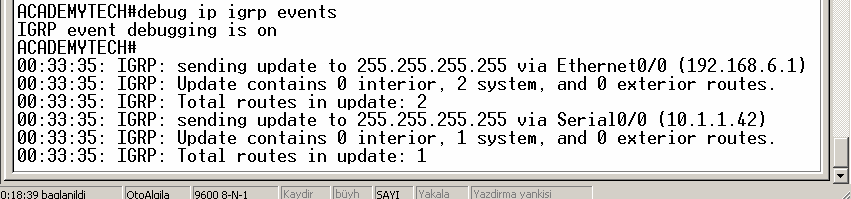
Burada Update sürelerini baz alarak Routing Protokolleri karşılaştırabiliriz.

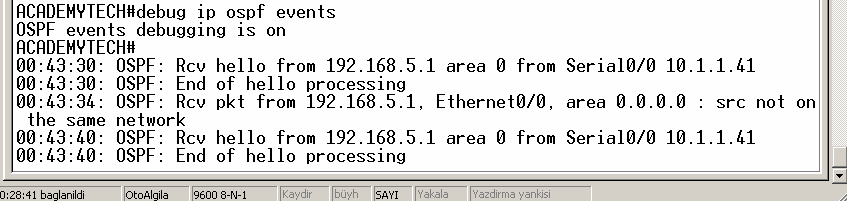


Rip, Ripv2 ve IGRP’ de updateler belirli zaman aralıklarında yapılırken, EIGRP ve OSPF için update gerektiğinde yani sistem üzerinde bir değişiklik olduğunda, yeni bir network eklendiğinde veya bir network down olduğunda yapılır. Ve burada yine aklımızda tutmamız gereken konu EIGRP ve OSPF gerektiğinde yaptığı updatelerde 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 izleyebiliyorduk.



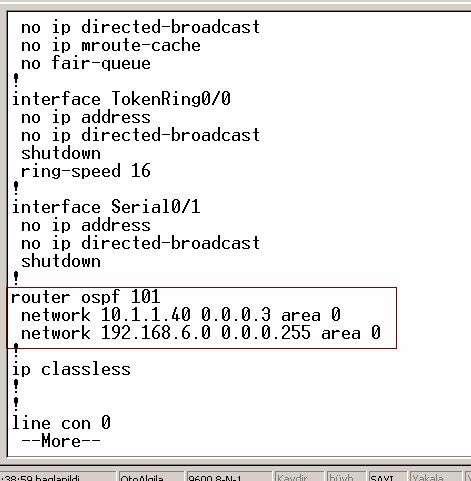




(OSPF Hello paketleri)

Routing Protokollerden bahsederken bahsettiğimiz konulardan biri de bazı protokollerin VLSM (Variable Length Subnet Mask) desteği verirken bazılarının vermemsiydi. Bundan kastettiğimiz şey protokollerin Classless veya Classfull olmalarıdır. Anladığınız gibi Classfull bir protokolde kullanacağımız network adreslerinde Subnet maskı biz belirleyemeyiz, protokol o adresin ait olduğu sınıfa göre Subnet maskını kabul eder. Classless protokollerde ise Subnet mask tamamen bizim kontrolümüzdedir.

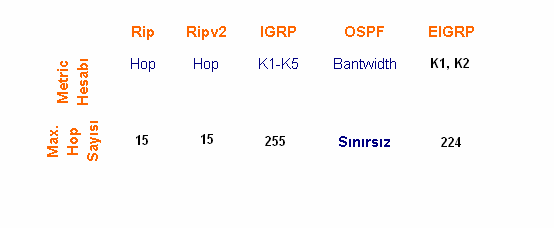




(OSPF Runnin-Config, VLSM Desteği)

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 maskı 255.0.0.0 olarak kabul edeceklerdir.

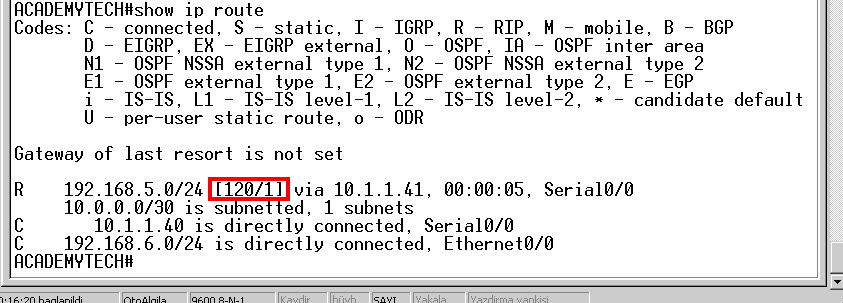
Routing Protokoller metric hesaplarında farklı kritelere bakarlar. Rip tamamen hop sayısına bakarken IGRP bahsettiğimiz K1’den K5’e kadar olan değerlere büyük ölçüde bant genişliğini baz alarak bakar. Tıpkı değişik kriterlere göre metric hesabı yapıldığı gibi Routing Protokollerin çalışacakları maksimum hop sayıları da farklı farklıdır.



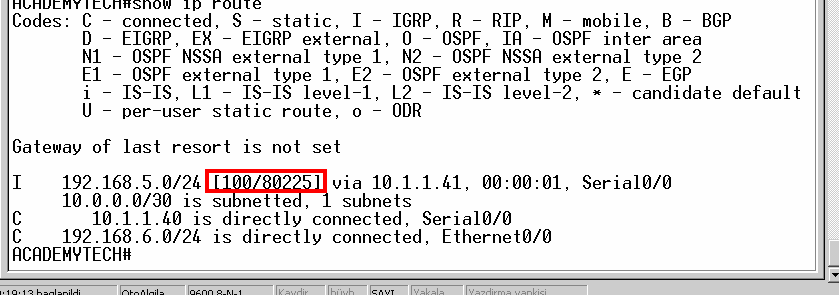
Routing protokoller Autonomous System numaralar kullanıp kullanmadıklar ve bir Area mantığı içine girerek hiyerarşik bir yapı oluşturup oluşturmadıklarına göre de incelenebilirler.



Routing protokollerde Administrative Distance ve metric hesaplarında, Routing Table’larına bakıldığında detaylı bilgi sahibi olunabilir.

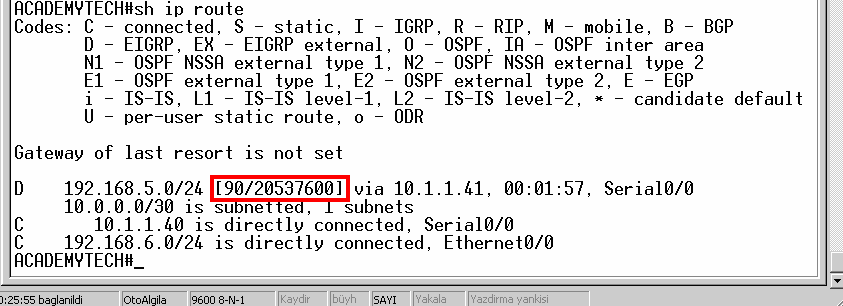


Rip için Routing Table görüntülendiğinde Administrative Distance’ ının 120 olduğu görülmektedir. Parantez içerisindeki bir diğer ifade (“1”) metriği yani Rip için hop sayısını belirtmektedir. Routing Table’ da Rip protokolüyle öğrenilen Networkler “R” harfi ile belirtilirler.

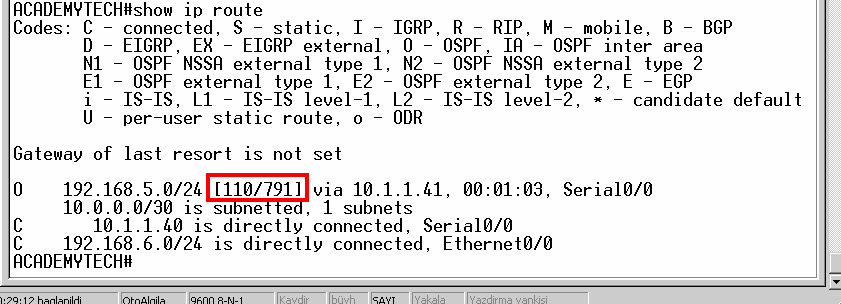


Igrp için Routing Table görüntülendiğinde Administrative Distance’ ının 100 olduğu görülmektedir. Parantez içerisindeki bir diğer ifade (“80225”) metriği belirtir. K1 ‘den K5’e kadar olan kriteler baz alınarak hesaplanmıştır.

OSPF ve EIGRP Routing Table’ ları aşağıdadır.



(D: EIGRP, Administrative Distance=90)



(O: OSPF, Administrative Distance=110)

# Cisco Ozel Protokoller

IGRP ve EIGRP Cisco ozel protokollerdir. Diger butun protokoller ise publicdir. Dolayisiyla sistemimizde Cisco disinda ureticilere ait Router’ larda varsa IGRP ve EIGRP bizim icin dogru secim olmayacaktir.

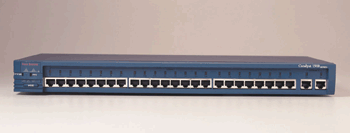
IGRP ve EIGRP Cisco tarafindan uretildiklerinden ayni AS icinde birbirleriyle haberlesebilirler. Fakat bu durumda sistemin IGRP konusan network bilgiler EIGRP konusan Routerlarda External EIGRP olarak etiketlenir ve bu networkler icin Administrative Distance 170’ dir.

# IPX- AppleTalk Destegi

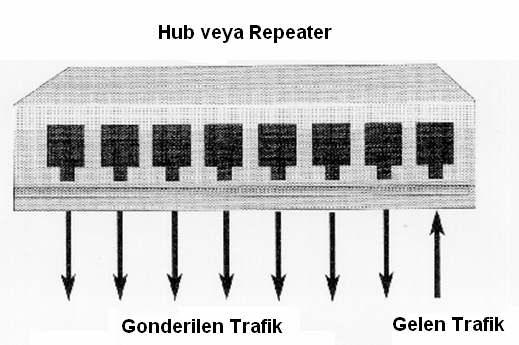
Cisco ozel bir protokol olan EIGRP Ip disinda IPX ve AppleTalk networklerini de desteklemesiyle diger protokollerden ayrilabilir.

Layer 2 Switching



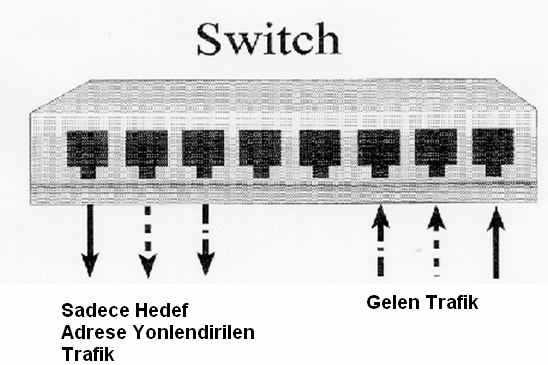


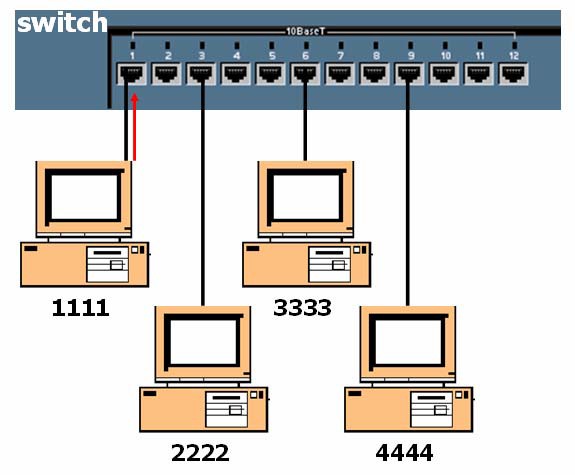
Hub'lar ile çalıştığımızda , hub aslında çok portlu bir repeater olduğu için ağdaki tüm bilgisayarlar aynı çakışma alanı içinde olacaklardır. Bu alana “collision domain” denire. Dolayısıyla network performansında düşme olacaktır.



Bu problemin çözümü olarak networklerde Switch adı verilen cihazlar kullanılmaya başlanmıştır.

Switch OSI 2. katmanda yani Data Link Layer Katmanında çalışır. Bir portuna bağlı bilgisayar veya bilgisayarları gonderdikleri framelerden 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.





Sekildeki yapi sistemin yeni baslatildigini varsayarsa. Switch 1111 MAC adresine sahip bilgisayarin gonderdigi frame’ I alacak ve buradaki source ma cadres alanindan okudugu degeri ma adresi tablosuna yazacak.

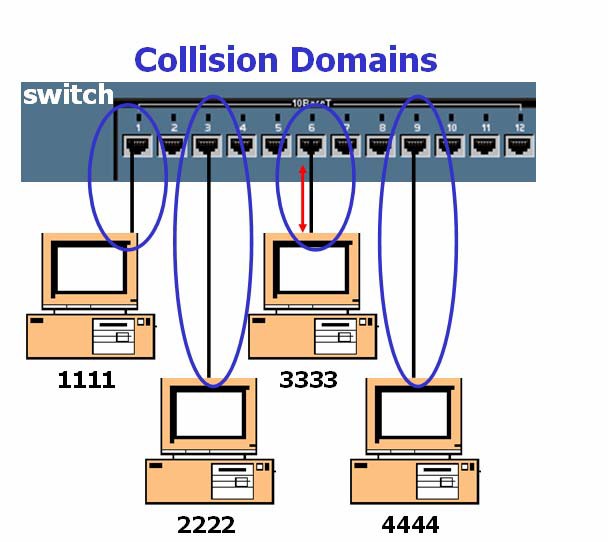
Su an icin MAC adresi tablosunda ki tek girdi 1111 MAC adresi oldugu icin hedef mac adresinin hangi portta oldugunu bilmedigimizi soyleyebiliriz. Bu durumda frame butun portlardan flood edilecektir.

Bu sekilde zamanla switch butun portlarinda ki bilgisayarlarin mac adreslerini onlari cache’ in 300 saniye tutmak uzere mac adresi tablosuna yazacaktir. Artik hedef MAC adreslerine gore frameleri yonlendirebilir.

