İçindekiler

[1. Ağ (Network) Nedir ? 4](#_Toc140336444)

[2. Devre Anahtarlama (Circuit Switching) ve Paket Anahtarlama (Packet Switching) Nedir ? 5](#_Toc140336445)

[2.1. Devre Anahtarlama (Circuit Switching) 5](#_Toc140336446)

[2.2. Paket Anahtarlama (Packet Switching) 6](#_Toc140336447)

[2.3. Paket Anahtarlama ve Devre Anahtarlamanın Karşılaştırılması 6](#_Toc140336448)

[2.4. Dağıtık Sistem Nedir ? 6](#_Toc140336449)

[2.5. Paketlerin Yönlendirilmesi 6](#_Toc140336450)

[2.6. Yönlendirme Türleri Nelerdir ? 6](#_Toc140336451)

[2.6.1. Statik Yönlendirme 6](#_Toc140336452)

[2.6.2. Dinamik Yönlendirme 6](#_Toc140336453)

[3. Paket Gecikmesi ve Paket Kaybı 7](#_Toc140336454)

[4. Uçtan Uca Gecikme (End to End Delay) 9](#_Toc140336455)

[5. Throughput 10](#_Toc140336456)

[6. İnternet (Ağların Ağı) Nedir ? 11](#_Toc140336457)

[6.1. Büklümlü Çift Kablolar (Twisted Pair Copper Wire) 11](#_Toc140336458)

[6.2. Koaksiyel Kablolar 11](#_Toc140336459)

[6.3. Fiber Optik Kablolar 12](#_Toc140336460)

[7. Router (Yönlendirici) Nedir ? 13](#_Toc140336461)

[7.2. Router ve Switch Arasındaki Fark 13](#_Toc140336462)

[7.3. Switch ve HUB Arasındaki Fark 13](#_Toc140336463)

[8. Routing Türleri 15](#_Toc140336464)

[8.1. Directly Connected Interfaceler 15](#_Toc140336465)

[8.2. Static Route 15](#_Toc140336466)

[8.3. Default Route 16](#_Toc140336467)

[9. LAN (Local Area Network), WAN (Wide Area Network) Nedir ? 17](#_Toc140336468)

[10. Wi-Fi 18](#_Toc140336469)

[11. Erişim Ağları (Access Networks) ve Access Point (Erişim Noktası - AP) Nedir ? 18](#_Toc140336470)

[12. VDSL ve ADSL Nedir ? 18](#_Toc140336471)

[13. Ethernet 18](#_Toc140336472)

[14. VLAN (Virtual Local Area Network) 19](#_Toc140336473)

[15. MAC Adresi 23](#_Toc140336474)

[16. Address Resolution Protocol (ARP) 23](#_Toc140336475)

[17. İstemci (Client) ve Sunucu (Server) Nedir ? (Network Edge) 25](#_Toc140336476)

[18. Protokol Nedir ? 25](#_Toc140336477)

[19. Güvenli Köprü Metni Aktarım Protokolü (HTTPS) 25](#_Toc140336478)

[20. HTTP (Hypertext Transfer Protocol) 25](#_Toc140336479)

[20.1. HTTP neden TCP kullanır? 25](#_Toc140336480)

[20.2. NON-Persistent HTTP (Kalıcı Olmayan Bağlantı) (HTTP 1.0) 26](#_Toc140336481)

[20.3. Persistent HTTP (Kalıcı Olan Bağlantı) (HTTP 1.1) 26](#_Toc140336482)

[21. Katmanlı Mimari 28](#_Toc140336483)

[21.1. Uygulama Katmanı (Application Layer) (TCP kullanır) 28](#_Toc140336484)

[21.2. Ulaşım – İletim – Taşıma Katmanı (Transport Layer) (SIP-WOW-FPS 🡪 TCP or UDP kullanır) 28](#_Toc140336485)

[21.3. Ağ-İnternet-IP Katmanı (Network Layer) 28](#_Toc140336486)

[21.4. Bağlantı Katmanı (Link Layer – Data Link Layer – Network Interface) 28](#_Toc140336487)

[21.5. Fiziksel Katman (Physical Layer) 28](#_Toc140336488)

[22. OSI Modeli (Open Systems Interconnection) 29](#_Toc140336489)

[22.1. Mesaj, Segment, Datagram ve Frame 29](#_Toc140336490)

[22.2. OSI Katmanlarının Görevleri Nelerdir? 30](#_Toc140336491)

[22.3. TCP/IP ve OSI Modeli Farkı Nedir? 30](#_Toc140336492)

[23. Ağ Saldırıları 31](#_Toc140336493)

[23.2. Sosyal Mühendislik Atakları 32](#_Toc140336494)

[23.2.1. Pretexting 32](#_Toc140336495)

[23.2.2. Phishing 32](#_Toc140336496)

[23.2.3. Vishing 32](#_Toc140336497)

[23.3. Güvenlik Araçları ve Uygulamaları 33](#_Toc140336498)

[24. Cookies 34](#_Toc140336499)

[25. Web Tampon Belleği (Web Caches) (Vekil Sunucu – Proxy Server) 34](#_Toc140336500)

[26. Conditional GET (Koşullu GET) 35](#_Toc140336501)

[27. HTTP/2 35](#_Toc140336502)

[28. E-Mail 36](#_Toc140336503)

[29. SMTP (Simple Mail Transfer Protocol) 36](#_Toc140336504)

[29.1. HTTP ile SMTP farkı 37](#_Toc140336505)

[29.2. Mail (E-Posta) Mesaj Formatı 37](#_Toc140336506)

[30. DNS (Domain Name System) 38](#_Toc140336507)

[30.1. DNS Sunucuları 38](#_Toc140336508)

[30.2. Local (Yerel) DNS Sunucuları 39](#_Toc140336509)

[30.3. DNS Kayıt Türleri 40](#_Toc140336510)

[30.4. DNS Protokol (Protocol) Mesaj 40](#_Toc140336511)

[31. Peert to Peer (P2P) Mimarisi (256 kb) 41](#_Toc140336512)

[32. Video Streaming and CDN 42](#_Toc140336513)

[33. Soket Programlama 42](#_Toc140336514)

[34. TCP / IP (Transmission Control Protocol) (Internet Protocol) 42](#_Toc140336515)

[35. TCP/IP ve OSI Modeli Arasındaki Farklılık Nedir? 42](#_Toc140336516)

[36. Transport Hizmetleri ve Protokoller(Taşıma – İletim – Ulaşım Katmanı) 43](#_Toc140336517)

[37. Multiplexing / Demultiplexing 43](#_Toc140336518)

[37.1. Bağlantısız (Connectionless) Demultiplexing 44](#_Toc140336519)

[37.2. Bağlantılı (Connection-Oriented) Demultiplexing 44](#_Toc140336520)

[38. URL-URN-URI 45](#_Toc140336521)

[39. UDP (User Data Protocol) 46](#_Toc140336522)

[39.1. Connectionless 46](#_Toc140336523)

[39.2. UDP Checksum 46](#_Toc140336524)

[40. TCP (Transmission Control Protocol) 47](#_Toc140336525)

[40.1. TCP Yaşam Döngüsü 49](#_Toc140336526)

[40.2. UDP ve TCP Segment Formatı 50](#_Toc140336527)

[40.3. TCP Fast Retransmit 52](#_Toc140336528)

[40.4. TCP Flow Control (Akış Kontrolü) 52](#_Toc140336529)

[40.5. TCP ve UDP Karşılaştırılması 53](#_Toc140336530)

[41. Güvenilir Veri Transferi Prensipleri (Reliable Data Transfer - RDT) 54](#_Toc140336531)

[42. Stop and Wait Protocol 54](#_Toc140336532)

[42.1. Rdt 1.0: Güvenilir Kanal Kullanarak Güvenilir Transfer 54](#_Toc140336533)

[42.2. Rdt 2.0: Bit Hatası Olan Kanal ile Çalışma 54](#_Toc140336534)

[42.3. Rdt 2.0’daki Yenilikler (Rdt 1.0’a göre): 54](#_Toc140336535)

[42.4. Rdt 2.0’da Karşılaşılan Problemler 54](#_Toc140336536)

[42.5. Rdt 2.1 55](#_Toc140336537)

[42.6. Rdt 2.2 NACK Kullanılmayan Protokol 55](#_Toc140336538)

[42.7. Rdt 3.0: Kanallarda Hata ve Kayıp 55](#_Toc140336539)

[43. Pipeline (Boru Hattı) Mimarisi 55](#_Toc140336540)

[44. Go-Back-N (Go Back N) Protokolü 56](#_Toc140336541)

[45. Selective Repeat (Seçici Tekrar) Protokolü 56](#_Toc140336542)

[46. Tıkanıklık – Sıkışıklık Kontrolü (Congestion) 58](#_Toc140336543)

[46.1. Tıkanıklığı Önleme Politikaları 58](#_Toc140336544)

[46.2. Tıkanıklık Denetimi 58](#_Toc140336545)

[46.3. Tıkanıklığın Nedenleri ve Maliyeti 58](#_Toc140336546)

[46.4. Tıkanıklık Denetimi Yaklaşımları 60](#_Toc140336547)

[46.5. TCP Tıkanıklık Denetimi 60](#_Toc140336548)

[46.6. Yavaş Başlatma (Slow Start) 60](#_Toc140336549)

[46.7. Tıkanıklıktan Kaçınma (Congestion Avoidance) 61](#_Toc140336550)

[46.8. Hızlı Toparlanma 61](#_Toc140336551)

[47. QUIC Protokolü (HTTP/3 Protokolü) 61](#_Toc140336552)

[48. Ağ Katmanı (Network Layer) 62](#_Toc140336553)

[48.1. Ağ Katmanı Servis Modelleri 63](#_Toc140336554)

[48.2. Datagram Ağlar 63](#_Toc140336555)

[49. IP (Internet Protocol) Nedir ve Nasıl Çalışır 64](#_Toc140336556)

[50. IPv4 Datagram Formatı 65](#_Toc140336557)

[51. IPv6 Datagram Formatı 66](#_Toc140336558)

[52. IP Datagram Fragmentation 67](#_Toc140336559)

[53. IPv4 Adresleme 67](#_Toc140336560)

[53.1. Statik ve Dinamik IPv4 Adresleme 69](#_Toc140336561)

[54. IPv6 Adresleme 71](#_Toc140336562)

[54.1. IPv6 Adres Gösterimi 72](#_Toc140336563)

[54.2. IPv4’ten IPv6’ya Geçiş 72](#_Toc140336564)

[54.3. Loopback IP Address 74](#_Toc140336565)

[55. Yazılım Tanımlı Ağ (SDN - Software Defined Networking) Nedir? 74](#_Toc140336566)

[56. Alt Ağ Maskesi (Subnet Mask) 75](#_Toc140336567)

[56.1. Subnetting (Alt Ağlara Bölme) 78](#_Toc140336568)

[56.1.1. IP Subnetting (Binary (İkili) Sayı Sistemini Kullanarak Bulma) 79](#_Toc140336569)

[57. Gateway 81](#_Toc140336570)

[58. ICMP (Internet Control Message Protocol – İnternet Kontrol Mesaj Protokolü) 82](#_Toc140336571)

[58.1. ICMP Paket Yapısı 82](#_Toc140336572)

[58.2. Mesaj Tipinin Alabileceği İfadeler 82](#_Toc140336573)

[59. TELNET 83](#_Toc140336574)

[59.1. Telnet'teki Güvenlik Sorunları 83](#_Toc140336575)

[60. Kablosuz Ağ Ayarları 84](#_Toc140336576)

[61. ISS 84](#_Toc140336577)

[62. Bulut Türleri (Veri Depolama Türleri) 84](#_Toc140336578)

[63. Ağ Sanallaştırma 84](#_Toc140336579)

[64. Komut Satırı Network Kontrol İşlemleri 86](#_Toc140336580)

[65. WIRESHARK 89](#_Toc140336581)

[66. Fiziksel Bağlantı Çeşitleri 91](#_Toc140336582)