Switche 300 saniye boyunca ilgili MAC adresinin bulundugu porttan bir istek gelmez ise o adres tablodan silinecektir.

Simdi sozgelimi 3333 mac adresine sahip bilgisayardan 1111 mac adresine sahip bilgisayar bir istek gonderildigini varsayalim. Bu durumda switch mac adresi tablosundan 1111 mac adresli bilgisayarin 1. portunda oldugunu group istegi sadece o porta fonderecektir. Bu sayede olasi collision’ lar engellenmistir.

Iste bu sebeple Switchin her bir portu bir Collision Domain’ dir denilebilir.

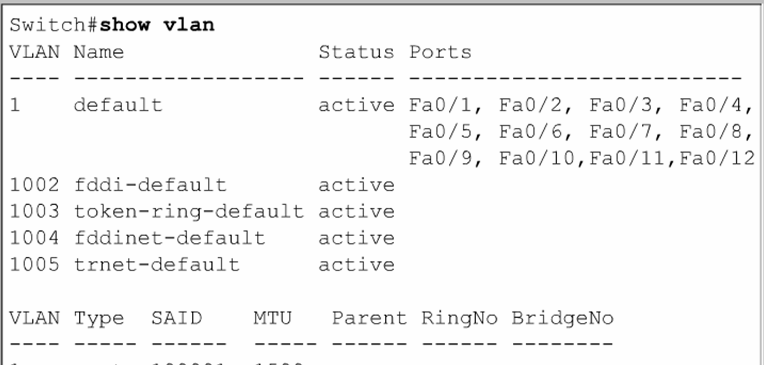
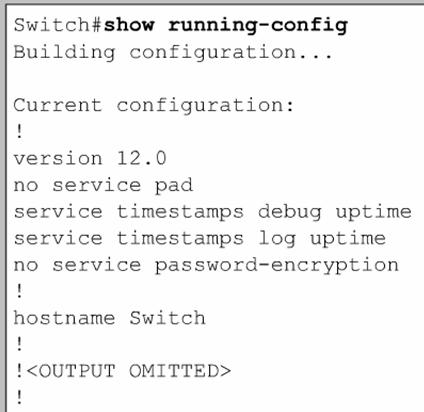


Switch Konfigurasyonu

Switch konfigurayonu bir cok yonde Router ile aynidir. Switch acildiginda user moddadir ve ‘enable’ yazilara Enable moda gecilebilir.

Switch>enable Switch#

Tipki Routerda oldugu gibi Switch de yaptigimiz konfigurayonlari goruntuleyip sorun cozmemizde bize yardimci olacak show komutlari vardir.



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 yonetim VLAN’idir ve switch icin verilecek ip adresi bu VLAN’da, default gateway adresi Global Configuration modda verilmelidir. 1900 serisi switchlerde ise durum biraz farklidir.

Switch(config)#interface vlan 1

Switch(config-if)#ip address 192.168.1.10 255.255.255.0 Switch(config-if)#exit

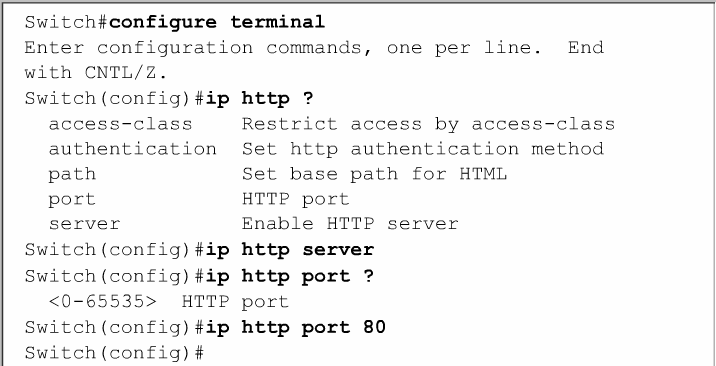
Switch(config)#ip default-gateway 192.168.1.1

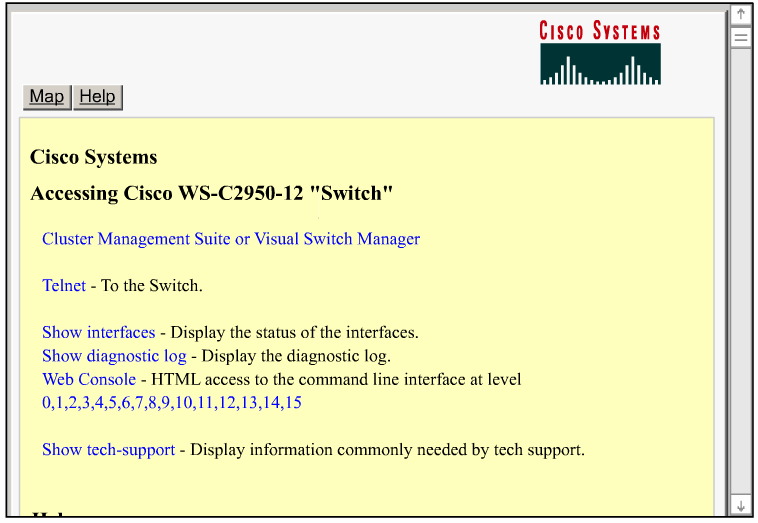
1900 serisi switchlerde ise durum biraz farklidir. Bu switchlerde ip adresi ve default gateway adresi Global Configuration modda verilir.

Switch(config)#ip adress 192.168.1.10 255.255.255.0

Switch(config)#ip default-gateway 192.168.1.1

Switche web browser ile erisilebilir. Bunun icin su konfigurasyon yapilmalidir.

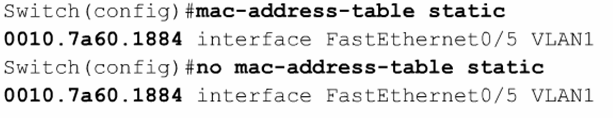




MAC Address Table

Daha once de belirttigimiz gibi switchler aldiklari framelerdeki source mac adres alaninda ki bilgiler ile MAC adres tablolarini olustururlar ve bu tablolara gore frameleri filtrelerler.

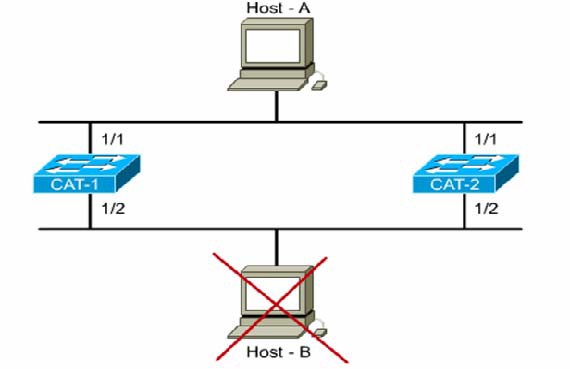
Bununla birlikte switchlere MAC adreslerinin static olarak atanmasi da mumkundur. Guvenligi artirmak icin yapilabilecek bu uygulamayla ayni zamanda 300 saniyelik max. Age suresi de gecersiz olacaktir.



Spanning Tree Protocol (STP)

STP Layer 2 cihazlarin haberlesmi sirasinda dogabiliecek olasi donguleri (loop) onleyen bir protokoldur. STP yapisi geregi kullandigi algorltma (Spanning Tree Algorithm) ile dongulere neden olmayacak bir topoloji olusturur.

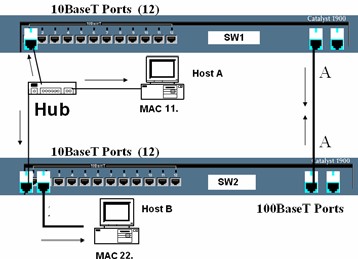
Ethernet Frame’ leri TTL alanina sahip olmadiklari icin STP ve loop yaratmayacak bir dizayn onemlidir. Aksi takdirde olusacak donguler switch kapatilana kadar devam edecektir.



Sekildeki gibi bir yapida, switchler kendilerine gelen ve hedefi bilinmeyen paketleri diger butun portlardan flood edeceklerine gore ciddi sorunlar yasanacaktir. 1/1 portlarindan frame’ i alan her iki switch de flood edecek ve hemen sonrasinda yine her iki switch ayni frame;leri ½ portlarindan alacak, devaminda neler olacagini kestirebiliyorsunuzdur sanirim ☺

Hattlari daha da zor durumda birakacak paketler ise Broadcastlerdir. Bilindigi gibi Switvhler Broadcast gecirirler, dolayisiyla kendilerine gelen broadcaastleri butun portlarina gonderirler.

Asagidaki sekilden hareketle Host A nin soz gelimi bir ARP Request’ te bulundugunu varsayalim. Arp Request frameleri broadcast oldugundan 1. portlarindan bu frameleri alan her iki switchde diger butun portlarindan bu frame’i iletecektir.



Her iki switch arasinda ki A ile gosterilmis baglantidan da broadcastler yayinlanacak dolayisyla her iki switch de bu broadcast frame leri bu kez farkli portlardan olmak uzere, yeniden alacaklar ve yeniden flood edecekler. (Bu durum Broadcast Storm olarak bilinir. )

Bu vebunun gibi bir cok nedenle dogabilecek sorunlar icin yardimimiza STP kosacaktir ☺

STP nin amaci genel olarak networklerdeki olasi loop lari onlemek ve bunun icin de her hedefe sadece bir yolun aktif olarak calismasini saglamaktir. Bunun icinde STP Spanning Tree Algorithm’i (STA) kullanir.

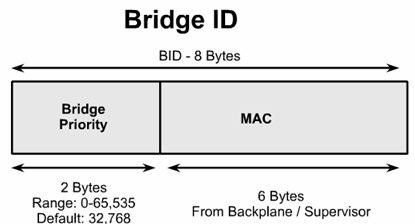
STA networkte bir referans noktasi olusturur ve bu referans noktasindan hareketle, birden fazla alternatif yol varsa, en iyi yol secimini yapar. Bu referans noktasina Root Bridge denir.

Peki Root Bridge nedir, nasil secilir, kim secer...

Aslinda ortamdaki butun switchler Root Bridge’dir. Yani kendilerini oyle sanarlar ☺

Ortamdaki en **kucuk** Bridge ID’ ye sahip bridge Root Bridge’dir.

Bridge ID Bridge Priority ve MAC adresinden olusur, 8 Byte’tir. Butun switchlerin Priority’si default olarak 32768’ dir.



Switchlerin default priority’ leri degistirilmediginde hepsi esit olacagindan MAC adreslerine bakilabilir, bu durum da en kucuk MAC adresine sahip Bridge Root olacaktir. (Switchlerin efendisi

☺ )

STP hesaplamalari sirasinda en iyi yol seciminin yapilmasini saglayacak kriter de Path Cost’ tur. 1000/ Bandwith ile hesaplansa da IEEE cok kullanilan bant genislikleri icin costlari yayinlamistir.

* 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 cok hesaptan kurtulduk sanirim ☺)

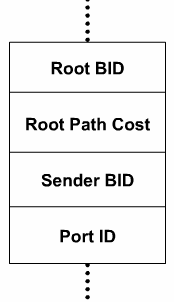
Butun bu ogrendiklerimizin isiginda switchlerin kriter olara aldiklari 4 adimi siralayabiliriz.

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 msajlari 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 arasinda daha iyisi varsa onu alip digerini silecektir.

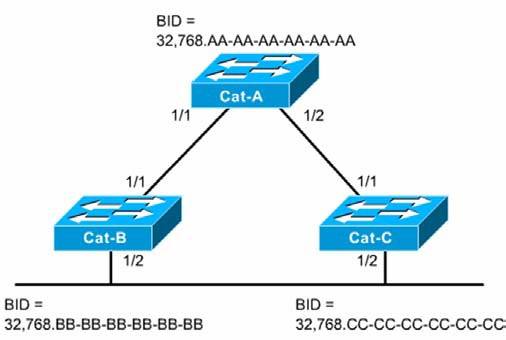


(BPDU Mesaj icerigi)

STP 3 basamak ile yapisini olsturur. Step 1 Root Bridge Secilir

Step 2 Root Portlar Secilir

Step 3 Designated Portlar secilir.



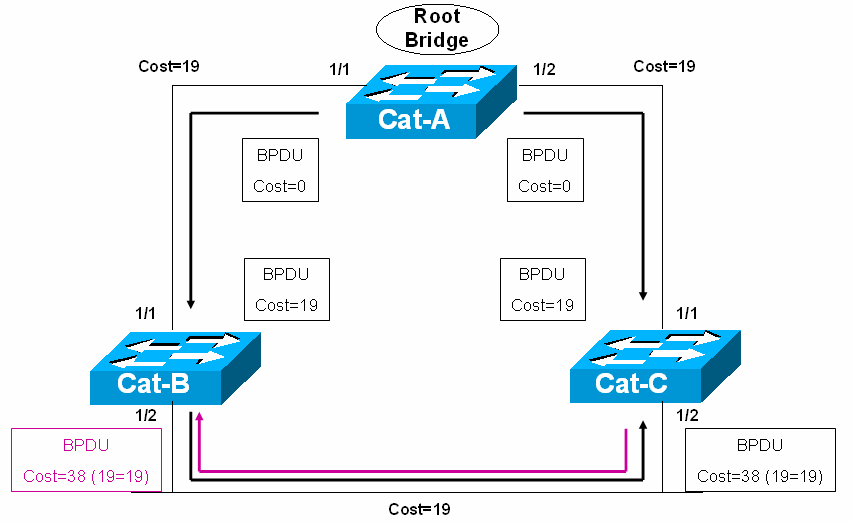
Sekilde ki gibi bir yapi uzerindenhareket ederek bu adimlari aciklamaya calisalim. Ilk adimda Root Bridge secilmesi gerekir ve bunu icin BPDU Mesajlari gonderilir. Oncede soyledigimiz gibi her

Switch kendisini Root Bridge varsayacagi icin BPDU mesajlarindanki Root Bridge ID alanina kendi Idlerini yazar. (Root War basladi ☺)

Cok gecmeden durumun oyle olmadigini anlarlar aslinda ortamda ki CAT-A Switchinin gercek Root oldugunu ogrenip BPDU larina bu Swirchi Root olarak eklerler. Root Bridgein portlari her zaman Designated porttur ve surekli forward durumundadir.

Root Bridge secildigine gore ikinci adima, Diger Switchler icin Root Portlarin secilmesine gecebiliriz.

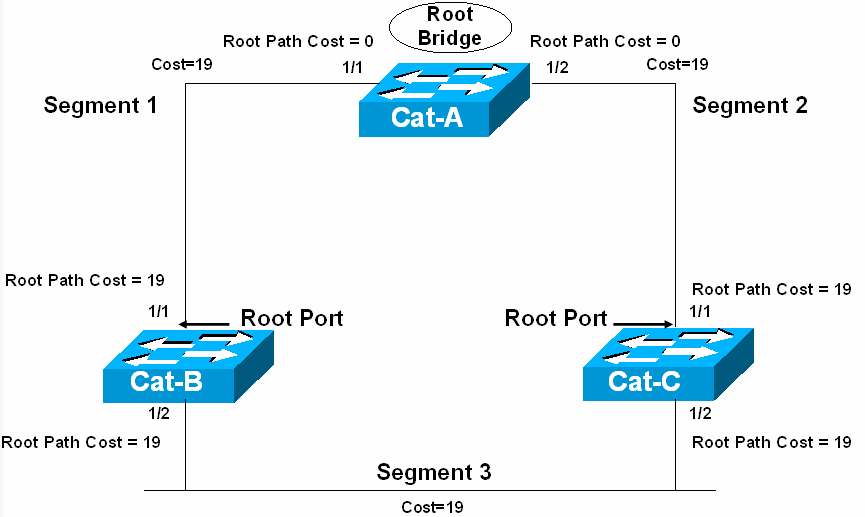
Switchlerin Root Bridge’e en yakin portlari Root Porttur. Buraa yakinliktan kast ettigimiz sey aslinda Root Bridge olan costtur.



Sekilde costlari inceledigimiz zaman rahatlikla Cat-B ve Cat-C’ nin 1/1 portlarinin Root Port oldugunu soyleyebiliriz. Bu arada her Switch icin sadece 1 tane Root Port , her segment icinde sadece 1 tane root port olmali.

Root Portlar secildigine gore Designated portlara gecebiliriz.

Asagidaki sekilde de gorulecegi gibi Segment 3’ e dahil olan sadece 2 switch var ve bu switclerin birer tane secilebilen root portlari bu segmentte degiller. Bu yuzden bu segment icin bu iki Switch portlarindan biri Designated Port olarak secilmeli.



Yine sekilde goruldugu gibi Her iki Switchi birbirlerine baglayan portlarin 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

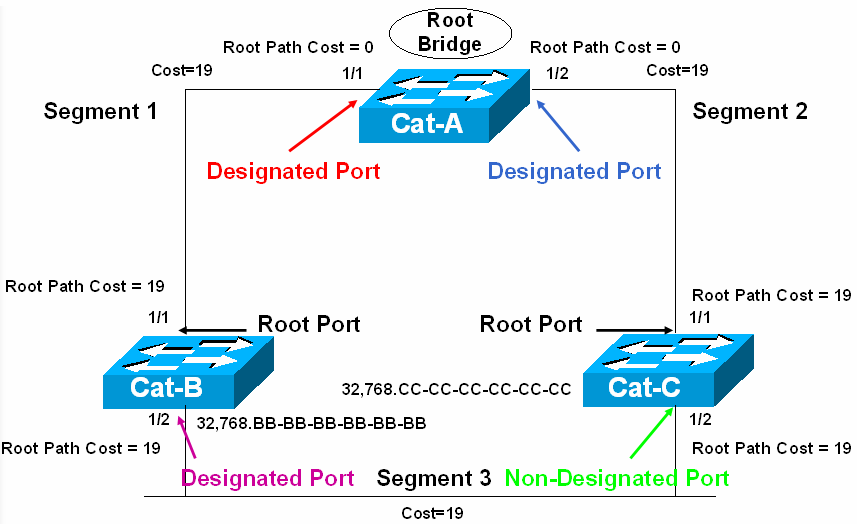
Step 2 - Lowest Path Cost to Root Bridge Step 3 - Lowest Sender BID

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 soyleyebiliriz.

Sonuc olarak Designated Port Forward duruma yani iletime 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 Durumlari

Spanning Tree yapisi icerisinde portlar 5 ayri durumda bulunabilirler.

1. Forwarding : Datalar gonderilir ve alinir.
2. Learning : Bridge Table olusturulur.
3. Listening :Aktif topology olusturulur.
4. Blocking : Sadece BPDU’lar alinir.
5. Disabled : Yonetimsel olarak down durumdadir.

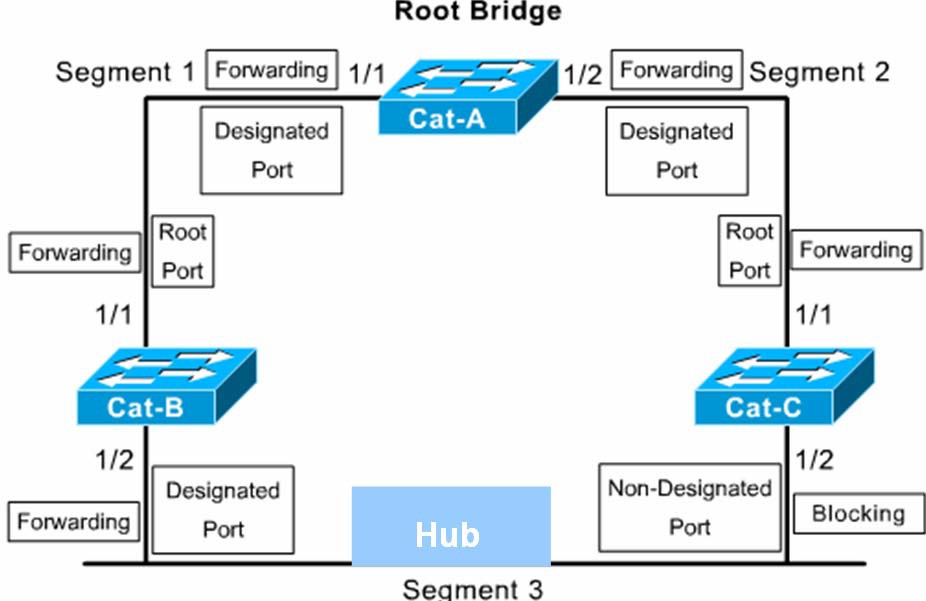
# STP Timers

**Hello Time:** Root Bridge tarafindan gonderilen BPDU mesajlari zaman araligidir. Default olarak 2 saniyedir.

**Forward Delay:** Portlarin Forward duruma gecmeden once Listening ve Learning adimlarinda gecen suredir, 15 saniyedir.

**Max. Age:** Bir BPDU’ nun saklanma suresidir, 20 saniyedir. 20 saniye boyunca daha once aldigi en iyi BPDU mesaji tekrarlanmazsa Max. Age dolmus olur ve port Listening Moda gecer.

Bir ornek ile STP zaman araliklarini inceleyelim.



Sekildeki duruma gore baslangicta Cat-C’ nin ½ portu Blocking durumda ve yanlizca BPDU mesahlarini dinliyor.

Simdi Cat-B’ nin ½ portunun down oldugunu varsayalim. Bu durumda Cat-C artik BPDU mesajlarini alamayacaktir. Cat-c 20 saniye boyunca Blocking durumda kalacak ve 20 saniyenin sonunda Max. Age’ e ulasildigi icin durumunu degistirecek, 15 saniye surecek Listening mod ve yine 15 saniye surecek Leraning Modun ardindan Forwarding duruma gececektir.

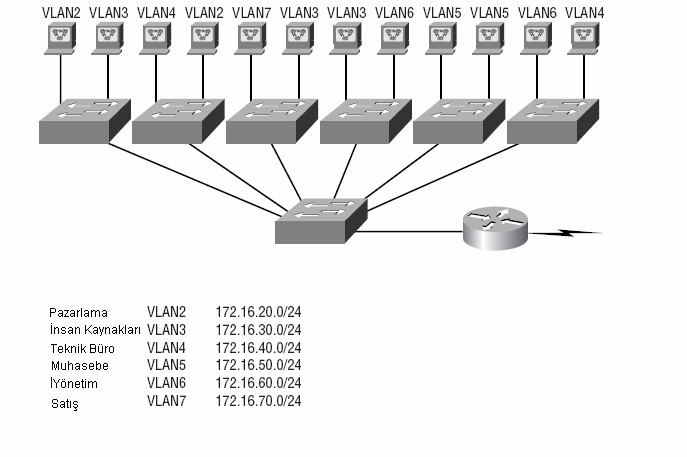
Yani 20 sn. max age + 15 sn. Listening + 15 sn. Learning modda kalacak dolayisiyla Cat-B ½ portu down olduktan 50 saniye sonra Cat-C ½ portu devreye girecekyir.

Fakat burada Cat-B’ nin MAC Adres Table’ inin silinmemesinden ve toplam 300 saniye botunca da Cache de kalacak olmasindan dolayi bir sorun var gibi gorunuyor.

Bu sorunda Root Bridge tarafindan 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 gruplama şekliden 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 VALN’ da bulundurulabilir ve bu sayede iki departman arasıdan ki iletişim engellenmiş olur.



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 bakımı Dinamik VLAN’ lara göre daha kolaydır.

Default olarak bir switch üzerindeki bütün portlar VLAN1’ dedir.

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 switchleriçin 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 switchler de her VLAN kendi alt modunda konfigüre ediliyor.

**NOT:** VLAN1 silinemez, değiştirilemez veya yeniden adlandırılamaz.

VLAN’ lar oluşturulduktan sonra artık ikinci adıma geçebiliriz. Bu adımda Switch portları VLAN’lar ile eşleştirilecek. Tabi burada VLAN üyeliğinin Static yada Dinamic olduğu da belirtiliyor.

# 1900 Seri Switchler için 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)#

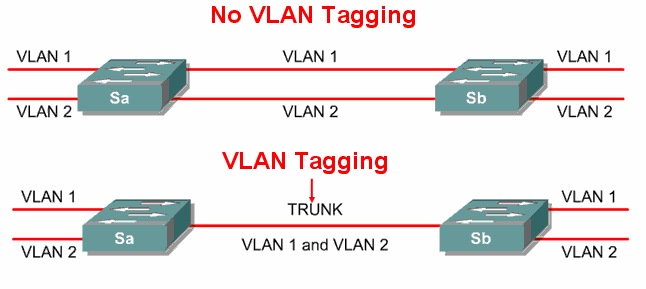
# 2950 Seri Switchler için VLAN Üyeliği:

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)#

# 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 tanımak için iki ayrı yöntem kullanır: **ISL** ve **IEEE802.1q**.



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

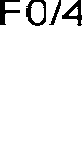
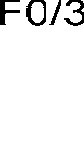
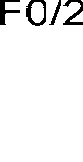
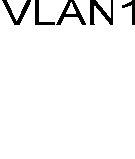
Bir VLAN'a bağlı cihazlar kendi aralarında iletişim kurabilir, broadcast'lerini gönderebilirler. VLAN'ların network'ü fiziksel olarak böldükleri varsayıldığı için VLAN'lar arasında cihazların iletişim kurabilmesi ancak 3. katman bir cihaza yardımıyla olacaktır.

Bu durumda yapılacak bir router üzerinde her VLAN için bir bağlantı eklemek ve Router üzerinde gerekli konfigürasyonları yaparak iletişimi sağlamaktır.



Böyle bir topoloji üzerinde çalıştığımızı varsayalım:

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’i altında sanal interface’ler oluşturmak, bu sanal interfacelereip 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 varsayarak encapsulation metodunu belirledik ve o sanal interfacein 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

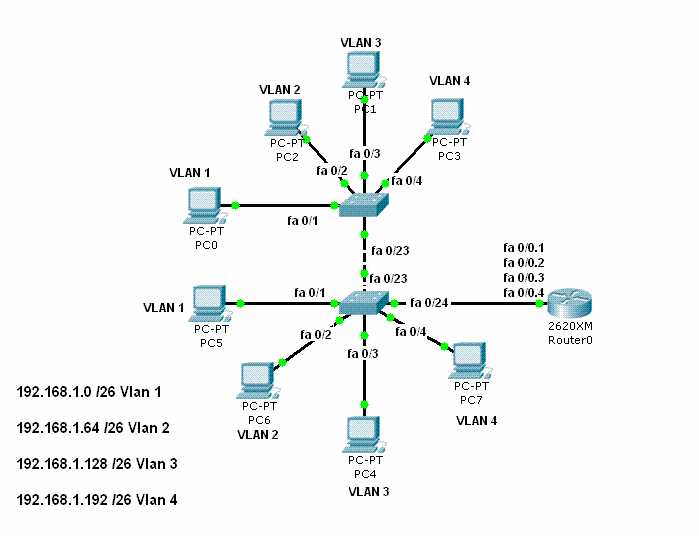
Router(config-subif)#ip address {ip adresi} {subnet maskı}

Şeklinde olacaktı. (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 calismamiz da VLAN 1 de dahil olmak uzere SwitchA ve SwitchB ye bagli 4 adet VLAN var. SwitchA ve SwitchB fa0/23 portlarindan birbirlerine baglanmis ve bu portlarda trunk uygulanmistir,

Ayni sekilde SwitchB fa0/24 portundan Routera baglanmis ve bu porttada trunk uygulanmistir. Encapsulation dot1q kullanilmistir.

Switch ve Router running-config dosyalari asagidadir.

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

1. egitim active Fa0/2
2. muhasebe active Fa0/3
3. 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

1. egitim active Fa0/2
2. muhasebe active Fa0/3
3. 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 konfigurasyonlarinin 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 alirlar.

* VLAN’lar VTP Server da olusturulur.
* VLAN bilgileri client switchlere gonderilir.
* Ayni domain icinde bulunan switchler VLAN bilgilerini alirlar.
* 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 portlarindan gonderirken kendisine gelen bilgileri almaz ve kendi VLAN database’ ini yapilandirmaz. Switchleri vu sekilde calismasi VTP Mode Transparent olarak adlandirilmistir. Bu modda calisan

Switchler VTP domaine katilmazlar.