[67. TERİMLER SÖZLÜĞÜ 92](#_Toc140336583)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. Ağ (Network) Nedir ?  * **Network** birden fazla cihazın herhangi bir protokol ile birbiriyle haberleşmesi sonucu oluşturdukları yapıdır. **Network hızını**; kullanılan cihazın ethernet hızı, network kablosunun türü ve switch portlarının hızı belirler. * Network’ün elemanları arasında; [host](#hostNedir), packet switch ve communication link’ler vardır.   Yani birden fazla cihaz var ve bunlar birbirleriyle iletişim halindedir. Bunları yönetebilmek için arada router’lar ve protokol’ler vardır.  Bir ev taşıdığımızı düşünelim Konya’daki evimiz uç sistem Ankara’daki taşınacağımız olan evde uç sistem olarak tanımlanmaktadır. Eşyalarımızı birden fazla tıra yükediğimizi düşünürsek bu tırlarıda paket olarak isimlendirmekteyiz. Tırların takip ettikleri yollar, köprüler, kavşaklar communication links olarak bilinirken rotasını belirleyen harita, asistan rehber gibi kişilerde router (yönlendirici) olarak adlandırılmaktadır.   2. Devre Anahtarlama (Circuit Switching) ve Paket Anahtarlama (Packet Switching) Nedir ?2.1. Devre Anahtarlama (Circuit Switching)  * Paketler sadece kendisi için ayrılmış olan yoldan ilerlerler, o yol sadece o pakete özeldir, başkası kullanamaz.   O yolu birkaç kullanıcıya tahsis etmek için FDM (frekans) ve TDM (zaman) kullanılır.   * Özetle, her bağlantı n tane devreye sahiptir ve eşzamanlı n tane bağlantıyı destekler. * Devre anahtarlamalı ağlar veri iletişimine başlamadan önce yol kurulumu yapar.  |  |  | | --- | --- | | Avantajları  Veri iletişimi paket anahtarlamalı ağlara göre hızlıdır. | Dezavantajları  Kaynakları verimsiz kullanmasıdır. |   **Egzersiz**  Host A ile Host B arasında devre anahtarlamalı ağ üzerinden 640.000 bit gönderilecektir. Ağdaki tüm bağlantılar 24 slot ile TDM kullanmaktadır ve bit oranı 1.536Mbps. Uçtan uca devre kurulumu 0.5s’de yapılmaktadır. Dosyanın gönderilmesi için gereken süre ne kadardır?  *Her devrenin iletim oranı = 1.536Mbps / 24 = 64kbps*  *Dosyanın gönderim süresi = 640.000 / 64kbps = 10s*  *Toplam süre = Devre kurulum süresi + Gönderim süresi*  *= 0.5s + 10s = 10.5s* 2.2. Paket Anahtarlama (Packet Switching)  * Bir paket belirli bir yoldan giderken diğer paket trafik oluşturmamak için başka bir yoldan hedefe gidebilir. * Paketlerin her birisi, bağlantılar ve paket anahtarlar aracılığıyla [kaynak ile hedef arasında iletilir.](#DagitikSistemNedir) * Günümüzde paket anahtar olarak [**yönlendirici (router)**](#routerYonlendiriciNedir)ve **anahtar (link-layer switch)** kullanılır. Anahtarlar [erişim ağları (access networks)’larında](#accessNetworksErisimAglari), router’lar ise ağın temel kısmında (network core) kullanılır. * Paket anahtarların çoğu [depola ve yönlendir (Store and Forward)](#transmissionDelayIletimGecikmesi)şeklinde iletim yaparlar. Yani, bir paketin tamamı alınır sonra yönlendirilir. * ARPANET internet tarihindeki ilk paket-anahtarlamalı (packet-switching) ağ oldu.   **Store and Forward gecikmesi = olur. (L=paket boyutu, R=iletim hızı)**   |  |  | | --- | --- | | **Avantajları:**  1. Birden fazla kullanıcı ortama paket koyabilir.  2. Paket yollarında sorun çıkarsa diğer yollar tercih edilebilir. | **Dezavantajları:**  1. Store and Forward Delay  2. Paket Gecikmesi  3. Kuyruk Gecikmesi |  2.3. Paket Anahtarlama ve Devre Anahtarlamanın Karşılaştırılması  * **Devre anahtarlamada** kaynak paylaşımı yapılmaz ve iletişim süresince bir kaynak kullanılmasa bile tek bir iletişime ayrılmıştır.   **Paket anahtarlamada**, kaynak paylaşımı yapılır ancak uçtan uca gecikme öngörülebilir.   * **Devre anahtarlamada**, veri gönderen kullanıcıya daha fazla kaynak ayırma, veri göndermeyen kullanıcıya kaynak ayırmama istatistiksel çoklama ile yapılır.  2.4. [Dağıtık Sistem Nedir ?](#paketAnahtarlama) Ağdaki bilgisayarların belirli bir hedefe ulaşmak için birbirleriyle etkileşim halinde olmasına **dağıtık sistem** denir. Dağıtık sistemi çalıştıran bilgisayar programına **dağıtık program** denir. Bu tür programları yazma işlemine **dağıtık programlama** adı verilir. 2.5. Paketlerin Yönlendirilmesi İnternette her paket gideceği hedefin adresini başlık bilgisi kısmında bulundurur. Bir router gelen her paketin adresini alır ve kendisinden sonra gideceği router’ı belirler. Her router gelen paketleri yönlendirmek için bir yönlendirme tablosuna **(forwarding table)** sahiptir ve hedef adres ile çıkış bağlantılarını eşleştirir. Yönlendirme tablosunu otomatik olarak güncellemek için internette çok sayıda yönlendirme protokolü vardır. Her router kendi yönlendirme tablosunda komşuluklarına ve hedef adreslere ilişkin bilgileri saklar. 2.6. Yönlendirme Türleri Nelerdir ? Yönlendiricinin yönlendirme tablolarını nasıl oluşturduğuna dayanan iki farklı yönlendirme türü vardır: 2.6.1. Statik Yönlendirme Statik yönlendirmede, bir ağ yöneticisi ağ yollarını manuel olarak yapılandırmak ve seçmek için statik tablolar kullanır. Bu yönlendirme tekniğinin statik yapısı, ağ tıkanıklığı gibi dezavantajlara sahiptir. 2.6.2. Dinamik Yönlendirme Dinamik yönlendirmede, yönlendiriciler gerçek ağ koşullarına göre çalışma zamanında yönlendirme tabloları oluşturur ve günceller. Dinamik yönlendirme tablosunu oluşturan, koruyan ve güncelleyen bir dizi kural olan dinamik bir yönlendirme protokolü kullanarak kaynaktan hedefe giden en hızlı yolu bulmaya çalışırlar.  Dinamik yönlendirmenin en büyük avantajı; trafik hacmi, **bant genişliği** **(bandwidth: bir ortamın veri taşıma kapasitesidir)** ve ağ hatası dahil olmak üzere değişen ağ koşullarına uyum sağlamasıdır. |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 3. Paket Gecikmesi ve Paket Kaybı  * Bir paket bir [host’tan](#hostNedir) (kaynak/source) yola çıkar, çok sayıda router’dan geçer ve en sonunda bir başka host’ta (hedef/destination) yolculuğu biter.   Eğerki gönderilen paketin, yönlendirildiği port o anda kullanılıyorsa **kuyruğa** alınarak bekletilir. Bu durumda **kuyruk gecikmesi (queuing delay)** oluşur. Diğer bir deyişle **buffer’da** harcadığı süredir. Kuyruk gecikmesi her paket için farklı olur. Boş bir kuyruğa 10 paket gelirse, ilk paket gecikme olmadan gönderilir, ancak sonuncu pakete kadar her pakette gecikme artarak devam eder.  Kuyruk gecikmesinin olup olmaması, paketlerin kuyruğa geliş trafiği, bağlantının iletim oranı ve trafiğin karakteristiğine (periyodik veya burst) bağlıdır.   |  |  |  |  | | --- | --- | --- | --- | | Paketlerin kuyruğa geliş oranı  a (paket/s) | İletim oranı ve kuyruktan çıkan bit sayısı  (Gönderim hızı)  R (bps) | | Paketlerin boyutu (bit)  L | | **Kuyruğa saniyede gelen bit sayısı:** **L\*a bps** | | **Trafik yoğunluğu:** | | |   Eğer L\*a/R > 1 olursa kuyruğa gelen bit sayısı kuyruktan ayrılan bit sayısından büyüktür. Eğer kuyruktaki paket sayısı sınırsız olursa **kuyruk gecikme** süresi sonsuza doğru **artarak** devam eder. Eğer sınırlı paket sayısı olursa bir süre sonra gelen paketler atılmaya başlar.  Eğer L\*a/R ≤ 1 olursa kuyruğa gelen bit sayısı kuyruktan ayrılan bit sayısından küçüktür. Eğer paketler periyodik olarak L/R saniye aralıklarla gelirse **kuyruk gecikmesi olmaz**. Eğer paketler periyodik ancak [burst](#burst) şeklinde gelirse, örneğin N paket (L/R) saniye aralıklarla gelirse, ilk pakette gecikme olmaz sonrakilerde gecikme artarak devam eder.  **Trafik yoğunluğu** 1’e yaklaştıkça kuyruk gecikmesi artar. Ancak trafik yoğunluğu 1’e yaklaşırken paket gecikmesi sonsuza doğru artış göstermez. Paket, tamamen dolu bir kuyruğa gelirse sıraya girmek için yer olmadığından gelen paket atılır ve **paket kaybı (packet loss)** olur. **Trafik yoğunluğu arttıkça paket kayıp oranı artar.**     * **İşlem gecikmesi (processing delay),** paketin başlık bilgisine bakılarak çıkış portunun belirlenmesi için geçen süredir. Processing delay, bit seviyesinde hata kontrolü için geçen süreyi de içerir. * **İletim gecikmesi (transmission delay),** bir paketin [tamamının iletim ortamına verilebilmesi](#StoreAndForward) için geçen süredir. Bir paketin toplam boyutu **L bit** ve router A ile router B arasındaki bağlantının iletim oranı **R bps** ise, **transmission delay, saniye** olacaktır.   L = 10 Kbits, R = 100 Mbps 0,1 msec   * **Yayılma gecikmesi (propagation delay)**, bir bitin bağlantının bir ucundan diğer ucuna ulaşması için geçen süredir. Propagation delay, sinyalin iletim ortamındaki **mesafe (uzunluk) ile yayılım (iletim) hızına bağlıdır** ve  **(distance (m), speed (m/s))** şeklinde gösterilir. * **Nodal processing delay,** ilk düğümün sıraya alınmadan önce paketin işlenmesi için geçen süredir. Hangi hedeflere gideceğine karar verir. * Paket kaybının birçok olası nedeni vardır: * Arızalı bir ethernet kablosu. * Bilgisayarınızdaki ağ arabirim denetleyicisi (NIC) ile ilgili sorunlar. * Ağ yoğunluğu. * Örnek-1   Aşağıdaki örnekte 10 araç bulunmaktadır her bir araç 1 bite tekabül etmektedir.  Gişeler (toll booth) bilgisayar ağlarındaki router benzeri görev yapar.  Gişeler arası mesafe de **link** olarak adlandırılmaktadır.  Bir araç için gişede geçen süre 12s ve araçların hızı 100km/saat ise, 10 aracın yola çıkması için geçen toplam süre (transmission delay) 10 x 12s = 120s olur.  Tüm araçların ikinci gişeye ulaşması için geçen süre (propagation delay) ise 120 saniye + 1 saat = 62 dakika olur. (100km/(100km/saat) = 1 saat)  Eğerki 10 arabadan 1 tanesi bile router'a yani gişelere ulaşamazsa bu bozuk paket anlamına gelmektedir ve iletim gerçekleşmez.  **NOT:** **İlk bit (paket) hiç gecikme olmadan ikinci router’a ulaşır çünkü önünde bekleyen bit (paket) yok.** |