Guvenlik acisindan VTP domainlerine password verilebilir. Bu durumda password o domain de bulunan butun switchlerde configure edilmelidir.

Gonderilen VTP mesaflari VTP database’ inden revision numarasi ile birlikte tutulurlar, her mesaj ile bu numara artirilir. Daha buyuk bir revision numarasi ile gelen bilgiler switchler tarafindan daha yeni olarak Kabul edilir ve gelen VLAN bilgileri eskilerinin uzerine yazilir.

Buraya kadar anlattiklarimizin isiginda VTP domainlerinde switchlerin 3 ayri modda calisabileceklerini soyleyebiliriz.

* 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

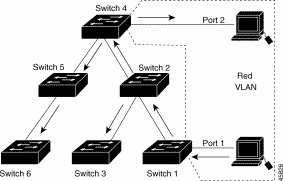
Switch# vlan database Switch(vlan)# vtp domain corp Switch(vlan)# vtp client

# VTP Pruning (Budama)

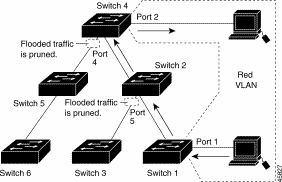
VTP Pruning networkteki broadcast, multicast, unknown unicast gibi gereksiz flood edilen paketleri azaltarak network bant genisligi kullanimini artirir. Cisco Switchlerde default olrak disable durumdadir.

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 ayni sekilde bu broadcast alacaktir. 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

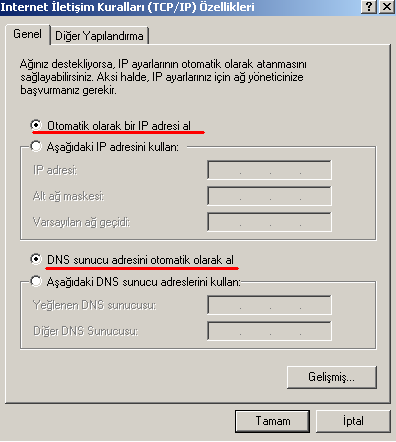
Switch(config-if)#switchport trunk pruning vlan remove *vlan*

Komutuyla pruning disinda birakilabilir.

# DHCP (Dynamic Host Configuration Protocol)

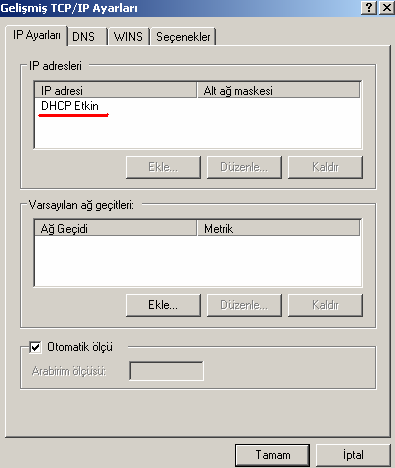
DHCP , DHCP kullanmak üzere yapılandırılmış bilgisayarlara merkezi ve otomatik olarak ip adresi atanması ile TCP/IP bilgilerinin yapılandırılmasını ve bunların yönetilmesini sağlar. DHCP’ nin uygulanması manuel olarak ip adresinin verilmesi nedeniyle ortaya çıkan bazı problemlerin azalmasını sağlar.

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 sunucu adresini otomatik olarak al ” seçeneği aktif hale getirilmelidir. Bu işlemler yapıldıktan sonra bilgisayarlar DHCP istemci durumuna gelecektir.



(Bir bilgisayarın 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 inceleyecek 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 dağıtılabilir 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’ larda 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. Yiene 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 dağıtılacağı bilgisinin Router’ a bildirilmesi gerekir.

Bunun için şu komutlar yazılmalı:

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 gerekir.

Bununla birlikte DNS ip adresi, etki alanı adı, NetBios Server ip adresi ve Default Gateway gibi adresleri de konfigürasyonunu yaptığımız da 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 bahsetmiştik. Bunun için ip adresi reserve edeceğimiz bilgisayarın MAC adresini bilmemiz gerekir. 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 kullanımda 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 etiğimiz ip adresleri hakkında bilgi,

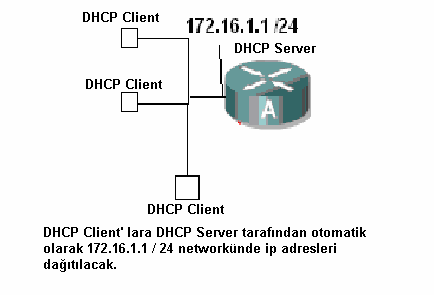
# Router# show ip dhcp conflict

Komutu ile dhcp’ de çakışan ip adreslerini görüntüleyebilir,

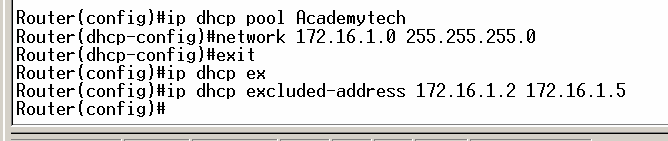
# Router# show ip dhcp server statistics

Komutu ile dhcp server hakkında istatistiksel bilgileri alabiliriz.

Şimdi örnek olucak bir konfigürasyon yapalım.

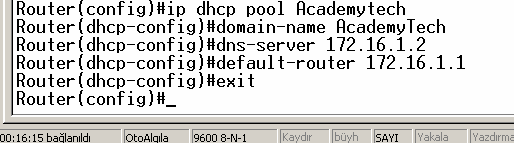


Senaryoyu biraz daha geliştirmek için 172.16.1.2 – 172.16.1.5 arasıdan ki ip adreslerinin dağıtılmamasını istediğimizi de düşünelim.

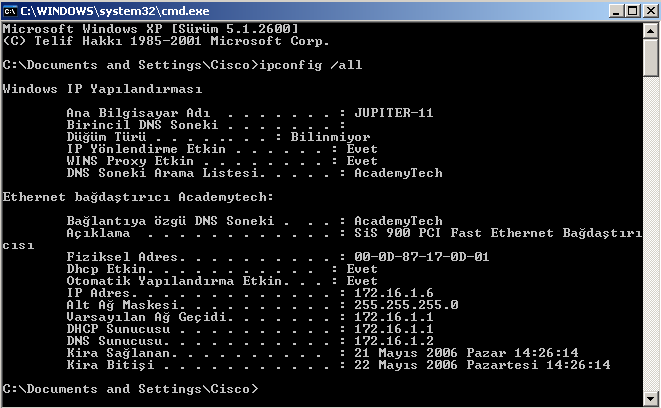


Burada ip havuzumu oluşturduk ve dağıtılmasını istemediğimiz ip aralığını router’ a bildirdik.

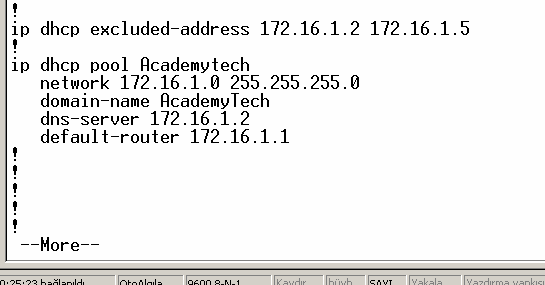
Söz gelimi etki alanı adımız “AcademyTech”, Dns Server’ın ip adresi:”172.16.1.2” ve Default Gateway’da 172.16.1.1 olsun. Bu bilgilerinde DHCP tarafından dağıtılmasını istersek konfigürasyona şu şekilde devam etmeliyiz:



Artık bilgisayarlarımızı DHCP Client olarak ayarladıktan sonra ip adreslerinin bizim router üzerinde yaptığımız konfigürasyona uygun olarak alacaklarır.



Bilgisayarın ip konfigürasyonunda görüldüğü gibi bizim istediğimiz şekilde bir çalışma oldu. Running Konfigürasyona baktığımız da ise DHCP ile ilgili şu bilgileri göreceğiz:



DHCP kullanmaktan vazgeçtiğimiz andan itibaren DHCP hizmetini devre dışı bırakabiliriz.



# 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 Adresses (özel adresler) diyoruz. İnternette kullandıklarımıza da Public (Halka Açık) Addresses diyoruzi

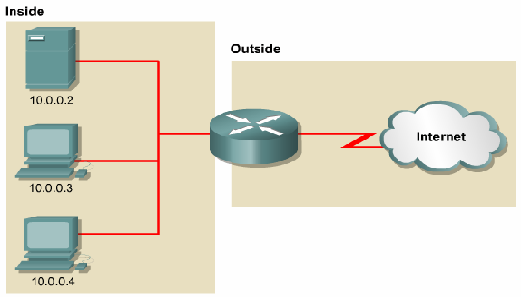
Özel IP adresleri RFC 1918 ile belirlenmiştir ve;

10.0.0.0 ile 10.255.255.255

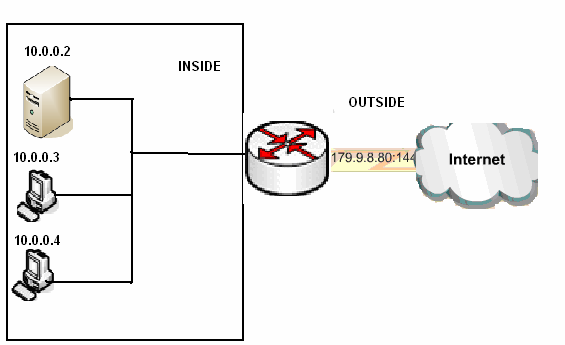
172.16.0.0. ile 172.31.255.255

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 gerekli. 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 biz<im istediğimiz bir ip adresine dönüşüyor, mesela modemin ip adresine.



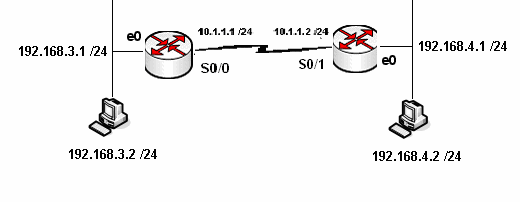
Bu topolojide 10.0.0.0/ 24 networküne ait bilgisayarlar internete erişecekler. Karşımıza “inside” ve “outside” olmak üzere iki kavram çıkıyor. Yine topolojiden anlaşılacağı gibi inside iç networkümüz ve outside da dış network yani internet yada hedef network. Bu kavramlar önemli zira NAT konfigürasyonu 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:

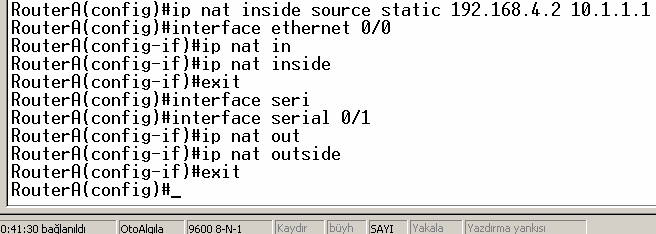
* + Static NAT : Birebir iç bloktaki IP adreslerini dış IP adreslerine çevirme.
  + Dynamic NAT: Bir havuz yaratarak dinamik olarak içerdeki adresleri bu havuzdaki dış IP bloklarıyla eşleme
  + Overloading: Bütün makinaları makina sayısına oranla daha az IP adresiyle dışarıya çıkarma

# NAT Konfigürasyonu

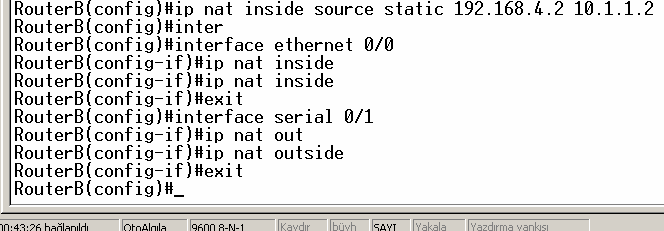


Böyle bir senaryoda Static NAT uygulaması yapacak olursak Routerlar şu şekilde konfigüre

edilmeli.

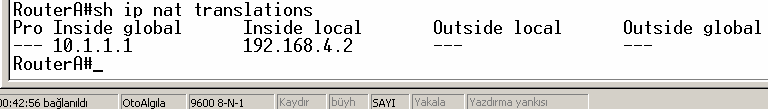


(RouterA için konfigürasyon)



(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 Routerı için iç networkte bulunan 192.168.4.2 ip adresinin Router Serial işnterface’ inden çıktıktan sonra 10.1.1.2 adresine dönüşmesini sağlamış olduk.



(Router A için Ip NAT Translations Tablosu)

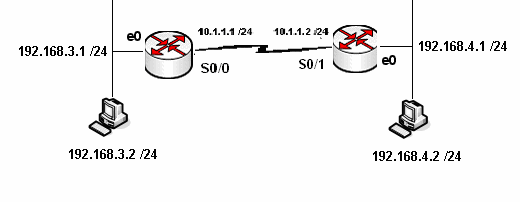
Benzer bir senaryo üzerinde Dynamic NAT uygulayabiliriz. Konfigürasyonda bazı farklılıklar olacaktır. Başta da bahsettiğimiz gibi Dynamic NAT için “outside” tarafında bir ip adres havuzu oluşturulmalı ve “inside” tarafta bir Access list yazılmalıdır.

RouterA#configure terminal

RouterA(config)# ip nat pool AcademyTech 10.1.1.1 10.1.1.5 netmask 255.255.255.0

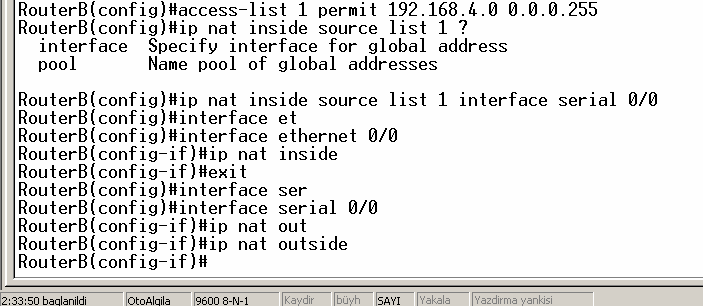
RouterA(config)# access-list 1 permit 192.168.3.0 0.0.0.255 RouterA(config)#ip nat inside source list 1 pool AcademyTech RouterA(config)#interface ethernet 0

RouterA(config-if)#ip nat inside RouterA(config-if)#exit RouterA(config)#interface serial 0 RouterA(config-if)#ip nat outside

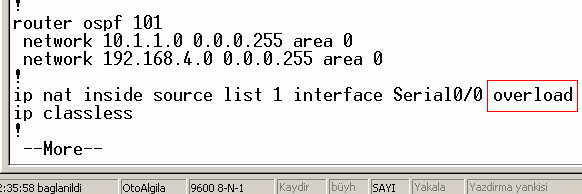


Overload uygulamasında tüm bir network aynı ip adres üzerinden çıkarılabilir. Burada ip adresi belirtmek yerine değişken interface kullanmak gerekir.

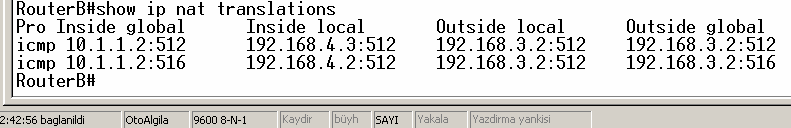
Router B üzerinde NAT konfigürasyonumuz şu şekilde yapılacaktır:



Overload uygulamasında da interface’ lerin “inside” ya da “outside” oldukları belirtilmelidir.



(Overload işlemi Running Konfigürasyondan incelenebilir)



(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 kullanilan 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 olusabiliecek olasi hatalar ile ilgili servis saglayaci sorumluluk kabul etmez, musterinin kendisinin cozum bulmasi gerekir.

**Synchronous:** Seri baglantilarda her iki noktada ki cihazlarin birbirlerine data gonderimi sirasinda hizlarini esitlemeye calistiklari durumdur seklinde anlatilabilir. Bu tarz bir iletimde bizim icin onemli olan nokta upload ve download hizlarinin esit olmasidir.

**Asynchronous:** Dial-Up modemler ornek olarak gosterilebilir. Baglarilarda ki her nokta veri iletim hizlarinin esit oldugunu kabil eder ama esit olmadigi durumlarda esitlemek icin bir calisma yapmaz. Bu durumda upload ve download hizlari da birvirinden farkli olacaktir.

**Data Services Unit/ Channel Services Unit (CSU/DSU) :** DTE olan musteri ekipmaninda clock uretimi saglayacak cihazlardir. WAN aslinda DTE networklerin DCE networkler uzerinden birbirlerine baglanan LAN; lar toplulugudur seklinde tanimlanabilir. Bu durumda ornegin DTE olan Routerlarda data iletimini baslatacak DCE bir cihaza ihtiyac olacaktir, ornegin modem.

Lanoratuar ortamlarinda kullandigimiz ozel kablolar ile DCE cihazlar yerine clock uretimini routerlara yaptiriyorduk. Gercek dunyada uctan uca DTE ve DCE olan kablolar ile baglanti saglamak imkansiz olacagi acik olduguna gore mevcut hatlar uzerinden iletimin saglanabilmesi icin DSU/CSU cihazlara ihtiyac vardir.



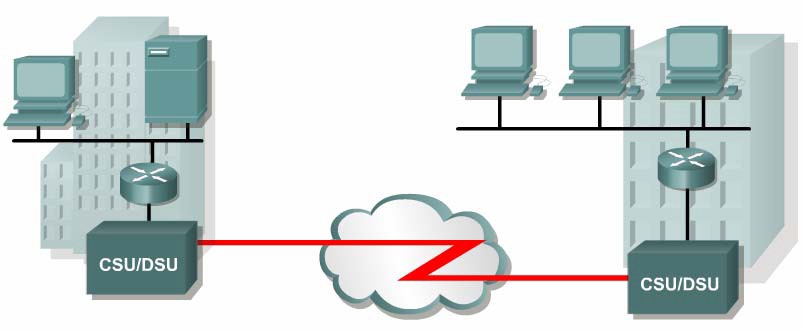
**E1 circuit**

**router**

# WAN Baglantilari

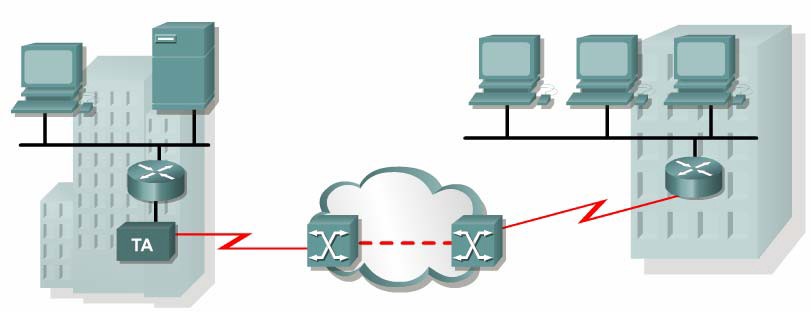
**Kiralik Hatlar (Leased Lines):** Kiralik hatlar tek bir firmaya atanmis, noktadan noktaya baglantinin saglandigi senkron iletim hatlaridir. Senkron iletisim kullanilmasindan dolayi upload ve download hizlari esittir. 45 Mbps’ e kadar hiz desteklemektedir. Baglanti kurulduktan sonra hat devamli aktif durumdadir.

Bu tur baglantilarda daha sonra detayli inceleyecegimiz HDLC, PPP veya SLIP protokolleri kullanilir.



**Devre Anahtarlama (Circuit Switching):** Asenkron iletisim cesididir ve dusuk bant genisligine ihtiyac duyulan durumlarda onerilebilir. Bu baglantilarda artik neredeyse tamamen vazgecilen Dial-ip modemler veya ISDN hatlari kullanilir.

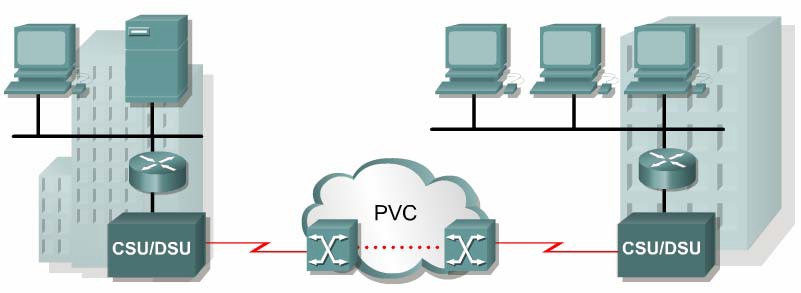
Burada modemler arasinda baglanti kurulduktan sonra hattin surekli aktif kalmasi maliyeti artiracai icin pek tercih edilmeyecektir ama soz gelimi zaten var olan bir hatta yedek olmasi ve o hat koptugunda devreye girip, hat tekrar aktif oldugundan devreden cikmasi saglanabildiginde son derece kullanisli olacaktir.



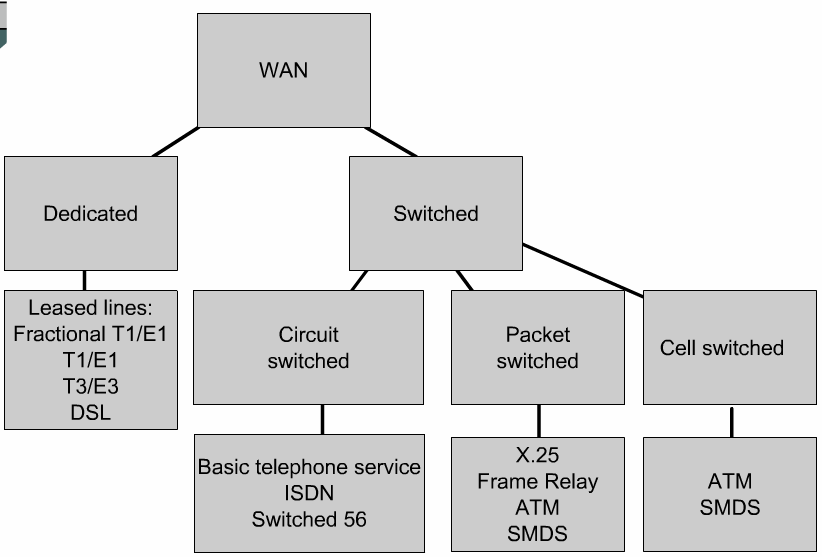
Asenkron iletisim sozkonusu oldugu icin upload ve download hizlari esit degildir. HDLC, PPP ve SLIP protokolleri kullanilabilir.

**Paket Anahtarlama (Paket Switching):** Genis alan aglarinda sabit miktarlarda datanin gonderilmesi durumunda en uygun cozum kiralik harlar olacaktir. Fakat agimizda belirli zaman araliklarinda yuksek datalar gonderilirken bazen cok daha az data gonderimi sozkonus ise bant genisliginin paylasimi esasina gore tarasarlanmis Packet Switching kullanilabilir.

Servis saglayicilar burda bir miktar bant genisligini garanti ederken, garanti etmedigi ama mumkun oldugunda kullanmasina izin verdigi daha yuksek bir bant genisligide saglarlar.



Bu tur baglatilarda detayli inceleyecegimiz Frame-Relay ile birlikte X.25 ve ATM protokolleri kullanilmaktadir.

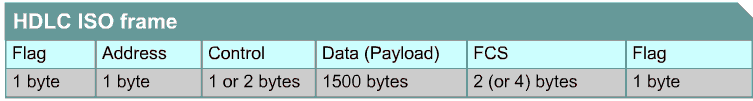


# HDLC

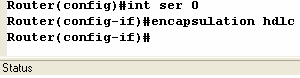
HDLC; IBM tarafından geliştirilmiş standart bit tabanlı bir protokoldür. HDLC (High-Level Data-Link Control) ; data link katman protokolüdür. Cisco’ nun geliştirdiği HDLC ile diğer üretici şirketlerin geliştirdiği HDLC iletişim kuramaz. Bu diger ureticiler icinde gecerlidir. Yani butun HDLC protokollerine ureticisine ozeldir diyebiliriz.

HDLC ; adres alanı , çerçeve alanı, kontrol dizisi alanı (FCS), ve protokol tür alanını içeren çerçevelemeyi tanımlar. HDLC hata düzeltimi aynen Ethernet gibi yapar. HDLC alt bilgisinde FCS alanını kullanır. Alınan çerçevede hatalar oluşmuş ise çerçeveyi düzeltmeden iptal eder.

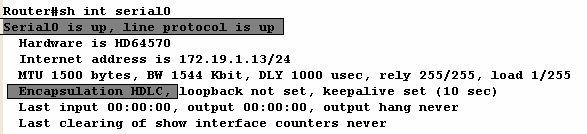
HDLC Çerçevelemesi :



İki router’ı HDLC kullanarak haberleştirmek için aşağıdaki komut satırları kullanılır. Serial Interface de encapsulation ‘ı HDLC olarak ayarlamak :



Yaptığımız konfigürasyonu görmek için show interface serial0 kullandık :

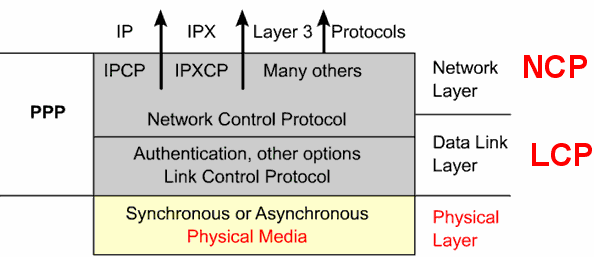


# PPP

PPP HDLC protokolune gore butun uretici firmalarin Routerlari tarafindan desteklendigi 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

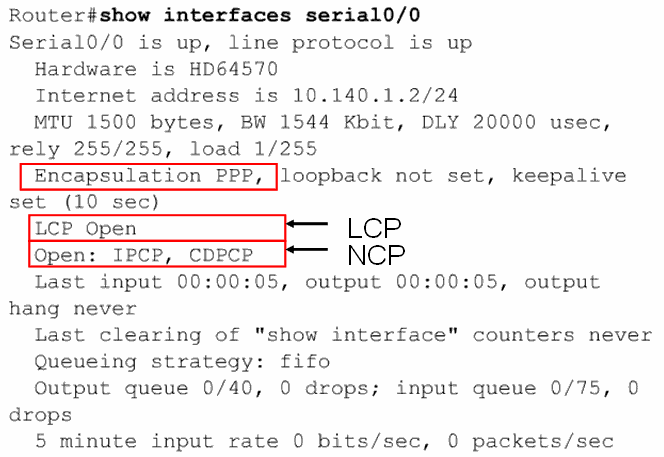


LCP point to point baglantinin saglanmasi icinkullanilirken NCP network katmani protokollerinin konfigurasyonu icin kullanilir.

LCP, Authentication, sikistirma, hata kontrol ve birden fazla yol arasinda load balancing gibi ozellikleri saglar.

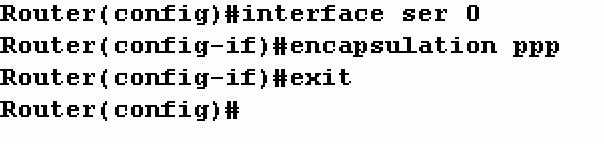
PPP oturumlar opsiyonel olan secimler ile birlikte 5 adimda olusur.

* 1. Link establishment - (LCPs)
  2. Authentication - Optional (LCPs)
  3. Link quality determination - Optional (LCPs)
  4. Network layer protocol configuration (NCPs)
  5. Link termination (LCPs)



# 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.



Ayrıca Router ‘lara “hostname” komutunu kullanarak bir isim verilmelidir.



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 gerekir. Bunun için öncelikle interface moda girilerek “ppp authentication” komutun kullanılır.