|  |
| --- |
| 4. Uçtan Uca Gecikme (End to End Delay)  * Uçtan uca gecikme, kaynak ile hedef arasındaki yol üzerinde bulunan router sayısına bağlıdır. * Formülde yer alan **N,** yol üzerindeki router sayısını göstermektedir. * End to end throught, minimum kavramına göre çalışır hangisi minimumsa o geçerli olur. * **Traceroute programı** **uçtan uca gecikme süresini elde etmek için kullanılır**. Bir hedef host adı girilerek traceroute program çalıştırıldığında, kaynakta çalışan program hedefe özel paketler gönderir. Bu paketler birçok router’dan geçerek hedefe doğru gider. Her router aldığı özel paket için kaynağa kendi adını ve adresini içeren mesaj gönderir.   Kaynak ile hedef arasında N-1 router varsa, kaynak N tane özel paketi ağa gönderir. Her paket 1 ile N arasında numaralandırılır. N hedef içindir.   * Traceroute programı ile **her router için 3 paket** gönderilir. * Aşağıda [TTL (Time to Live)](#TTL_timeToLive) değeri 30 [hops](#HOPS) olarak (varsayılan değer) belirlenmiştir. Bu da demek oluyorki gönderilen paket 30 hops’tan sonra düşecek demektir.      * 7. router'dan sonra [RTT](#RTT_roundTripTime) değeri 34ms’e çıkıyor bu da demek oluyorki bu 7 ile 9. router arasındaki uzaklık artıyor. * **"Request timed out."** dese bile bir sonraki router'ı görebiliyorsak bir şekilde isteğin öteki router'a ulaştığını anlayabiliriz. Böyle bir şey olmasının sebebini o şekilde configure edilmiş diyerek açıklayabiliriz. * Eğerki **"\* \* \*"** şeklinde bir şey görüyorsak ya router o şekilde configure edilmiş demektir ya da siber saldırıyı önlemek için yapılmış yani atılan isteklere cevap vermeyecek şekilde configure edilmiş demektir. * **"tracert -h 4 google.com"** dersek 4 adet hops gönderir. * Eğerki destination'a varana kadar 5 router varsa ve biz 3 hop gönderdiysek ilk 3 router'a ulaşacak bizim kaynağımız ama 5. router'a ulaşamadığı için bu 3 router arasında döngüye girmiş olacaktır.    5. Throughput  * Bilgisayar ağlarında, **saniyede aktarılan toplam veri miktarı (throughput)** sınırlıdır. * Host A ile Host B arasında bir dosya transferi yapıldığını varsayalım.   İki host arasında veri aktarımı yapılırken, kurulan yol üzerinde **en düşük** **transmission** oranına sahip link iletişimin **troughtput** değerini belirler.   * **Anlık (Instantaneous) Throughput:** Herhangi bir anda Host B’nin dosyayı alma oranıdır (bps).   **Ortalama (Average) Throughput:** Dosyanın toplam boyutunun (F) toplam transfer süresine (T) oranıdır ve F/T şeklinde gösterilir.   * Sunucudan istemciye bir dosya transfer edilsin. server ile router arasındaki iletim oranını, router ile client arasındaki iletim oranını göstermektedir. [Server ağa bps oranında bit gönderir ancak router bps oranından fazla bit gönderemez.](#darbogaz)      * Eğer < ise, throughput değeri olur [(darboğaz)](#darbogaz).   Eğer > ise, throughput değeri olur ve router içinde bekleyen bit sayısı sürekli artar.  Eğer server ile client arasında N tane link varsa, throughput değeri min { , , ... , } kadar olur.  F bit boyutundaki dosyanın transfer süresi F/min{, … , } olur.     * R ağın, core (çekirdek) kısmındaki iletim oranıdır.   Eğer R >> veya R >> ise ağda tıkanıklık olmaz. (>> birkaç yüz kat)     * Örnek-1   = 2Mbps, = 1Mbps R=5Mbps oranına sahiptir.  10 download aynı anda yapılırsa her iletişim için iletim oranı 5Mbps/10 = 500kbps olur. Uçtan uca throughput değeri 500kbps olur.  Bu durumda throughput değeri erişim ağları tarafından değil, ağdaki çekirdek kısım tarafından belirlenir. |

|  |
| --- |
| 6. İnternet (Ağların Ağı) Nedir ?  * Uç sistemler, **iletim bağlantıları (communication links)** ve **paket anahtarlar (packet switches)** aracılığıyla birbirlerine bağlanırlar. * Bir paketin gönderici uç sistem ile alıcı uç sistem arasında kullandığı iletim bağlantılarına **route** veya **path (yol)** denir. * İletim ortamları iki gruba ayrılır:  1. **Guided Media (Kablolu),** fiber optik kablo, büklümlü çift kablo, koaksiyel kablo. 2. **Unguided Media (Kablosuz),** atmosfer ve uzay boşluğu.  * Uç sistemden başka bir uç sisteme veri göndermek için **message segmentation (segment)** oluşturulur.   **Message segment,** gönderilecek olan verinin paketlere bölünerek gönderilmesidir. Örneğin 100mb’lık bir veri göndereceğiz biz bu veriyi 10 mb’lık paketlere bölerek message segmentation oluşturmuş oluruz.  Gönderici uç sistem tarafından segmente başlık bilgileri eklenerek paket (packet) oluşturulur. Alıcı uç sistem gelen paketleri birleştirerek orijinal veriyi elde eder. 6.1. Büklümlü Çift Kablolar (Twisted Pair Copper Wire) Gürültünün etkisini azaltmak için teller birbirine sarılarak oluşturulur. Dış kısmında metal kılıf olmayanlar, Unshielded Twisted Pair (UTP) olarak adlandırılır ve en yaygın kullanılandır. Dış kısmına kılıf kullanılanlar, Shielded Twisted Pair (STP) olarak adlandırılır ve maliyeti daha yüksektir. Günümüzde CAT 6 yaygın kullanılmaktadır.     6.2. Koaksiyel Kablolar Özel yalıtkan malzemesi ve kılıfı sayesinde daha yüksek bit oranı sağlarlar. Koaksiyel kablolar TV sistemlerinde çok yaygındır. Çok sayıda uç sistem doğrudan koaksiyel kabloya bağlanabilir.   6.3. Fiber Optik Kablolar Fiber optik kablolar görünen ışık palsleriyle iletişim yaparlar. Bir fiber optik >100Gbps bit oranı sağlayabilir ve 100km’den fazla mesafeye zayıflamadan iletilebilir. Özelikle kıtalar arası (long-haul) iletim ortamı olarak kullanılır. |