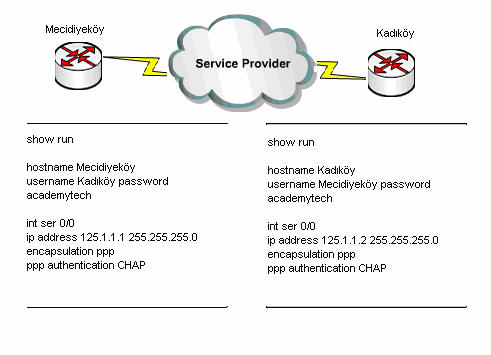
Daha önce bahsettiğimiz PAP yada CHAP metodlarından biri seçilir. Dikkat edilmesi gereken seçilen metodun her iki router da ortak seçilmesidir. Eğer bir router da PAP diğer router da CHAP seçildiyse iletişim kurulamayacaktır.



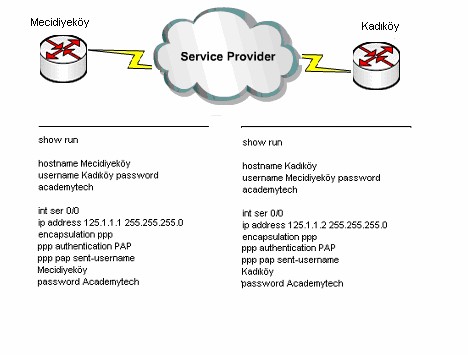
Eğer authentication metodunu PAP seçmiş olsaydık , interface içerisinde PAP ‘ı aktif etmemiz gerekecekti. Çünkü Cisco IOS 11.1 ve sonrasında PAP default olarak disable durumdadır.



# CHAP KONFİGÜRASYONU :

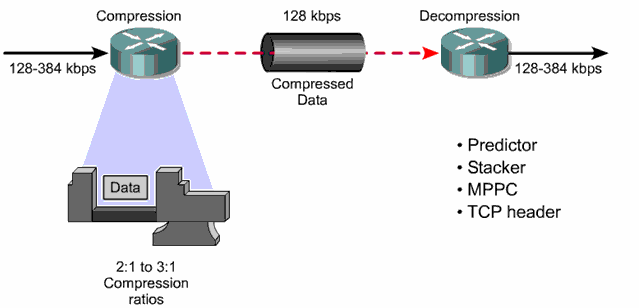


**PAP KONFİGÜRASYONU :**



**PPP Compression**

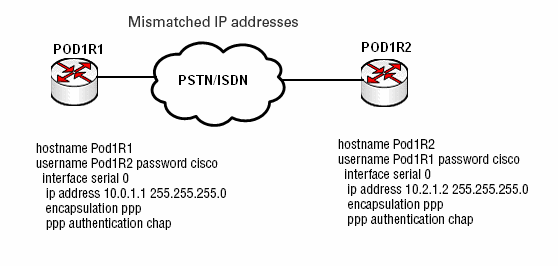
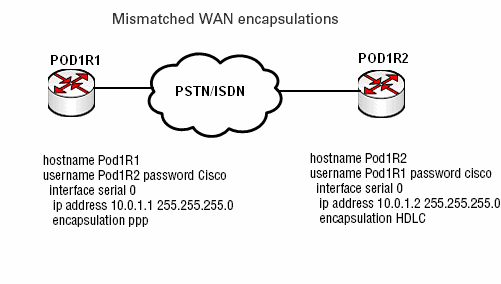
PPP datayi sikistirabilme ozelligi ile dusuk bant genisliginde dahi yukse performans saglayabilmektedir.



4 farkli Compression tipi vardir.

1. Predictor
2. Stacker
3. MMPC
4. Tcp Header Sikistirma

# Hatali PPP Konfigurasyon Ornekleri

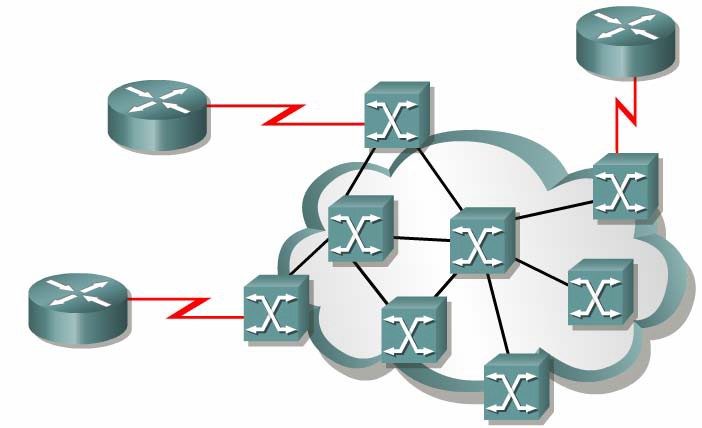


# Frame Relay

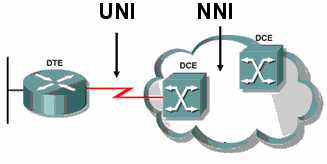
Frame Relay Packet Switching teknolojisyle tumlesik bir WAN servisidir ve OSI referans modelinin Data-Link Katmaninda calisir. Frame Relay HDLN ‘ nin bir alt bileseni olan Link Access Procodure for Frame Relay (LAPF) protokolunu kullanir.

Burada data frameler halinde musterinin DTE cihazlarindan diger nokta veya noktalardaki DTE cihazlarina DCE cihazlar uzeriden tasinir. Burada ki DCE cihazlar ya da DCE network telekom firmalarinin sagladigi network ve cihazlardi, kontrolu ve konfigurasyonu bu firmalar tarafindan yapilir.

Frame Relay networklerinde genellikle 56 Kbps ve 2 Mbit arasinda babnt genislikleri kullanilmaktadir fakat 45 Mbit’e kadar desteklenmektedir.



Frame Relay networklerinde musteri ve servis saglayici arasinda ki baglatiya User-To Network Interface (UNI), birbirinden farkli servis saglayaicilara ait cihazlarin baglandiklari noktalara ise Network-To-Network Interface (NNI) denir.

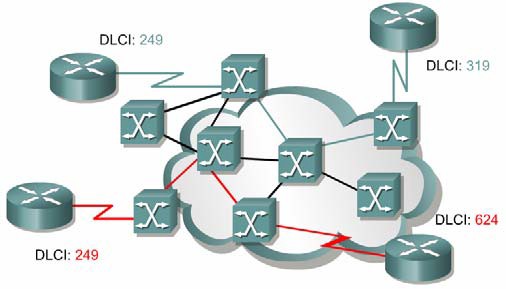


Frame Relay Networkleri olusturulurken servis saglayinin verecegi DLCI numaralirinin tanimlanmasi onemlidir. Çünkü servis saglayici yada telekom bu DLCI numaralarina

kendi switchleri uzerinden yol verecek ve iki nokta arasinda sanal bir devre olusturarak baglantinin kurulmasini saglayacaktir.

Burada bahsettigimiz sanal devreler pek kullanilmayan Switched Virtual Circuits (SVCs) ve Permanent Virtual Circuits (PVCs) olmak uzere ikiye ayrilir.

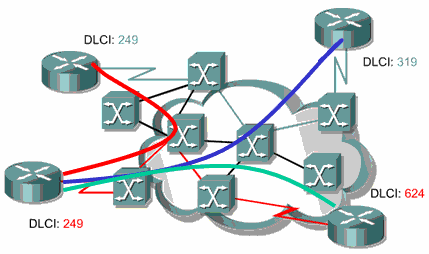
SVCs’ lerde iki nokta arasinda kurulan baglanti icin gecerli olan yol dinamik olarak degismekteyken PVCs’ lerde iki nokta arasinda sabir bir yol tanimi yapilir ve manuel olarak degistirilmedigi surece surekli o yol kullanilir.



Sozgelimi sekilde kirimiz olarak isaretlenen yol soz konusu iki nokta arasindaki PVC’ yi belirtir. Fakat burada SVC ile bir calismadan bahsetseydik mevcut yol yerine alternatif yollardan biri de kullanilabiliyor olacakti.

Cunki SVC’ ler gecici olarak olusturulurlar ve bu baglantilarin olusturulmasi icin bir cevrim (call setup) gerekmektedir.

Frame Relay konusan bir router birden fazla nokta arasindan birebir baglanti yapilmasi gereken durumlarda her nokta icin ayri PVC ler olusturabilir. Merkez nokta uzerinde yapilacak ve her noktaya erisim icin farkli olan DLCI numaralari ile bu mumkun olacak ve o dakikadan itibaren merkez Frame Relay router butun noktalara ayni anda hizmet verebilecektir.



# Frame Relay Headers

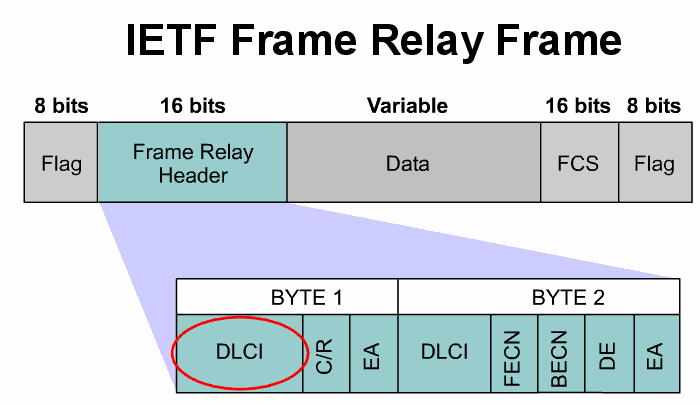
Frame Relay ile konfigure edilmis routerlar iki farkli Frame Relay Header’i desteklerler.

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 onerilen degildir.

IETF ise birden fazla ureticiyi destekler ve framelere Cisco’ nun aksine sadece 2 byte’ lik headerlar ekler.

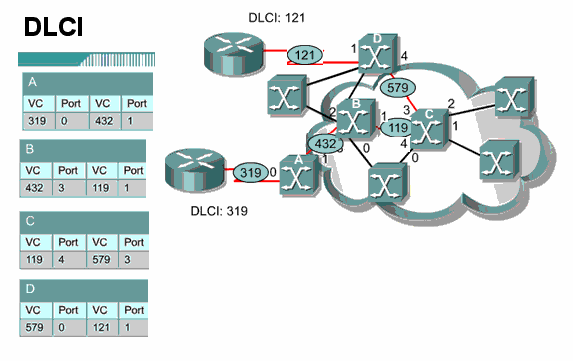
Bu headerlarin iceriginde bizim icin oneli olacak DLCI’ lar vardir.



# DLCI

Data Link Connection Identifier’ in kisaltmasi olan DLCI musteri cihazi ve frame relay switch arasindaki sanal devreyi tanimlaya yarar.

DLCI numaralari servis saglayicalar tarafindan belirlenen mantiksal adreslerdir denebilir. 0-15 ve 1008 – 1023 arasinda ki numaralar ozel amaclar icin ayrildigindan servis saglayicilar tarafindan 16-1007 arasindaki numaralardan secilerek atama yapilir.



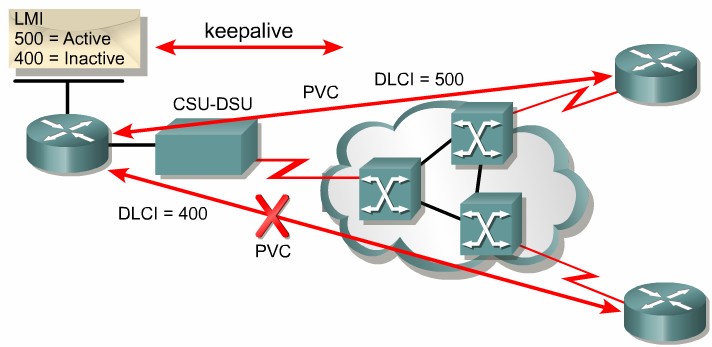
Sekilde A, B, C ve D switchleri uzerinde ki DLCI yonlendirmeleri incelendiginde DLCI mantigi daha iyi anlasilacaktir.

# LMI

Local Management Interface (LMI) DTE cihazlar ve Frame Relay switchler arasindaki signaling standardidir.

Cisco Routerlar 3 cesit LMI Type’i destekler.

1. Cisco
2. Ansi 3. q933a

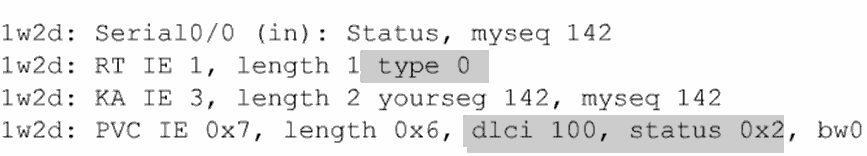


Frame Relay networku ve DTE Router icin LMI type ayni olmadigi takdirde calismayacaktir. Turkiyede kullanilan LMI Type Ansi’ dir. Routerda Cisco IOS 11.2 ve uzeri varsa LMI Type tanimlaya gerek kalmaz, Router Frame Relay networkundeki LMI Type’i algilar.

Frame Relay Switchler konfigure edilen PVC’ lerin durumlarini belirtmek icin LMI’ i kullanirlar. PVC’ ler 3 ayi durumda olabilirler.

* 1. **Active State:** Routerlarin data transferi yapabildigi, baglantinin aktif oldugunun belirtildigi durumdur.
  2. **Inactive State**: Frame Relay switch ile Localimiz arasinda ki baglantinin aktif oldugu ama uzaktaki Router ile uzaktaki Frame Relay switch baglantisinin duzgun calismadigi durumdur.

1. **Deleted State:** CPE ve Frame Relay switch arasinda herhangi bir servisin calismadigi durumdur.



(debug frame-relay lmi)

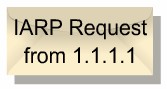
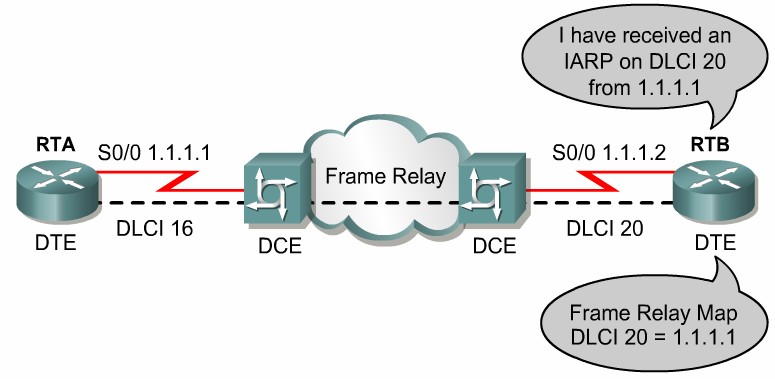
Burada 0x2 aktif durumu gosterir, diger durumlar su sekildedir; 0x0: Inactive

0x4: Deleted

# DLCI Mapping

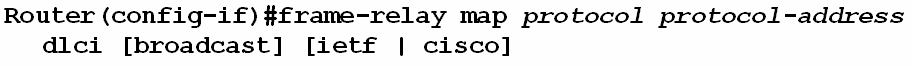
Frame Relay networklerinin konfigurasyonu sirasinda onemli bir adimda servis saglayicilarin Frame Relay switchlerinde yol verdikleri DLCI numaralarinin next hop 3. katman adreslerine map edilmesidir.