|  |
| --- |
| 7. Router (Yönlendirici) Nedir ?  * **Yönlendirici,** bilgi işlem cihazlarını ve ağları diğer ağlara bağlayan bir ağ cihazıdır. Yönlendiricinin her bir ucu yani arayüzüı ayrı bir LAN’ı temsil etmektedir. Yönlendiricinin bir arayüzüne atanan IPv4 adresi, o arayüzün baktığı yerel ağa bağlı tüm cihazlar için varsayılan ağ geçidi adresidir. * Aşağıdaki örnekte "Wireless Router" uç cihazlara IP ataması yaparak DHCP server gibi davranmış. Wireless Router, ISP’den örneğin Türk Telekom’dan IP talebinde bulunmuş.   Özetle kablosuz yönlendirici genellikle yerel ağdaki bilgisayarlar için IP adresleme bilgileri sağlar.  İnternet’e bağlanan bir yönlendirici de genellikle internet sağlayıcısından DHCP’yi kullanarak adres bilgilerini alır.     * Yönlendiricilerin üç görevi vardır.  1. **Yol Belirleme**   Yönlendirici, en iyi yolu bulmaya çalışır.   1. **Veri İletme**   Yönlendirici, hedefine ulaşmak için verileri seçilen yoldaki bir sonraki cihaza iletir.   1. **Yük Dengeleme**   Bazen yönlendirici birden çok farklı yol kullanarak aynı veri paketinin kopyalarını gönderebilir. Bunu, veri kayıplarını azaltmak ve yedek oluşturmak için yapar.   * **Layer2 (L2) switch,** ağ cihazlarının iletişimini sağlamak için kullanılan cihazdır. Bu cihazlar, Ethernet veya Wi-Fi gibi fiziksel katman protokollerindeki veri iletimi işlemlerini gerçekleştirebilir. Bir layer2 switch, **bir ağda birden fazla cihazın birbirleriyle iletişim kurmasına izin verir**.  7.2. Router ve Switch Arasındaki Fark **Router** iki LAN, iki WAN ya da LAN veya WAN gibi farklı ağları birbirine bağlar. Ancak, **switch** bir ağ oluşturmak için birden fazla cihazı birbirine bağlar. 7.3. Switch ve HUB Arasındaki Fark **Hub,** veri paketini alır daha sonra ağda bulunan tüm portlara gönderir, **hedef cihaz paketi alır, diğer cihazlar ise paketi duymazdan gelirler.**  **Switch,** **gelen veri paketini yalnızca veri paketine ihtiyaç duyan bağlantı noktasına gönderir**. Switch’in avantajı, akıllı yönlendirme yetenegi ile Hub’tan çok daha yüksek hızdadır.   * **NAT,** bir şirket içinde kullanılan private IP adreslerini internette yönlendirilebilen public bir adrese dönüştürmek için kullanılır.   Bir public adres, bağlantı noktası numaralarını kullanarak birçok private IP adresi arasında paylaşılabilir. Yani aynı IP’ye bağlı birden fazla cihaz port numaraları farklı olması hasebiyle paylaşılabilir.    **Yukarıdaki örnekte Host A, Host B’ye sadece ping atmak istemektedir. Burada neler yaşandığını adım adım inceleyelim.**   1. Internet Control Message Protocol (ICMP), bir echo request oluşturur. 2. ICMP, bu request’i Internet Protokolüne (IP) gönderir, böylece bir paket oluşturulur. Bir paket, en az bir IP kaynak adresi ve IP hedef adresi içerir. 3. Paket oluşturulunca, IP hedef adresinin lokalde mi yoksa uzak network’te mi olduğunu belirler. Kendi networkünde mi, değil mi? 4. IP, bunun uzak bir istek olduğunu belirleyince, paketin uzak ağa route edilebilmesi için varsayılan ağ geçidine yani bilinmeyen adreslerin yönlendirildiği default gateway’e gönderir. 5. Host A üzerinde default gateway 192.168.5.1 olarak ayarlanmış. Bu durumda öncelikle Fa0/0’a ulaşması gerekiyor. Bu paketin yani LAN içerisinde iletişim gerekiyor bunun için de FA0/0 in MAC adresine ulaşması gerekiyor. (192.168.5.1’e ulaşması için.) Host A bir broadcast başlatır ve MAC adresi öğrenmek ister. 6. ARP broadcast den sonra Fa0/0’ın donanım adresi öğrenilir ve paketler 192.168.5.1’in MAC adresine doğru iletilir. Layer 2 katmanında olacağı için bu iletim Frameler şeklindedir. 7. Router’a ulaşan bir frameler, IP katmanına yollanır. 8. IP katmanı bu paketi alıp hedef IP adresini kontrol eder, eşleşen bir bilgi olup olmadığını kontrol etmek için routing tablosuna bakar. 9. Eğer routing tablosunda hedef IP adresine ait bir veri yoksa paket atılacak(discard) ve ping yollayan makineye yani Host A’ya network erişilemez mesajını dönecektir. 10. Şayet router, routing tablosunda hedef IP adresi için kayıt bulursa, hedef IP adresin bağlı olduğu FA0/1’e gönderir. 11. Router Fa0/1 den paketi 10.0.0.5’e yollayacağını bilir fakat haberleşme tekrar LAN içerisinde olacağı için öncelikle yine donanım yani MAC adresini öğrenmesi gerekir, bu yüzden FA0/1 den bir ARP sorgusu 10.0.0.5 in MAC adresini öğrenmek için gönderilir. 12. MAC adresi cevabı FA0/1 e döndükten sonra paket HostB’nin MAC adresine iletilir. 13. ICMP isteği Host B’ye geldikten sonra, Host B bu echo requeste cevap olarak echo reply oluşturur ve geri Host A’ya göndermek için işlem başlatır. 14. Aynı döngüler bu sefer Host B tarafından Host A’ya doğru işlenerek, Host A’ya ulaşır ve Host A bütün bu işlemlerin sonunda tek bir ! işareti görür ve aynı icmp'den 4 tane daha yollar.  8. Routing Türleri8.1. Directly Connected Interfaceler Önceki örneğimizde gördüğümüz gibi bir router üzerindeki interfaceler’de bulunan IP networklerini birbirleri ile haberleştiriyordu bunun sebebi bir router’da bir interface aktif ederek IP verildiğinde, router bu verilen IP’nin network olarak route tablosuna ekler. Örnekten yola çıkarsak. **C              192.168.5.0/24       is             directly    connected,              FastEthernet0/0 C              10.0.0.0/8               is             directly    connected,              FastEthernet0/1**  Route tablosunda connected şekilde bu networkler görünecektir. 8.2. Static Route Yaptığımız örneklerde 1 adet router’a bağlı LAN netwokleri arasında haberleşmeyi inceledik. Peki router sayısı artığında nasıl bir yol izleyeceğiz?  Routerlar kendi üzerindeki interfaceleri routing tablosuna ekliyordu. Kendi bilmediği network’e gitmek için de static route ekleyerek kendinde olmadığını fakat erişebilmesi için şuraya gitmesi gerektiğini belirtmiş olur.  **R1 192.168.1.0/24 ve 192.168.12.0/24 R2 192.168.2.0/24 ve 192.168.12.0/24E’e nasıl erişeceklerini bilmekteler.**  Fark ettiyseniz burada R1 ve R2’nin ortak bir network’e sahip olduğunu göreceksiniz. Yani R1 ve R2, 192.168.12.0 network’ü üzerinden birbirlerine erişebilirler. Routerlar farklı networkleri IP katmanında haberleştireceği için kendileri haberleştiği zaman üzerindeki networkleri de haberleştirebilirler.    H1’in H2’ye ulaşırken öncelikle bilmediği bir network olduğundan paketi direk default gateway’ine atacak yani R1’e. R1 192.168.2.0/24 networkünü bilmediğinden normalde packeti discard edecektir. Fakat biz ona öğretirsek bu network için packetleri nereye atması gerektiğini görevi gereği yönlendirecektir. R1’e 192.168.2.0/24 gitmesi için R2’ye gitmesini söyleyeceğiz. Burada packetleri göndereceğimiz nokta R1’e en uzak, R2’ye en yakın yerdir, burası da 192.168.12.2 IP’sine sahip gi0/2 interface’idir. Aynı şekilde R2 router’ına da 192.168.1.0/24 networküne olan istekler için 192.168.12.1’e göndermesini belirteceğiz böylece giden paket aynı zamanda geri dönebilsin.  Static route yapılırken hedef adresin network bilgisi yani hem IP hem de subnet mask bilgisi, gönderileceği IP bilgisi yani;  192.168.2.0            255.255.255.0                        192.168.12.2          şeklinde route yazılmaktadır. (hedef network)   (hedef network mask)          (yönlendirileceği IP)    Router sayımızı 3’e çıkarırsak nasıl bir işlem yapacağımızı görelim.  172.16.1.0 networkünden 172.16.2.0 networküne erişeceğiz. İki router da birbirine bağlı networklerle ilgilenmiştik. Burada arada farklı bir geçiş router’ı mevcut. Routing yaparken yaptığınız router en uzak ulaşacağınız networke en yakın yere route edeceğinizi söylemiştik.  R1 için route yazarsak, **172.16.2.0 255.255.255.0 192.168.12.2** paketleri R2’ye atmış olduk.  R2 için route yazarsak, **172.16.2.0 255.255.255.0 192.168.23.3** R2’ye de networkün R3’te olduğunu öğrettik.  Bu durumda haberleşme sağlanmayacaktır çünkü haberleşme kuralımızda bir packet gönderiliyorsa geri gelmesi gerekiyor. Mesela herhangi bir işlem yapmazsa paket R1’in Fa0/0’ından çıkacak hedef adresini takip ederek R3’ün Fa0/0’ına gelecek. 172.16.2.0’a ulaşacak daha sonra, 172.16.2.0’daki cihaz 172.16.1.0 networkünü bilmediğinden default gateway’ine atacak. R3 bu paketi alacak ve geri dönmesi gereken IP’ye baktığında 172.16.1.0 networkünü tanımadığını görecek ve paketi discard edecek. Gördüğünüz gibi bir haberleşme tam olabilmesi için her router’a mevcut networkleri öğretmemiz gerekecek. Şimdi geri dönüş için routeları yazalım.  R3 için route yazarsak; **172.16.1.0 255.255.255.0 192.168.23.2** paketleri R2’ye göndereceğiz  R22’de ise; **172.16.1.0 255.255.255.0 192.168.12.1** şeklinde routeları yazarak haberleşmeyi sağlamış olduk. Fark ettiyseniz R2’nin diğer iki LAN networkleri ile bir ilgisi yok ama üzerinden geçen paketleri yönlendirebilmesi için bu bilgiyi ona verdik. 8.3. Default Route PC’lerde olan default gateway kavramına benzemektedir. PC’ler bilmediği networkleri default gateway’e atardı yani kendinden başka networkler. Routerlar’da da bu default route yazılarak yapılır. Bir router da bir adrese doğru default route yazarsanız, routing tablosunda görmediği networkler için diğer bütün paketleri oraya yönlendirecektir.  Default route’ın yazılımı şu şekildedir;  0.0.0.0 0.0.0.0 192.168.1.1  burada ne olursa olsun paketleri **192.168.1.1’e** at demek istiyoruz.  Routing’de bilinmesi gereken en önemli şeylerden biri de mask küçüldükçe önem derecesi daha da artar.  Örnek vererek incelersek;  Router 1 üzerinde 2 adet route mevcut.  **PC0 192.168.10.13** IP’sine sahip bir hedef bilgisayara gitmek isterse, daha küçük maskli route olan R3’ü tercih edecektir. Bu çok önemli bir ayrıntıdır. Bir ağdaki en önemli routing bilgisinin /32 mask olduğunu böylece söyleyebiliriz.   9. LAN (Local Area Network), WAN (Wide Area Network) Nedir ?  * **LAN (yerel alan ağları),** belirli bir coğrafi alanla sınırlandırılmış bir grup bağlı cihazdır. Bina veya daire içi gibi "Ethernet" teknolojisinin kullanıldığı yerlerde kullanılırlar.   İnternet kafeler LAN’ı en iyi şekilde tanımlar. LAN‘larda temel amaç aynı yapı içerisinde kullanılan bilgisayarların kendi donanımlarını paylaşmasını sağlayarak zamandan tasarruf edip bilginin hızlı bir şekilde elde edilmesini hedefler.   * **WAN (geniş alan ağları),** farklı bölgelerde oluşturulan LAN’ların birbirine bağlanması sonucu oluşturulan geniş ve büyük bilgisayar ağlarıdır.   Örneğin, ülke genelinde birden fazla yerde faaliyet gösteren büyük kuruluşlar ve şirketler, her bir konum için ayrı LAN'lara ihtiyaç duyacak ve daha sonra bir WAN oluşturmak için diğer LAN'lara bağlanacaktır.  WAN geniş bir alana dağıtıldığından, genellikle birden çok yönlendirici ve anahtar gerektirir.   * Aşağıdaki örnekte 2 adet LAN ağı bulunmaktadır. Çünkü router’ın her bir yüzü yani her bir ucu 2 farklı bağlantıya gitmekte.     Yukarıdaki gibi uzak segmentte ana bilgisayarlara sahip olmanın **avantajları**:   1. Daha büyük, daha karmaşık ağlar için uygundur. 2. Yayın etki alanlarını böler ve trafiği azaltır. 3. Her segmentte performansı arttırabilir. 4. Cihazları diğer yerel ağ segmentlerindekilere görünmez hale getirir. 5. Daha fazla güvenlik sağlaybilir.   **Dezavantajları** ise şunlardır:   1. Router ihtiyacını doğurur. Router kullanımı segmentler arasındaki trafiği yavaşlatır böylelikle daha fazla karmaşık ve maliyetli bir hal alır.  * Aşağıdaki örnekte router olmadığı için bir adet LAN ağı, bir tane broadcast domain bulunmaktadır.     Yukarıdaki gibi tek bir yerel segmentin **avantajları** şunlardır:   1. Daha basit ağlar için uygundur. 2. Daha az karmaşıklık ve daha düşük ağ maliyeti vardır. 3. Cihazların diğer cihazlar tarafından görülebilmesini sağlar. 4. Daha hızlı veri aktarımı ve doğrudan iletişim imkanı sunar.   **Dezavantajları** ise şunlardır:   1. H1’in H4 ile haberleşmek istediğini varsayalım H4’ü ARP tablosunda ararken bulamadığını ve. H4’ün de IP adresinin 192.168.1.4 olduğunu varsayalım.   H1, 192.168.1.4 adresli cihazın MAC adresi nedir diye broadcast yaptığı yani genel yayın yaptığı takdirde ortamda router olmadığı switch1’in her iki ucundaki arayüzlerden yayın başlar ve switch2’ye de bu yayın ulaşır böylelikle yalnızca H1-H2-H3-H4-H5’in duyması yeterliyken H1-H2-H3-H4-H5-H6-H7-H8-H9-H10’da duymuş olacaktır.  Bu da ağ trafiğini arttıracak ve yavaşlığa sebep olacaktır. Ayrıca güvenliği sağlamakta zorlaşacaktır. 10. Wi-Fi  * Günümüzde iki tür kablosuz internet erişimi yapılmaktadır. **Wireless LAN (WLAN)** ile erişimde, kullanıcılar paketleri bir [**erişim noktasına (Access Point (AP))**](#accessPoint) iletirler.   **Wide-Area Wireless Access Network** ile erişimde, kullanıcılar kendilerinden kilometrelerce uzakta bulunan baz istasyonuna (Base Station (BS)) bağlanırlar.   * İnternet altyapısı üzerinden veri gönderimi için **Application Programming Interface (API)** hizmeti kullanır. İnternet API veri gönderimi için gerekli kurallardan oluşur.   Mektupla haberleşmede, zarfın üzerine adres yazılarak posta kutusuna konulmasına kadar olan işler kişi (internet uygulaması) tarafından yapılır, mektubun alınıp karşı tarafa iletilmesi posta şirketi (internet API) tarafından yapılır.   11. Erişim Ağları (Access Networks) ve Access Point (Erişim Noktası - AP) Nedir ?  * **Erişim ağları,** uç sistemi **ilk yönlendiriciye (edge router)** bağlayan fiziksel bağlantıları (kablo-kablosuz) sağlarlar. * **Access point**, ethernet bağlantısını kablosuz bir ağa dönüştürür ve bu sayede cihazlar arasında kablosuz bağlantı kurabilmelerine olanak tanır. Access point, bir Wi-Fi router gibi çalışır ve kullanıcıların internete bağlanmasına, dosya paylaşmasına ve diğer ağ hizmetlerine erişmelerine yardımcı olur.  12. VDSL ve ADSL Nedir ?  * **VDSL,** internet kullanıcılarına daha hızlı ve kesintisiz hizmet sunabilmek adına özel olarak üretilen bir teknoloji. VDSL’in sunduğu internet kalitesi, santrale yaklaştıkça artıyor. * **ADSL** için kısaca, yüksek bant genişliğinde, hızlı şekilde veri aktarımını sağlayan teknoloji denilebilir. ADSL sayesinde çoklu işlemleri aynı anda yapabilirsiniz. * ADSL-DSL **asimetriktir** yani upload ve download hızları farklıdır. * Hem ADSL hem VDSL kullanıcılara bakır kablolar ile ulaştırılıyor ancak *VDSL’de ek olarak* *fiber kablolar* da yer almaktadır. Fiber kablolar ile kişilere sağlanan internet hızında bir kayıp yaşanması engelleniyor. * **ADSL modem ile VDSL hizmeti alamayız.**  13. Ethernet  * Bir şirket veya üniversitede **Local Area Network (LAN)** bir uç router’a bağlanmak için kullanılır.   Ethernet ağlara bağlantı büklümlü çift kabloyla yapılır.     * Örneğin, 802.3 1OOBASE-T   + 100 Mbps hız sunar.   + BASE, taban bandı iletimi anlamına gelir.   + T, bükümlü çift kablo standartları olan kablo tipini ifade eder. * Ethernet’in minimum paket boyutu 64, maximum paket boyutu 1518 olur. Hesaplamaya "preamble and SFD" katılmaz çünkü preamble mesajın nerede başladığını ifade eder.    14. VLAN (Virtual Local Area Network)  * VLAN, yerel bir alan ağ yani LAN üzerindeki kaynakların ve ağ kullanıcılarının gruplandırılmasını sağlayan teknolojidir. VLAN sisteminde belli bir algoritmaya göre sınıflandırma yapılır ve gruplanan veriler switch üzerindeki portlara atanır. VLAN kullanıldığında her VLAN yalnızca kendi broadcast’ini alır. Bu sayede broadcast trafiği azaltılır ve bant genişliği artırılır. Farklı VLAN türleri ihtiyaca yönelik çözümler üretmenize yardımcı olur.   Bir anahtar içerisinde bir VLAN kurduğunuzu hayal edin. Her şerit ayrılır ve paralel olarak çalışmaya başlar. Şeritlerdeki farklı bağlantı noktaları VLAN üyeliklerini temsil eder. Peki, VLAN ID nedir? VID olarak da bilinen VLAN ID, yerel ağdaki çerçevelerin hangi VLAN’a ait olduğunu belirten bir işarettir.   * Sanal bir ağı simgeleyen VLAN, fiziksel LAN’lardan meydana gelir. Güvenli bir ağ sistemi kurmayı sağlayan bu teknoloji, büyük ölçekli firmaların da sıklıkla tercih ettiği özellikler sunar. Şirket ağlarında güvenliğin sağlanmasına önemli katkılarda bulunan VLAN, aynı zamanda ağ trafiğini dengelemeye yardımcı olur. Bilgi teknolojileri personelleri, ağları bölümlere ayırarak bunların her birini ayrı ayrı sınıflandırır. Daha sonra firmadaki farklı departmanlara özgü VLAN ağları tanımlanır. VLAN, bu ağların kontrol edilmesini ve güvenli bir şekilde kullanılmasını sağlar. Çeşitli fiziksel ağlar hem kablosuz bağlantı hem de Ethernet ile kullanılabilen sanal LAN’ları destekler.   VLAN, geniş ve kapsamlı bir ağ içerisinde oluşturulan küçük bir mantıksal bölümdür ve sanal bir alandan meydana gelir. Yerel ağın birden fazla anahtara yayılması, LAN grubuyla uyum sağlandığı sürece sorun yaratmaz. Yönettiğiniz anahtarlar sanal LAN oluşturmanızı sağlar. İşletmenizin ihtiyaçlarına uygun olarak Berqnet SASE veya firewall ürünlerini tercih ederek VLAN teknolojisini verimli bir şekilde kullanabilirsiniz. Berqnet ile ağ güvenliği ve ağ yönetimi konusunda çağın ihtiyaçlarına hitap eden avantajlardan yararlanabilirsiniz.  Mesela şirketinizde bir misafir ağı oluşturmak istiyorsunuz ve misafir ağına bağlanan kullanıcıların sizin sisteminize, sunucularınıza erişmesini istemiyorsunuz. Burada misafir ağı için ayrı bir VLAN kullanırsanız sizin networkünüze erişemeyecek, böylece izole etmiş olacaksınız.   * VLAN 2.katmanda çalışır ve switchlerden meydana gelir. Maximum 4096 adet oluşturulabilir. Data VLAN: Standart trafiği taşımak için yapılmıştır. Default VLAN: Varsayılan VLAN yapılandırmasıdır. Genel "VLAN 1" olarak bulunur. Switch’in bütün portları default VLAN olarak gelir. Herhangi bir konfigürasyon yapılmazsa bu switchin portlarına takılan cihazlar birbirleri ile haberleşir. Voice VLAN: Üzerinden ses trafiği geçmesi için kullanılır, ses iletimi için IP telefonlar kullanılır. Native VLAN: Bu VLAN türü herhangi bir VLAN etiketine sahip olmayan çerçevelerin trunk portlardan taşınmasını sağlar. Özel olarak bir VLAN’a atanmamış paketler native VLAN’a yönlendirilir. Native VLAN varsayılan olarak VLAN 1’dir. * Switch üzerinde VLAN’lar oluşturulduktan sonra kullanılacak cihaz için belli portlara atanır, birkaç faklı port türü vardır.   Access Port: Bir Access portu sadece bir VLAN’a atanabilir ve o VLAN ile ilgili bilgileri sadece o porta iletir. Son client’lar VLAN bilgilerini anlamazlar, yani switch’ler access port’a gönderilecek frame’lerden VLAN bilgilerini çıkartır.  Trunk Port: Trunk port ile aynı anda, farklı VLAN’ların tamamını tek bir portun parçası yapabilirsiniz. Bu trunk port, switch-switch veya router-switch arasında kullanılır. Trunk porttan iletilen veri de VLAN bilgisi bulunur. Bu sebeple bir bilgisayar gibi bir cihaz bu VLAN bilgisini anlamayacağı için trunk portlara son cihaz takılmaz. Son cihazlar paketler etiketsiz gönderilir. Switchler arasında olan trunk portlar var olan bütün VLAN bilgilerini aktarır.   * Aşağıdaki şekilde SW1’e bağlı bir client SW3’teki client’a erişmeye çalışırken, öncelikle bir broadcast istek olan ARP isteği gönderecek ve fiziksel adresini öğrenmek isteyecek. Fa0/1’den giren broadcast switche geldiğinde switch tarafından yine aynı VLAN’a atanmış portlara ve trunk portlara gönderilecek. Normalde bütün portları default VLAN olan switch’te bu broadcast bütün portlara gönderilir. Trunk porta gönderilirken bir sonraki switch'in bu gönderilen paketin VLAN 10’dan geldiğini anlaması için gönderen switch tarafından etiketleme işlemi yapılarak gönderilir ve bu paketler **"Tagged Frame"** olarak adlandırılır. Bu tagged frameler sadece router ve switchler tarafından açılıp işlenir. SW2 den geçip trunk porta iletilen paket SW3 e geldiğinde, SW3 frame deki VLAN 10 tag’ini görüp bu paketi açar ve VLAN10 portunun bulunduğu FA0/2 ye gönderir. Gönderilen pakette herhangi bir etiket bulunmaz, bu paketlere de "Untagged Frame" denir.      * Trunk olarak ayarlanan bir port, switch üzerinde bulunan bütün VLAN broadcastlerini geçirecek şekilde davranır. Yukarıdaki topology’i ele alırsak, mesela SW1’deki VLAN20 cihaz bir broadcast yayın yaptığı zaman bu broadcast trunk portlardan SW3’e kadar gidecek. Fakat topology’i gözden geçirirsek SW3’de VLAN20 broadcast’in gelmesinin gerek olmadığını görüyoruz çünkü karşılayan herhangi bir uç cihaz bulunmuyor. Bu yüzden bu SW3 için fazladan bir broadcast trafiği meydana geliyor. Bunu engellemek için biz TRUNK portlara sadece istediğimiz VLAN bilgilerini geçir diyebiliyoruz. SW2’nin Gi0/0 portuna sadece VLAN10 ve VLAN30 bilgilerini geçir dediğimiz anda VLAN20’den başlayan herhangi bir broadcast SW3’e gelmeyecektir.   Untagged frame ile Tagged Frame, switchin anlayacağı VLAN bilgisi TAG olarak frame’e eklenir.     * Voice Port: Bir Access portunun sadece bir VLAN’a atanabileceğini söyledik. Bugünlerde, birçok switch, bir switch porttaki Access portuna, ses trafiğiniz için ikinci bir VLAN daha eklemenize izin vermektedir. O, voice VLAN’ı olarak belirtilmektedir. Bu teknik olarak farklı bir link tipi olarak kabul edilse de hala hem veri hem de ses VLAN’ı için yapılandırılabilen bir Access linkidir. Tek switch portuna hem PC hem de telefon bağlamanıza izin verir. Burada verinin ayrımı Trunk portta gördüğümüz tagged ve untagged olayıdır. Bir switch’in portuna Access VLAN 15 ve Voice VLAN 5 dediğimiz anda switch voice ile ilgili paketleri etiketleyecek DATA ile ilgili paketleri etiketlemeyecektir. IP telefonda sadece etiketli paketleri açacağından diğer etiketsizler bilgisayara gidecektir.      * Collision Domain: VLAN ve haberleşme için broadcast domainden bahsetmiştik, broadcast domainden daha basit bir haberleşme şekli olan L1 katmanındaki (fiziksel katman) haberleşmeden bahsedelim. HUB’ların L1 katmanında çalıştığını TCP/IP katmanlarında belirtmiştik. Peki bu HUB’lar nedir? **HUB** portlarına cihaz bağlı bir haberleştirme cihazıdır. HUB’a bağlı bir cihazdan bir veri gönderilirse bu veri diğer tüm portları gidecektir ve diğer cihazlar da bu veriyi alacaktır. 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. Eğer boşalmadan aynı anda veri yollamaya kalkarlarsa collision dediğimiz, çarpışma yaşanır ve verinin tekrar gönderilmesi gerekmektedir. Bu yüzden her cihaz aynı yolu kullandığından HUB’lar tek bir collision domain oluştururlar.  Switchlerin HUB’lardan en büyük farklarından biri switch’in her bir portu ayrı collision domain oluşturur. Veriler anahtarlanarak gönderilir. İki cihaz haberleşirken başka bir cihaz farklı bir cihazla haberleşebilir ve collision yaşanmaz. Özet olarak söylersek,   HUB – 1 collision domain, 1 broadcast domain SWITCH – (x portuna cihaz bağlıysa) x collision domain, 1 broadcast domain (her portu default VLAN ise) ROUTER – her portu 1 broadcast domain oluşturur. Bir örnekle inceleyelim;     * HUB’in her portuna bağlı cihazlar 1 collision domain oluşturuyor demiştik, switchin her bir portu da 1 coliision domain demiştik yani burada 6 tane collision domain var, router’ın her portu bir broadast domain oluşturacağından burada 5 tane broadcast domain vardır.   Bridge: Bridgeleri, 2 portlu switch olarak düşünebilirsiniz. Bridge tek anahtarlamalı sistem olarak çalışır. Hublardan daha spesifiktir.    Comp1 Comp2 ile haberleşeceği zaman aynı anda Com5 ve Comp6 da aralarında haberleşir. İki ayrı collision domain olur bu sayede. Ne zaman comp1 comp5 ile haberleşme ihtiyacı duyar o zaman anahtarlama sayesinde paket diğer porta gider. Üstüne tanımlanan MAC adresleri sayesinde iki ağı birbirinden ayırarak haberleştirir.  Bir örnekle inceleyelim,    Hublar’ın bütün portları tek collision domainken bridge 2 portu da ayrı collision domain oluşturur demiştik, switchin bütün portları ayrı collision domain bu yüzden 5 tane collision domain mevcuttur.  Yine gördüğümüz bütün network cihazları ile bir örnek inceleyelim,    Gördüğünüz gibi kuralları net belirlediğiniz sürece collision domain sayısını bulmak zor olmuyor. |