Burada mao islemi Dinamik ve static olmak uzere iki sekilde yapilabilir. Static map isleminde frame relay map komutu kullanilir. Dinamik map isleminde ise Inverse ARP protokolu calisir. Burada Inverse ARP her DLCI icin Inverse ARP Request mesaji gonderir ve aldigi cevap ile data- link katman adresi DLCI numarasi ve Network Katmani adresi Next Hop ip adresini map eder.



Kisaca Inverse ARP Lan’ lardaki ARP protokolu gibi calisir.

# Static Map



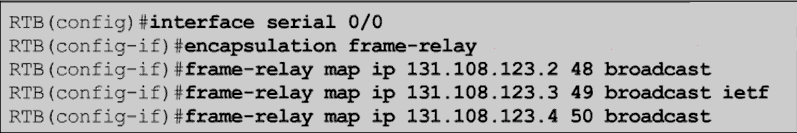
Buradaki ip adresi remote ip adresi DLCI numarasi ise local DLCI numarasidir. Router(config-if)#frame-relay map ip 10.1.1.1 101 broadcast

# Dinamik Map



Buradaki DLCI numarasi local DLCI’dir. Router(config-if)#frame-relay interface dlci 100

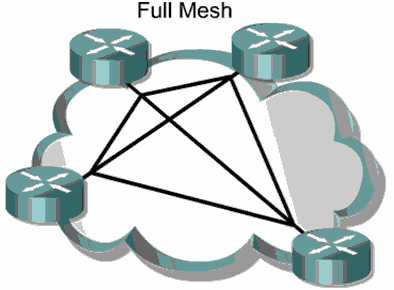
Frame Relay networklerinde Frame Relay encapsulation kullanilir. Cisco ve Ietf olmak uzere 2 ayri standardi vardir, default olarak cisco’ dur. Sistem de Cisco disinda Routerlar var ise RFC 1490 ile tanimlanmis Ietf standardi kullanilmalidir.



Encapsulation Frame Relay secildikten sonra Frame Relay ao komutunda encapsulation secilmeyebilir, bu durumda frame relay encapsulation gecerli olacaktir.

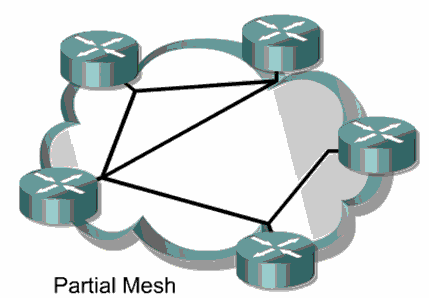
Frame Relay Map satirlarinin ikincisinde ki gibi farkli bir encapsulation secilirse gecerli olan o olacaktir. Ornegimiz de ikinci satir icin gecerli olan encapsulation ietf’dir.

# Frame Relay Topolojileri

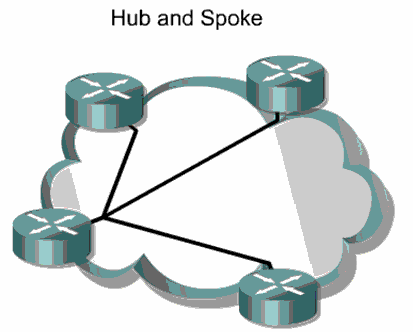


Mesh Topolojide esas olan Tum noktalar arsinda ayri birer PVC olmasidir. Oldukca pahali bir topolojidir. Fakat baglantilardan biri down oldugunda bile bir cok alternatif yoldan hedefe ulasilabilir.

Full Mesh ve Partial Mesh olarak ikiye ayrilir.



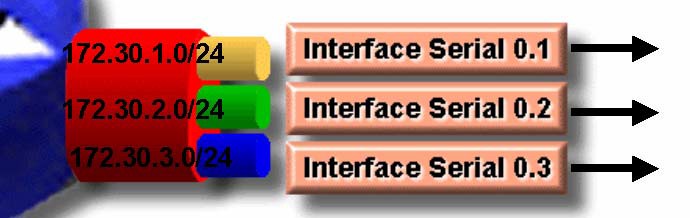
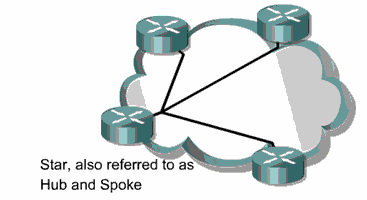
Hun and Spoke Topoloji en cok kullanilan Frame Relay topolojidir ve Star Topoloji olarak da anilir. Bu topoloji genellikle birden fazla uzak networkun merkezi bir routera baglanmasi ile sekillenir.



Burada merkez router multipoint baglanti yada point to poitn baglanti saglayabilir.

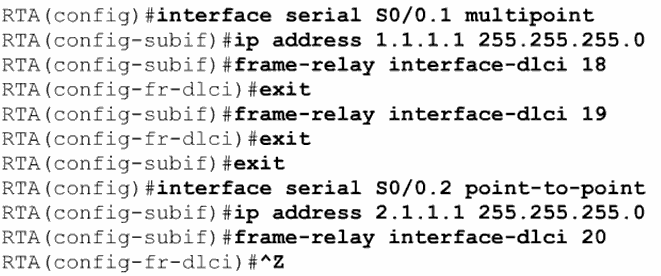
# Frame Relay Sub-Interface Konfigurasyonu

Frame Relay networklerinde sozgelimi merkezde olan bir router birden fazla sayiya sahip subeye hizmet verebili, subeyle baglantiyi saglanabili. Burda interfacein altinda 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 kullanian bu yontem ile ilgili uygulama ilerde yapilacaktir. Simdilik ornek olmasi acisindan cisco.com ‘ dan alionan konfihurasyonu veriyorum.



Hub and Spoke topology Frame Relay networklerinin en cok kullanilan seklidir. Point to multi point veya sub-interfaceler ile point to point olarak tasarlanabilir. Fakat bu topology, Point to multipoint networklerde routing islemi icin Routing Protokoller kullanilmissa Split Horizon kuralindan dolayi sorun yasatacaktir.

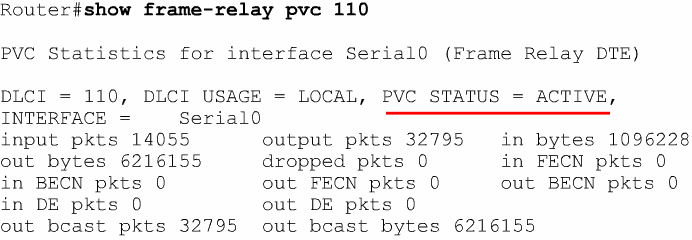
Cunku Split Horizon kurali geregi bir Router aldigi update’ i aldigi interfaceden geri gondermez. Bu durumda Split Horizon kurali devre disi birakilmalidir.

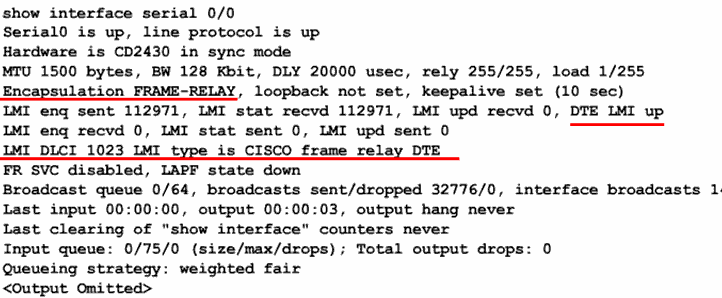
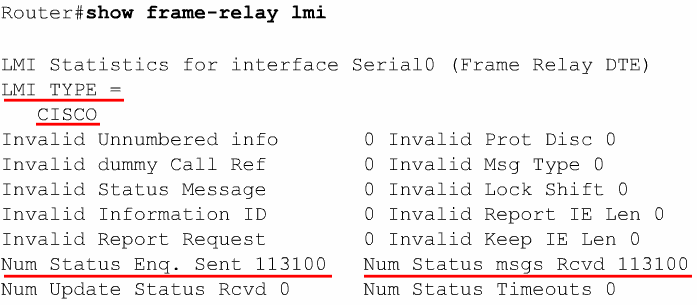
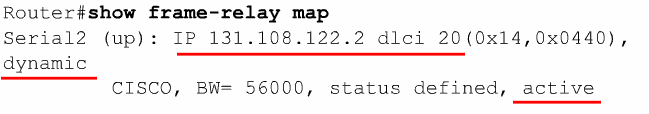


Split Horizon kurali Link State protokolleri ornegin OSPF protokolune etkilemez.

# Frame Relay Show Komutlari:

Asagidaki show komutlari Cisco’ nun CNAP egitimi icin ongordugu program slaytlarindan alinmistir.





# Frame Relay Switch Konfigurasyonu

Laboratur ortaminda Frame Relay ugyulamalri icin Frame Relay Switche ihtiyac vardir. Fakat Frame Relay switch olmadigi durumlarda bir Router Frame – Relay Switch olarak konfigure edilebilir.

Bunun icin Routerlara takilan DTE kablolara DCE kablolalar ile Frame Relay Switch olarak konfigure edilecek Router,a baglanir ve interfacelerine “clock rate” komutu verilir. Burada interfacelere interface type’ in DCE olduguda soylenir.

Ornek:

interface Serial0 no ip address

encapsulation frame-relay no fair-queue

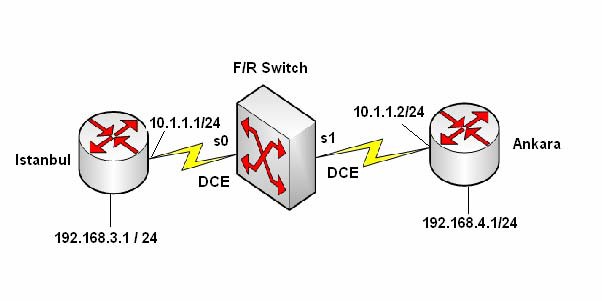
clockrate 64000

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 kullanacagimiz topoloji su sekildedir;



Burada DLCI numaralari İstanbul icin 100, Ankara icin 101`dir ve bir Router laboratuar ortaminda Frame Relay Switch olarak konfigure edilmisitr. Ip adresleri atandiktan sonra, Frame Relay calisma icin;

# İstanbul Routerinda;

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

Konfigurasyonlari yapilmistir.

# 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

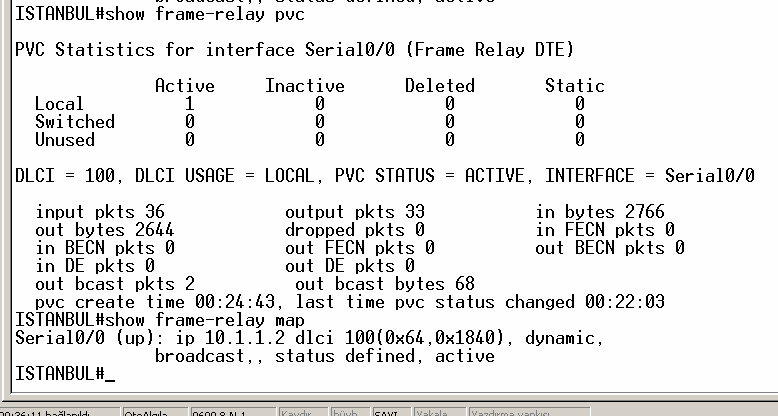
line aux 0

line vty 0 4 login

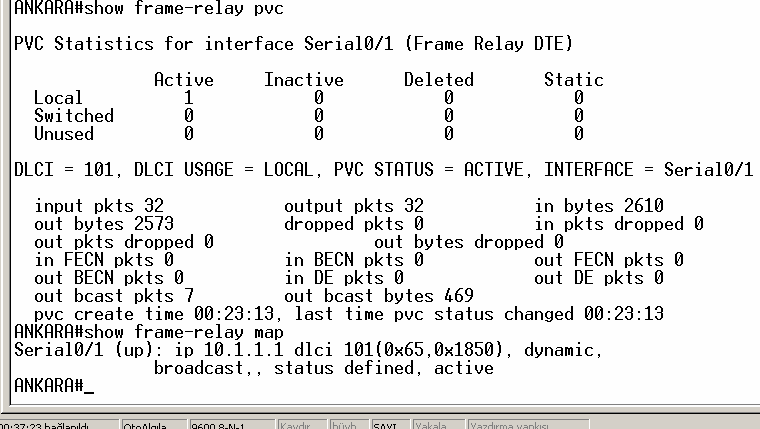
!

end

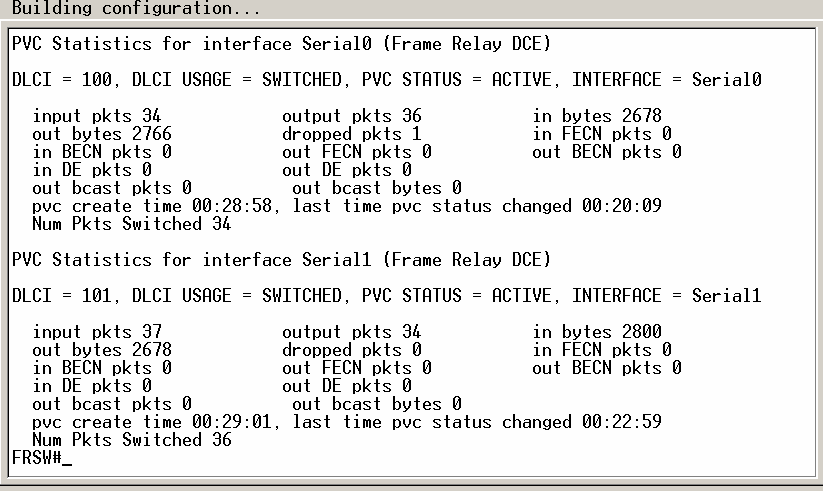
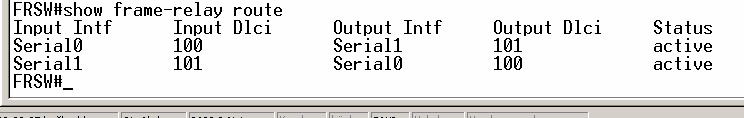
# İstanbul Routeri Frame Relay PCV ve Frame Relay Map