|  |
| --- |
| 15. MAC Adresi  * **MAC adresi** 6 oktet, 48 bitten oluşur. Bu 6 oktetin ilk 3 okteti IANA tarafından belirlenir. Bir firma Ethernet Kartı üretmeye karar verirse ilk başvuracağı yer IANA’dır. IANA firmaya o firmanın ID’si gibi düşünülebilecek 3 oktetli bir sayı verir son okteti de firmaya bırakır.   Birbirinin MAC adresini öğrenen iki PC birbiri ile haberleşebilir. Burada şu soru sorulması gerekiyor; peki MAC adreslerini nasıl öğrenecekler? Burada devreye IP adresleri giriyor. IP adresleri logic adreslerdir. 2 PC'nin aynı ağ içinde olmaları için IP adreslerinin aynı network ve subnet içinde bulunmaları gerekmektedir. 16. Address Resolution Protocol (ARP)  * **ARP (Adres Çözümleme Protokolü)**, aynı yerel ağdaki bir aygıtın MAC adresini bulmak için kullanılır. Diğer bir deyişle, client kendi networkünde olan bir başka client ile haberleşeceği zaman ARP sorgusu yapıp MAC adresini öğrendikten sonra haberleşir.   **Bilgisayarlar ARP tablosunu, switch’ler MAC tablosunu tutarlar.**   * Bir paketin bir bilgisayardan çıktığında nereye gideceğini IP numarası değil gideceği bilgisayarın fiziksel adresi (MAC) belirler. Bu adreste paketin gideceği IP numarası kullanılarak elde edilir. Ardından paket yönlendirilir. ARP adres çözümlemek istediği zaman tüm ağa bir ARP istek mesajı gönderir ve bu IP adresini gören ya da bu IP adresine giden yol üzerinde bulunan makine bu isteğe cevap verir ve kendi fiziksel adresini gönderir. ARP isteğinde bulunan makine bu adresi alarak verileri bu makineye gönderir. Tüm ağa gönderilen bu tür paketlere **broadcast** denir.   Diğer bir ifade ile, bir mesajın ağdaki tüm cihazlar tarafından alınmasını sağlayan protokoldür.  Aşağıda üçüncü resimdeki örnekte *"Production, Sales ve Marketing"* olmak üzere 3 farklı broadcast yayını bulunmaktadır.     * Bilgisayar ağları üzerinde çalışan programlar, diğer bilgisayarlarda çalışan programlar ile iletişim kurarken çeşitli yöntemlerle gönderilen mesajın kimlere ulaşacağına karar verirler. Bazı paketler direk olarak bir bilgisayarı ilgilendirirken, bazılarının belirlenmiş bir grup bilgisayara bazılarının ise aynı ağdaki tüm bilgisayarlara ulaşmasını isteyebilirsiniz. Tüm ağa ulaşmasını istediğimiz bu tür paketlere **broadcast** denir. ARP'in bir broadcast protokol olduğundan bahsetmiştik. A makinesinin tüm ağa attığı paket broadcasttir, makine B'nin direk Makine A'ya cevap verdiği pakete ise **unicast** denir.   Broadcast paketler tüm ağa iletileceğinden gereksiz band genişliği harcanmasına sebep olur, bu sebeple **multicastler** (Grup adresleri kullanılarak, birden fazla cihazın tekil bir adresi dinlemesi (buradan veri beklemesi) sağlanmaktadır.) vardır. Belirli bir gruba gönderilirler bu sayede band genişliğinden tasarruf edilmiş olur. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 17. İstemci (Client) ve Sunucu (Server) Nedir ? (Network Edge)  * **Network Edge = end system, host ve client’tan meydana gelir.** * Bilgisayar ağlarında, internet’e bağlı tüm cihazlar (bilgisayarlar, sunucular, mobil cihazlar vb.) **uç sistem (end system)** veya **host** olarak adlandırılır. **Hostlar,** istemci (client) ve sunucu (server) olarak iki gruba ayrılır. * Haberleşecek taraflardan bir tanesi gelecek istekleri dinlemekte diğer taraf ise istekte bulunmaktadır. **Dinleyen taraf sunucu (server), istekte bulunan taraf ise istemci (client)** olarak isimlendirilmektedir.  18. Protokol Nedir ?  * **Protokol**, bilgisayarların birbirleriyle bir ağ üzerinden nasıl iletişim kuracaklarını belirleyen bir dizi kural sistemidir.   **Diğer bir deyişle** **protokol**, **network entitie’ler** (bilginin bir yerden bir yere gidene kadar üzerinden geçtiği her bir elemandır. Router, switch gibi) arasında gelen ve giden mesajların sırasını ve formatını belirleyen bir elemandır.   * Ağlarda, iletişim yapan iki birimin bir işi başarıyla yapabilmesi için **aynı protokolü** kullanması gereklidir.  19. Güvenli Köprü Metni Aktarım Protokolü (HTTPS)  * **HTTPS,** iki bilgisayar arasındaki iletişimin güvenli hale getirilmesi için kullanılır. HTTP, istemci tarayıcısı ile web sunucusu arasında köprü metni biçiminde veri aktarımı için kullanılır; HTTPS protokolünde ise veri aktarımı şifreli bir biçimde yapılır.  20. HTTP (Hypertext Transfer Protocol)  * **HTTP** protokolü; client (istemci) (bilgisayar veya telefon) ile sunucu (server) arasındaki kuralları ve iletişimi sağlayarak ağ üzerinden, web sayfalarını görüntülemenizi sağlayan bir internet protokolüdür. * HTTP **Uygulama (Applicaiton) katmanında çalışır** ve iletişim için **TCP** **veri aktarım protokolünü ve 80 portunu kullanır.**      * **"host name"** dediğimiz alan adımızı, sunucu adımızı belirtmiş oluyor. **"Path name"** ise web sayfası içerisinde var olan nesnenin sunucu içerisindeki yolunu belirtmiş oluyor.  20.1. HTTP neden TCP kullanır? TCP protokolü veri kaybı olmadan veri gönderimi sağlar. Eğer bir paket bozuk veya eksik ise yeniden gönderir.  **HTTP neden UDP kullanmaz?**  UDP güvenli veri transferi yapmaz yani paket eksik gitmesi UDP’nin umrunda olmaz.  Ayrıca 3'lü el sıkışma olayını gerçekleştirmez anında bağlanır.   * **HTTP "****Stateless" bir protokoldür.**   HTTP protokolü, istemci ve sunucu tarafından sonraki istekler ve cevaplarda kullanmak üzere **durum bilgisini (state)** tutmaz. Tutmamasının sebebi sunucu veya istemci tarafından bir hata oluşursa eski durum bilgisi gönderildiğinde yanlış bir bilgi veriyor olacağından unutup tekrar yollaması daha sağlıklı olmaktadır.  **HTTP bağlantısı 2 türlü yapılabilir;** 20.2. NON-Persistent HTTP (Kalıcı Olmayan Bağlantı) (HTTP 1.0) Kalıcı olmayan HTTP üzerinden en fazla 1 adet web nesnesini gönderir ve daha sonra TCP bağlantısıı kapatılır. Her nesne için ayrı ayrı TCP bağlantısı açılır. Yani birden fazla nesne varsa TCP bağlantısı açılır 1 nesne alınır, TCP bağlantısı kapanır ve daha sonra diğer nesneler için aynı işlem tekrar tekrar yapılır.  Kalıcı olmayan HTTP bağlantısı için bir örnek verelim; "<http://www.ceyhan.com/anasayfa>" sayfamızın adresi olsun, içerisinde 1 adet HTML dosyası ve 8 adet PNG dosyası, toplamda 9 adet web nesnesi olduğunu varsayalım. Sırasıyla;   1. İstemci 80 numaralı porttan <http://www.ceyhan.com> adresini barındıran sunucuya bir TCP bağlantısı başlatır. 2. İstemci sunucuya, bize göndermesini beklediğimiz objenin yolunu (/anasayfa) *Request (istek)* gönderir. 3. Sunucu isteği kabul eder, istemcinin istediği objeyi kendi belleğine alır, istemciye yollamak üzere *Response (tepki)* mesajına ekler ve gönderir. 4. Sunucu TCP’ye TCP bağlantısını kapatmasını söyler. **<<** İstemci tepki mesajının tamamını alana kadar kapanmaz. (**Connection-Oriented) >>**   **<<** Paket kaybolur veya bozulursa tekrar gönderilmesi sağlanır. (**Acknowledgement-****Retransmission**) **>>**   1. İstemci response mesajını alır (Bu response mesajının içerisinde HTML dosyamız ve referans edilmiş 8 adet PNG formatı vardır) ve TCP bağlantısını kapatır. 2. İlk yapılan 4 işlem sırasıyla referans edilmiş diğer nesneler için tekrarlanır.   **Özetle** bu örnekte web sayfasının içeriğini görmek için 9 defa TCP bağlantısı kurmak zorunda kalmış olduk. Tabii burada bekleme süresi hayli fazla olacaktır ve her bir obje için 2 kat [RTT](#RTT_roundTripTime) zaman kaybı yaşanır.  **(9 RTT + Transmission Delay (Dosya İletim Gecikmesi))** Bu her bir TCP bağlantısında TCP’nin arabelleği (Buffer) bizim için ayrılmış olur ve TCP değerleri hem sunucuda hemde istemci de tutulmak zorunda kalır. Bu yoğun istek yapılan sunucularda büyük sorunlar yaratabilir.  **Non-Persistent http response time = 2 RTT + file transmission time** 20.3. Persistent HTTP (Kalıcı Olan Bağlantı) (HTTP 1.1) Kalıcı olmayan bağlantının aksine; sunucu her bir response mesajı yolladıktan sonra TCP bağlantısını kapamayıp açık bırakır ve bütün nesneleri tek bir TCP bağlantısı üzerinden yollar, tüm nesneler alınana kadar TCP bağlantısı sonlandırılmaz. Kalıcı TCP bağlantı içerisinde sunucu tarafından istemci tarafına gönderilen nesneler gönderilip gönderilmediği emin olduktan sonra diğer bir veri gönderilir.   * HTTP ‘de 2 adet mesaj formatı vardır.   **Request** istemci tarafından sunucu tarafına tam olarak ne yapacağımızı bildirdiğimiz bir mesajdır.  Sunucu istek karşılığında istemciye bir cevap verir bu tepkiyede **"Response"** denir.   |  |  | | --- | --- | | **HTTP Request Message (1.0 Version)** | **HTTP Request Message (1.1 Version)** | | **GET,** sunucudan veri almak için kullanılır.  Gönderilen veriler (şifre-kullanıcı adı vs.) URL kısmında gözükür. | **HEAD’in,** GET metodundan farkı yanıt gövdesinin (body) bulunmamasıdır. Genelde bir kaynağın var olup olmadığını sorgulamak için kullanılır. | | **POST** ile sunucu üzerine bir veri yazdırılabilir.  Veriler, URL’de görünmez. | **PUT,** belirli bir server’a (sunucuya) nesne yüklemeye yarar. | | **HEAD** | **DELETE,** sunucu üzerinde belirtilmiş kaynağı silmek için kullanılır. | |  | **GET** | |  | **POST** |  |  |  | | --- | --- | | **Response Status Code ( HTTP Yanıt Kodları)** | | | **200 OK** | İstek başarılı alınmış ve cevap başarılı verilmiştir. | | **301 Moved Permanently** | Bir sayfanın kalıcı olarak başka bir yere taşındığını bildirir ve o yere yönlendirme sağlar. | | **404 Bad Request** | İsteğin hatalı olduğu belirtilir. | | **404 Not Found** | İstek yapılan sayfanın bulunamadığını belirtir. | | **505 HTTP Version Not Suppoerted** | HTTP Protokol versiyonu desteklenmiyor. |     En üst kısmın istek kısmı **(request)** olduğunu görüyoruz. Yani request mesajının hangi metot ile gönderileceğini (GET-POST), URL bilgisini ve HTTP versiyon bilgisinin olduğu alanlar mevcut.  Daha sonra başlık bilgisinin yer aldığı bir alan.  Son olarak mesajın bir içeriği varsa, body kısmında gösterilir. |

|  |
| --- |
| 21. Katmanlı Mimari İnternet son derece karmaşık bir sistemdir. Çok sayıda uygulama, protokol, farklı uç sistemler, paket anahtarlar ve iletim ortamlarına sahiptir.  İnternetin bu karmaşıklığı, uçakla yolculuk sırasında yapılan **bilet alımı, bagaj kontrolü, kapıya gidiş, uçağa biniş, uçağın uçması, hedefe varması** gibi bir dizi işe benzetilebilir. Her iş parçasının diğer iş parçasıyla ilişkisi bulunmaktadır. Her katman bir hizmet sağlar. Her katman üst katmandan bir giriş alır ve alt katmana çıkış sağlar.  diyagram içeren bir resim  Açıklama otomatik olarak oluşturuldu  Bilgisayar ağlarında, her katman, kendisinin altındaki katmanın hizmetlerini kullanarak belirli işleri yapar. Farklı katmanlardaki protokoller **protokol yığını (protocol stack)** olarak adlandırılır. İnternet protokol yığını 5 katmandan oluşur. 21.1. Uygulama Katmanı (Application Layer) (TCP kullanır) Uygulama katmanı protokolü çok sayıdaki uç sistemde dağıtık çalışır ve bir uç sistemden diğerine veri aktarır. İnternet uygulama katmanı; **HTTP, IMAP, SMTP, FTP** ve **DNS** gibi protokolleri bulundurur.  Uygulama katmanı veri parçası **mesaj** olarak adlandırılır.  Transport layer ile applicaation layer arasındaki iletişimi sağlamak için **socket** kullanılır. 21.2. Ulaşım – İletim – [Taşıma Katmanı](#transportHizmetleriVeProtokoller)(Transport Layer) (SIP-WOW-FPS 🡪 TCP or UDP kullanır) İnternet ulaşım katmanı, uygulama katmanı mesajlarını uç sistemlerde çalışan uygulamalar arasında aktarır.  Ulaşım katmanı veri parçası **segment** olarak adlandırılır.  İnternet ulaşım katmanında iki protokol vardır. Bunlar [**TCP**](#TCP)ve[**UDP'dir**](#UDP)**.**  **Transport Layer’ın Sunmuş Olduğu Servisler (Hizmetler)**  **Data integrity** (veri kaybının olmaması *reliable data transfer-everflowing*-*loss tolerant apps*), [**throughput**](#throughput), **timing** (karşı tarafa en kısa sürede varması yani delay’in olmaması) ve **security** (verilerin gizli olması, veri bütünlüğünün bozulmaması ve 3. kişinin yani MITM olmaması lazım.) 21.3. Ağ-İnternet-IP Katmanı (Network Layer) İnternet ağ katmanı, kaynak ile hedef host arasında ağ katmanı paketlerinin taşınmasını sağlar.  Ağ katmanı veri parçası **datagram** olarak adlandırılır.  İnternet ağ katmanı, **IP protokolünü** ve **yönlendirme (router) protokolünü** bulundurur. 21.4. Bağlantı Katmanı (Link Layer – Data Link Layer – Network Interface) Bir datagram hedef hosta giderken bir link üzerinde **Ethernet** protokolü ile yani verinin fiziksel aktarımı sağlar.  Link katmanı protokolleri, **Ethernet, WiFi** ve **Point-to-Point Protocol (PPP)**.  Link katmanı veri parçası **frame (çerçeve)** olarak adlandırılır. 21.5. Fiziksel Katman (Physical Layer) Link katmanı, frameleri bir düğümden sonraki düğüme aktarırken, fiziksel katman frame’in içinde yer alan bitleri bir düğümden sonraki düğüme taşır. 22. OSI Modeli (Open Systems Interconnection)  * ISO (International Standard Organization) tarafından bilgisayar ağları 7 katmanlı organize edilmiş ve OSI olarak adlandırılmıştır. * OSI modelinde, **application, presentation (sunum), session (oturum), transport, network, data link** ve **physical katman** bulunmaktadır. Application, transport, network, data link ve physical katmanları ***internet katmanlarıyla (IP) hemen hemen aynı işleri yaparlar.*** * **Sunum (presentation) katmanı**, verinin gösterimi, şifreleme ve sıkıştırma hizmetlerini sağlar. **Oturum (session) katmanı**, veri gönderimi sırasında checkpoint oluşturur ve bir sorun oluşursa recovery işlemlerini yapar.  22.1. Mesaj, Segment, Datagram ve Frame  * Router ve switch protokol yığınındaki tüm protokolleri bulundurmazlar. Alt katmanları bulundururlar. **Link-layer switch 1. ve 2. katmanı, router ise 1., 2. ve 3. katmanı  bulundurur**. * İnternet router’ları IP protokolünü çalıştırır. **Link layer switch’ler** IP protokolünü çalıştırmazlar dolayısıyla **IP adreslerini bilmezler** ve katman 2 adresleriyle işlem yaparlar ancak **router’lar IP adreslerini bilirler**. Hostlar 5 katmanı da çalıştırırlar. * Gönderen host’ta her katman üst katmandan aldığı veriye kendi başlık bilgisini ekler ve alt katmana gönderir.   Uygulama katmanı mesajı (M) ile ulaşım katmanı başlık bilgisi (Ht) birleştirilerek ulaşım katmanı segment’ini oluşturur (encapsulation).  Ağ katmanı, ulaşım katmanından aldığı segment’e başlık bilgileri (Hn) ekleyerek ağ katmanı datagram’ı oluşturur.  Ardından link layer başlığı (Hl) eklenerek frame oluşturulur.  diyagram içeren bir resim  Açıklama otomatik olarak oluşturuldu  Mesaj, Segment, Datagram ve Frame Simülasyonu   * Her veri parçasında iki tür alan vardır: **overhead data (başlık bilgisi)** ve **payload data (üst katman veri parçası)**. Her katmana ait veri parçası alt katmanda birden fazla parçaya bölünebilir.  22.2. OSI Katmanlarının Görevleri Nelerdir?  22.3. [TCP/IP ve OSI Modeli Farkı Nedir?](#tcpIPveOSImodeliFarkı) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 23. Ağ Saldırıları  * **Malware (kötücül yazılım),** kötü amaçlı yazılımların hepsine verilen genel ve kısa addır. * Bilgisayarlar **binlerce cihazla (botnet)** aynı ağda çalışır. Kötücül yazılımlar bir hosta bulaştıktan sonra, bu hosttan internet üzerindeki diğer hostlara da bulaşır **(self-replicating)**. Böylece internet üzerinde çok hızlı yayılırlar.   Kötü amaçlı bot yazılımı, bir uzaktan kumanda işlevini indirip yükleyerek, genellikle bir e-posta veya web sayfası bağlantısı aracılığıyla bir bilgisayara bulaşır.  Enfekte olan **"zombi"** bilgisayarlar, botnet yaratıcısı tarafından yönetilen sunucularla iletişim kurar.  Sunucular, **botnet** adı verilen güvenliği ihlal edilmiş aygıtlardan oluşan bir ağın tamamı için bir komut ve kontrol (C & C) merkezi görevi görür.   * Kötücül yazılımlar, **virus**, **trojan (truva atı)** veya **worm (solucan)** olarak kullanıcı bilgisayarlarına bulaşırlar.  1. Virus   Virusler bilgisayara bulaşmak için kullanıcının etkileşimine ihtiyaç duyarlar (e-posta eki açmak vb.). Aktif hale getirildiğinde, bir virüs kendini çoğaltmaktan ve yayılmaktan başka bir şey yapamaz.  Virüs, diğer programları veya dosyaları değiştirerek yayılan bir programdır.   1. Worm   Worm’lar kullanıcı etkileşimi olmadan bulaşabilirler. Kullanıcı bilmeden korumasız bir uygulamayı çalıştırır, bu uygulama internet’ten bir kötücül yazılımı alır ve çalıştırır. Ardından diğer hostları tarar ve aynı uygulamayı çalıştıran diğer bilgisayarlara bulaşır.   1. Trojan (Truva Atı)   Trojan atları ise, faydalı bir yazılımla gelen kötücül yazılımdır.  Bir Truva atı kendini kopyalayamaz ve kurbanı programı başlatması için kandırmak için meşru görünümüne güvenir.   * **DoS (Denial Of Service- Servis Hizmet Reddi)** saldırısı sunucuda **trafiği arttırarak hizmetin kesintiye uğramasını** sağlar. **DDoS (Distributed Denial of Service- Dağıtılmış Hizmet Reddi)** ise saldırının bir kaynaktan değil de fazla sayıda farklı kaynaktan başlatılmasıyla gerçekleşir. İnternette **DoS saldırıları 3 şekilde** gruplanır:  1. Vulnerability Attack (Ping of Death)   İyi oluşturulmuş mesajlar hedef sunucu uygulamasına gönderilir. Belirli bir sırada gönderilen mesajlar sunucu uygulamasını durdurur, hizmeti yavaşlatır veya bozabilir.   1. Bandwith Flooding   Çok sayıda paket hedef host’a gönderilir. Mesajlar bağlantıda tıkanıklığa neden olur ve normal paketlerin erişimi engellenir.  Bandwidth flooding saldırısında saldırı yapan kişi sunucunun bant genişliği kadar (R bps) trafik oluşturursa hasara neden olur. Bu duruma bir saldırı ile neden olunamaz. Çok sayıda saldırı ile yeterli düzeyde trafik oluşturulması gerekir. Eğer router aynı kaynak host üzerinden gelen trafiği algılar ve engellerse saldırı engellenmiş olur.   1. Connection Flooding (SYN – Synchronous Flooding)   Hedef host’a çok sayıda TCP bağlantısı açılır ve normal bağlantı istekleri kabul edilemez.   * İnternet’te paketlere yanlış adresler kullanarak bulaşma **IP spoofing (aldatma)** olarak adlandırılır. Bunu engellemek için **end-point** **authentication** yapılır. Gelen mesajın doğru yerden gelip gelmediğini belirler. * **Brute force’da** saldırgan bir koda erişmek veya kodu kırmak için art arda çok sayıda olası parola dener. * **Man in the Middle (Ortadaki Adam)** olarak adlandırılan saldırıda **iki uç sistem arasında bir noktada kötü amaçlı kişi iletişime dahil olur.** Sadece paketlerin kopyasını almakla kalmaz, paketlere bulaşabilir, paketleri silebilir veya değiştirebilir. Public Key Infrastructure (PKI) veya karşılıklı authentication ile önlem alınabilir. * **Sniffing (koklama)** bir network ağı üzerinde yer alan **veri akışını dinleyerek veriyi ele geçirmeyi amaçlar**. Veri trafiği içerisinde yer alan paketleri ele geçirebilmek için o veriyi dinler.  1. Pasif Siniffing   HUB cihazı olan network ağlarında kullanılır. Ağ ortamında yer alan veriler birden fazla bilgisayara transfer edilir. Bu aşamada veri paketlerini koklamak daha kolay olur.   1. Aktif Sniffing   Switch cihazlarının kullanıldığı sistemlerde tercih edilir. Sadece belirli bir MAC adresine gönderilen verilerin saldırganlar tarafından MAC adresini istemci gibi gösterip Switch cihazının beynini karıştırır. Bu şekilde Switch cihazının HUB olarak çalışmasını sağlar ve tüm portlardan veri çıkar.   * Veri gönderen bilgisayar için; verinin uygulama katmanından başlayarak en alttaki fiziksel katmana doğru ilerlerken her bir OSI katmanında yeni başlık bilgilerinin eklenip fiziksel katmanda bitlere dönüştüğü sürece**veri kapsülleme (data encapsulation)** denir.   Veri alan bilgisayar için; kablodan elektrik sinyali şeklinde alınan bitlerin fiziksel katmandan uygulama katmanına doğru her bir OSI katmanında başlık bilgilerinin ayrılmasına ve en üst katman olan uygulama katmanında veriye ulaşmaya **veri kapsül açma  (data de-encapsulation)**denir. 23.2. Sosyal Mühendislik Atakları23.2.1. Pretexting Genellikle telefonla saldırgan, kurbanın bilgi paylaşmasını veya bir eylem gerçekleştirmesini sağlamak için genellikle uydurulmuş bir senaryo kullanır. Saldırganın etkili olabilmesi için hedeflenen mağdur ile meşruiyet kurabilmesi gerekir. 23.2.2. Phishing Bu saldırıda, dolandırıcı başka bir kuruluştan meşru bir kişiyi temsil ediyormuş gibi yapar. Kimlik avcısı genellikle kurbanla e-posta veya kısa mesaj yoluyla iletişim kurar. 23.2.3. Vishing IP üzerinden Ses (VoIP) kullanan daha yeni bir sosyal mühendislik biçimi. Şüphelenmeyen kullanıcılara, yasal bir telefon bankacılığı hizmeti gibi görünen bir numarayı aramaları için talimat veren bir sesli posta gönderilir. Çağrı daha sonra bir hırsız tarafından ele geçirilir. Doğrulama için telefonla girilen banka hesap numaraları veya şifreler çalınır.   * **Casus yazılımlar,** izleme çerezleri, reklam yazılımları ve açılır pencereler, reklam, pazarlama ve araştırma amacıyla kullanılabilecek kullanıcılar hakkında bilgi toplamak üzere tasarlanmıştır. Bir bilgisayara zarar vermese de, gizliliği istila ederler ve can sıkıcı olabilirler.  1. Spyware   İzniniz veya bilginiz olmadan bilgisayarınızdan kişisel bilgiler toplayan herhangi bir programdır. Bu bilgiler, reklam verenlere veya İnternet'teki diğer kişilere gönderilir ve şifreler ve hesap numaraları içerebilir-   1. Cookies   Bir çeşit casus yazılım, ancak her zaman kötü değildir. Kullanıcı web sitelerini ziyaret ettiğinde bir internet kullanıcısı hakkındaki bilgileri kaydetmek için kullanılırlar. Çerezler, kişiselleştirmeye ve diğer zaman kazandıran tekniklere izin vererek yararlı veya arzu edilebilir olabilir. Birçok web sitesi, kullanıcının bağlanmasına izin vermek için çerezlerin etkinleştirilmesini gerektirir.   * **Adware,** kullanıcının ziyaret ettiği web sitelerini temel alan bir kullanıcı hakkında bilgi toplamak için kullanılan bir casus yazılım biçimidir. Bu bilgiler daha sonra **hedefli reklamcılık** için kullanılır.   **Popups ve pop-unders,** açılır pencereler, bir web sitesi ziyaret edildiğinde görüntülenen ek reklam pencereleridir. Reklam yazılımlarından farklı olarak, açılır pencereler ve açılır pencereler kullanıcı hakkında bilgi toplamak için tasarlanmamıştır ve genellikle yalnızca ziyaret edilen web sitesiyle ilişkilendirilir.  **Popups,** geçerli tarayıcı penceresinin önünde açılır.  **Pop-unders,** bunlar geçerli tarayıcı penceresinin arkasında açılır. 23.3. Güvenlik Araçları ve Uygulamaları  |  |  | | --- | --- | | **Firewall** | Ağa gelen ve giden trafiği denetleyen bir güvenlik aracı. | | **Patches and Updates – Yamalar ve Güncellemeler** | Bilinen bir güvenlik açığını düzeltmek veya işlevsellik eklemek için bir işletim sistemine veya uygulamaya eklenen yazılım. | | **Virus Protection - Virüs Koruması** | Virüsten koruma yazılımı, virüsleri, solucanları ve truva atlarını dosyalardan ve e-postalardan algılamak ve kaldırmak için son kullanıcı iş istasyonuna veya sunucusuna yüklenir. | | **Spyware Protection - Casus Yazılımlara Karşı Koruma** | Casus yazılımları ve reklam yazılımlarını algılamak ve kaldırmak için son kullanıcı iş istasyonuna casus yazılım önleme yazılımı yüklenir. | | **Spam Blocker - Spam Engelleyici** | İstenmeyen e-postaları tanımlamak ve kaldırmak için yazılım bir son kullanıcı iş istasyonuna veya sunucusuna yüklenin | | **Popup Blocker -Açılır Pencere Engelleyicisi** | Pop-up ve pop-under reklam pencerelerinin görüntülenmesini önlemek için yazılım son kullanıcı iş istasyonuna yüklenir. | |