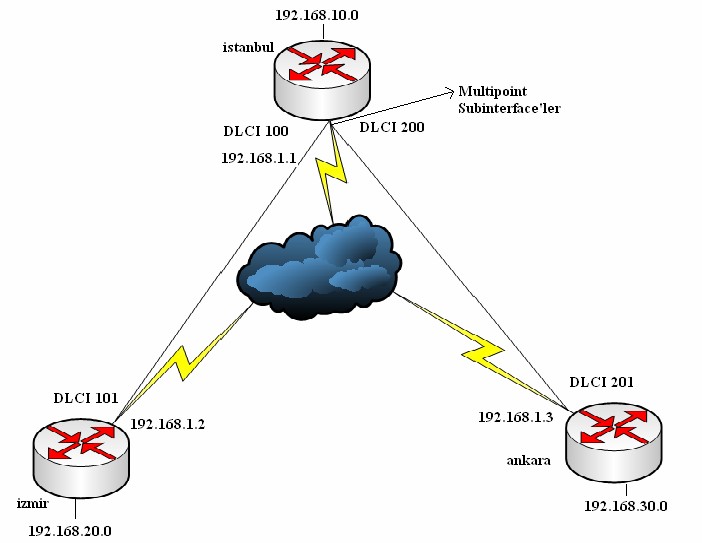
**Ankara Routeri Frame Relay PCV ve Frame Relay Map**



**Frame Relay Switch PCV ve Route**



**Frame Relay Hub and Spoke MultiPoint Konfigurasyonu**



**Routerlarin Konfigurasyon Dosyalari Router Istanbul**

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 |

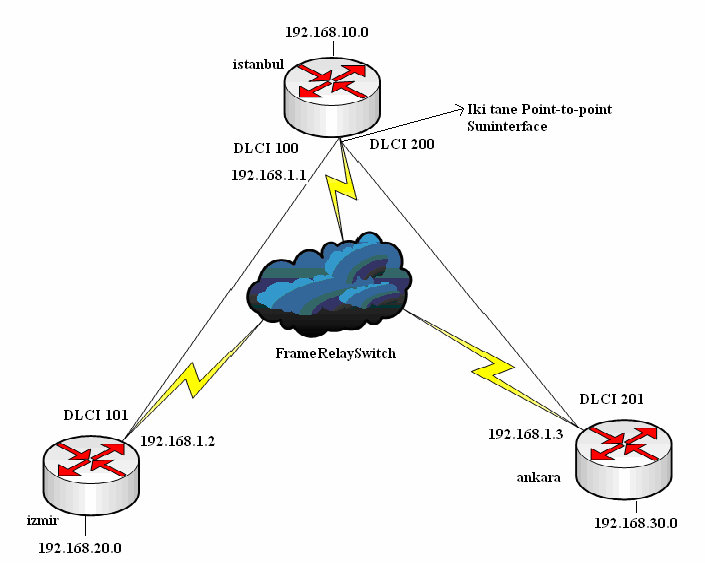
DLCI = 201, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0.1

|  |  |  |
| --- | --- | --- |
| 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 Dosyalari Router Istanbul**

version 12.0

!

hostname Istanbul

!

# 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. Dolayisiyla ISDN ile birlikte PCM’e (Pulse Code Modulation) ihtiyac yoktur.

# ISDN’in Avantajlari:

ISDN dial-ip baglantilardan daha genis bant genisligi saglar. Dial-up maodemlerden daha hizli cevrim saglar.

PPP encapsulation ile birlikte kullanilabilir.

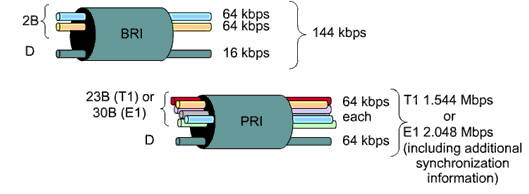
# ISDN’in Dezavantajlari:

ISDN DSL veya kabloya gore daha yavas ve daha pahalidir.

# ISDN Kanallari

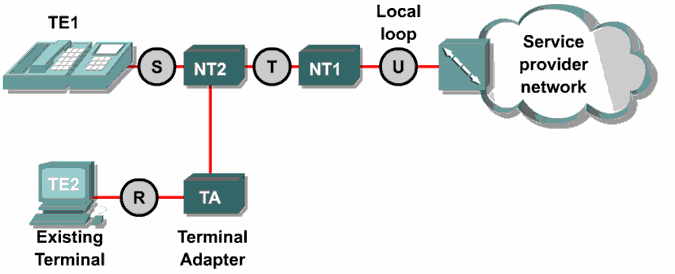
ISDN iki tür hat içerir. **BRI (Basic Rate Interface)** ve **PRI (Primary Rate Interface)**.

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 taşır. BRI hatlarda 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 sınıftaki cihazlar direkt olarak ISDN ağına bağlanabilir.

**TE2 :** Bu sınıftaki cihazlar ISDN standartlarını anlamazlar. ISDN ağına bağlanabilmeleri için bir terminal adaptör (TA) ‘ e ihityaç duyarlar.

**NT1 :** Fiziksel katman özelliklerini tanımlar. Cihazları ISDN ağına bağlar.

**NT2 :** Servis sağlayıcı cihazlardır.

**TA :** T2 kablolamasını T1 kablolamasına dönüştürür.



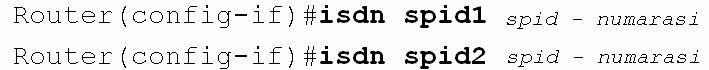
(Bri kart)



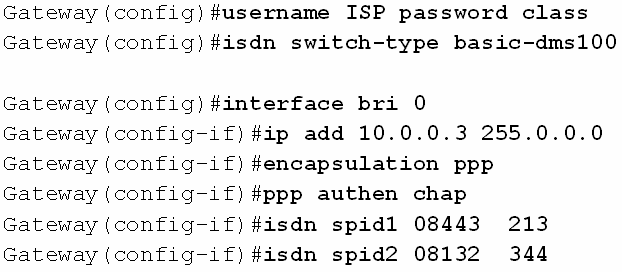
(Terminal Adapter)

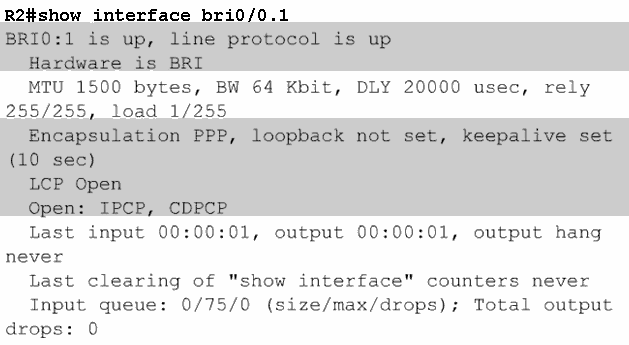
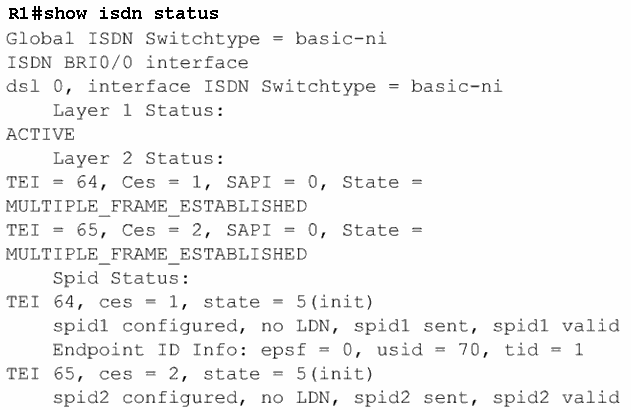
BRI konfigüre ederken her kanal için verilen SPID (Service Profile Identifer ) numarasına ihtiyaç vardır. SPID ‘ler kullandığımız telefon numaralarına benzer. Internet Servis Sağlayıcısından bize verilen SPID numaralarını **“isdn spid1”** ve **“isdn spid2”** komutlarını kullanarak girebiliriz. Ayrıca konfigüre ederken servis sağlayıcının kullandığı switch türünü de router üzerinde belirtmemiz gerekiyor. Kullandığımız router ‘ın ne tür switchlere destek verdiğini görebilmek için **“isdn switch-type ?”** komutu kullanılabilir.

(Turkiyede basic-net3 kullanilmaktadir.)



PPP ve CHAP authentication kullanimi;





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

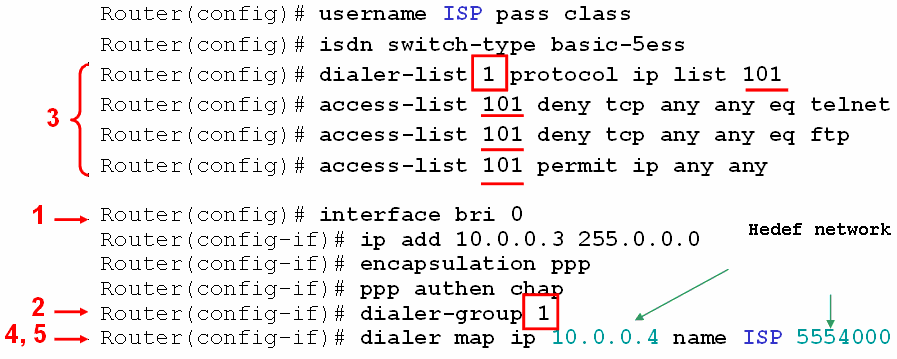
# Router(config)#controller e1 1/0 Router(config-controller)#framing crc4 Router(config-controller)#linecode hdb3

**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**

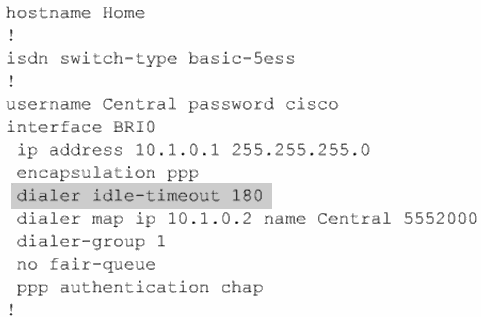
DDR (Dial-on-Demand Router) iki veya daha fazla Cisco router’ ın ISDN dial up bağlantı yapmasını sağlar. Genellikle PSTN veya ISDN kullanılarak gerçekleşen periyodik network bağlantılarında kullanılır. Böylece gerek duyulunca bağlantı gerçekleşir ve ödenecek ücret azalacaktır.

DDR bağlantı konfigürasyonu yapılırken öncelikle bağlantı kurulacak interface içerinde 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.



* 1. Routing table ilgili trafigin bri 0; uzerinden olacagini gosterdigi icin bu interface in konfigurasyonu kontrol edilir.
  2. Router bu interface deki “dialer-group 1” komutundan ayni id numarasina sahip dilaer-list den bu trafige izin verilip verilmeyeceginin arastirilmasi gerektigini anlar.
  3. Bu trafige izin verilip verilmeyecegi ilgili “dialer-list 1 protocol ip list 101” de belirtilen 101 numarali access list ile kararlastirilir.
  4. Trafige izin verilecek ise next hopu bulmak icin dilaer map’ e basvurulur.
  5. Dialer map kullanimdaysa data gonderilir, kullanimda degilse call setup islemi baslar.

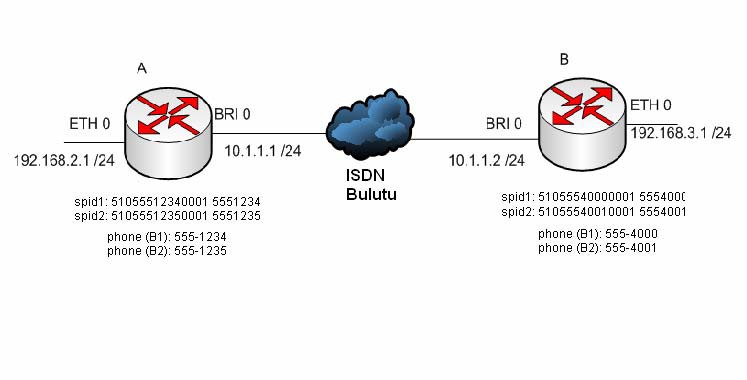
Burada artik bir kez baglanti kurulduktan sonra access list ile belirlenen kriterlere uymayan paketler de gonderilecektir. Fakat sadece bu kriterlere uyan paketler konfigurasyona eklenebilecek iddle-time suresini resetleyecektir.



# 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 icerisinde SPID numaralari ve telefon numaralari kullanimistir.

B kanallarinin her ikisi birlikte kullanilacagi icin her iki kanal icin de telefon numaralari ve SPID numaralari verilmistir.

Her iki Router ‘ da ISDN networkune BRI 0 portlarindan baglanmistir. Konfigurasyon icerisinde ppp authentication chap kullanilmistir.

Yonlendirme icin IGRP konfigurasyonu kullanilmistir ve IGRP icin AS numarasi 100 olarak secilmistir.

# 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 active

RouterA#**show isdn active**

---------------------------------------------------------------------------

ISDN ACTIVE CALLS

---------------------------------------------------------------------------

**Call** Calling **Called Remote** Seconds Seconds Seconds Charges

**Type** Number **Number Name** Used Left Idle 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**](http://www.cisco.com/)[**www.ietf.org**](http://www.ietf.org/)

**CNAP Official Curriculum**