|  |
| --- |
| 24. Cookies  * Cookie’ler sitelerin sizin hakkınızda pek çok şey öğrenmesine izin verir. Sitelere e-mail ve isim bilgilerinizi verebilir. Arama motorları daha fazlasını öğrenmek için **redirection & cookie’leri** kullanırlar. * **GDPR,** cookie’lerin kendi isteğimiz doğrultusunda kabul edilmesi isteğimiz dışında ise kullanılmamasıdır. * Cookie’lerin sağladıklar: * **Yetkilendirme (authorization)** * **Alışveriş sepeti (shopping carts)** * **Kullanıcı oturum durumu** (user session state) (Web e-mail) * Suzan internet’e her zaman aynı PC’den ulaşmaktadır. Diyelim ki bir e-ticaret sitesini ilk defa ziyaret ediyor.   İstek siteye ilk defa ulaştığında, site benzersiz bir tanımlama numarası oluşturur (8734) ve arka-uç veritabanında sunucu tanımlama numarası (1678) ile indekslenen bir giriş oluşturur.     * **Bir web sitesinde cookie kabul edilince gelecek olan bilgier:**  1. HTTP cevap mesajındaki cookie’de yer alan başlık satırı 2. HTTP istek mesajındaki cookie’de yer alan başlık satırı 3. Kullanıcının bilgisayarında tutulan cookie dosyası. 4. Web sitesinin arka uç veritabanı.  25. Web Tampon Belleği (Web Caches) (Vekil Sunucu – Proxy Server)  * **Web cache**, HTTP isteklerini **kök web sunucusu yerine tutan bir ağ varlığıdır**.   Web cache, alt yapıyı değiştirmeden daha fazla para harcamadan **trafiği azaltır** ve **darboğazı önler**.   * Tarayıcı, kullanıcın tüm HTTP isteklerini önce web tampon belleğine yönlendirecek şekilde yapılandırılır.   **Nesne tampon bellekte** **ise**, tampon bellek nesneyi istemciye döndürür.  **Nesne tampon bellekte değilse**, tampon bellek kök sunucudan nesneyi ister ve sonrasında nesneyi istemciye döndürür.   26. Conditional GET (Koşullu GET)  * Herhangi bir web sitesini talep ettiğiniz zaman Get mesajı kullanılır, buradaki koşullu anlamı eğer cache in içindeki kopya nesne güncel ise asıl sunucuya gitme, güncel değilse asıl sunucuya git güncel olanı al demektir. * Tampon bellek: HTTP isteği içerisinde belleğe alınmış kopyanın tarihini belirler   If-modified-since: <date>   * Sunucu: eğer tampondaki kopya güncelse cevapta nesne bulunmaz:   HTTP/1.0 304 Not Modified   27. HTTP/2  * Çok nesneli HTTP isteklerindeki gecikmeyi azaltmayı hedefler. * HTTP1.1: Tek bir TCP bağlantısı üzerinden birden çok ardışık düzene sahip GET'ler kullanıma sunuldu. * Sunucu **FCFS** (first come first served - ilk gelen alır zamanlama) sırasıyla yanıt verir. * FCFS ile, küçük nesnenin büyük nesne(ler)in arkasında sıra beklemesine **(hat başı (HOL) blokajı)** denir.   Bunu aşmak için objeler framelere bölünebilir.   * Aşağıdaki örnek HTTP 1.1 için yapılmış bir örnektir ve O1 en büyük dosyadır ve O2, O3, O4 bunun arkasında beklemktedir. * Ancak ikinci örnek HTTP 2 için için yapılmış bir örnektir ve O2, O3, O4 hızlı bir şekilde iletilir. (Pipeline teknoloji) |

|  |
| --- |
| 28. E-Mail  * Elektronik postalar 3’e ayrılır: * Kullanıcı temsilcileri (user agents) * Posta sunucuları (mail servers) * SMTP (simple mail transfer protocol - basit posta transfer protokolü)  29. SMTP (Simple Mail Transfer Protocol)  * **SMTP,** internet üzerinden gönderilen e-postaların transferini sağlayan iki uçlu iletim sistemini ifade eder. Bu sistem sayesinde iletilerin gönderildiği bilgisayarda ve iletinin gideceği bilgisayarda depolanmasını sağlayarak çevrim içi ya da çevrim dışı olarak kullanılabilmesini sağlar.   **Port 25** üzerinden **TCP** kullanılarak gönderim sağlanır.   * E-posta’ların gönderilmesinde ve arşivlenmesinde SMTP dışında IMAP ve POP adlı iki ayrı protokol daha mevcuttur.   IMAP (Internet Message Access Protocol)  İnternet üzerinden alınan e-postaların arşivlenmesi, görüntülenmesi ve işlenmesi için kullanılan bir e-posta protokolüdür. SMTP’ye göre daha az gelişmiştir. POP’a göre daha hızlıdır.  Bir bulut sistemi üzerinden e-posta iletilerinin depolanmasını ve ihtiyaç duyulduğu sürece ulaşılabilmesi için dosyalanarak saklanmasını sağlar.  IMAP protokolünün en önemli özelliği, alınan e-postaların bütün elektronik cihazlar üzerinden senkronize olarak indirilmesini ve kaydedilmesini sağlar.  POP (Post Office Protocol)  POP özellikle tek ortamda kullanılan e-posta iletileri için verimlidir. Çoklu ortamlarda POP protokolünün en önemli özelliği ise depolama işleminden sonra çevrim dışı olarak da e-postaların kontrol edilebilmesidir.  SMTP’nin IMAP ve POP ile İlişkisi  **IMAP** **ve** **POP** e-posta protokolleri internet üzerinden gelen iletilerin alınması, depolanması ve arşivlenmesi için kullanılan uygulamalardır. **SMTP** **ise** gönderilen iletilerin kayıt altına alınması ve değerlendirilmesi için kullanılmaktadır.   * SMTP’de veriler direkt olarak transfer edilir arada router gibi üçüncü bir şahıs yoktur yani client’tan server’a iletişim vardır. Transfer 3 aşamadan oluşur. * **El sıkışma (greeting) (handshaking)** * **Mesajın iletimi** * **Kapatma (closure)** * Komut /cevap etkileşimi (command/response interaction): Mesajlar **7 bit’lik ASCII** metinleridir. * SMTP **kalıcı (persistent) bağlantı** kullanır. * SMTP sunucusu, mesajın sonuna gelindiğini belirtmek için **CRLF.CRLF** kullanır. * **SMTP:** E-posta mesajlarının, alıcının sunucusuna, teslimi/depolanmasını da sağlar. * **IMAP,** sunucuda saklanan mesajların alınmasını, silinmesini ve klasörlerini sağlar. * **HTTP:** Gmail, hotmail, Yahoo! vb., e-posta mesajlarını almak için STMP (göndermek için), IMAP (veya POP) üzerinde web tabanlı arayüz sağlar. * Elektronik posta sunucuları  1. **Posta kutusu** kullanıcıya gelen mesajları içerir. 2. **Mesaj kuyruğu** giden (gönderilecek) mesajları içerir. 3. **SMTP protokolü** posta sunucuları arasında posta göndermek için kullanılır.   İstemci (client): gönderen posta sunucusu  Sunucu (server): alıcı posta sunucusu 29.1. HTTP ile SMTP farkı  * SMTP, HTTP'den daha eskidir bu yüzden gönderilen tüm mesajlar **7-bit ASCII** biçiminde olması gerekir (HTTP'de ASCII kullanır ancak 7 bit kısıtlaması yoktur). * HTTP: çekme **(pull)**   SMTP: itme **(push)**   * **HTTP’de,** her nesne kendi cevap mesajı içerisine sarmalanır.   **SMTP’de,** tüm nesneler bir mesaj içerisine yerleştirilir.   * HTTP **bant içi** SMTP **bant dışı** kontrolü kullanır.    29.2. Mail (E-Posta) Mesaj Formatı  * SMTP’de mesajın kimden gittiği **(from)**, kime gittiği **(to)** ve konu başlığı **(subject)** header kısmında yer almaktadır.    30. DNS (Domain Name System)  * İnternetteki her birim kendine ait bir IP adresine sahiptir. Bu **IP adresleri** kullanıcıların kullanımı için www.siteIsmi.com gibi **kolay hatırlanır adreslere karşılık** düşürülür. **DNS sunucuları**, internet adreslerinin IP adresi karşılığını kayıtlı tutmaktadır. * Tüm DNS sunucuları internet ağında uygun bir şekilde **DNS Zone’larına (alanlarına)** konumlandırılarak dağıtılmıştır. Her biri kendi bölgesinde sürekli olarak IP adreslerini sorgulamak ve onlarla ilişkili alan adlarını eşleştirmek ve onları kayıt altında tutmak için çalışırlar. * IP adreslerin (32 bit) - **datagramlarını adreslemek için kullanılırlar**. * Ana sistemlerin dağıtık veritabanı sorgulamasını sağlayan bir **uygulama (appliaction) katmanı** **protokolüdür**.   **UDP protokolünü** kullanır.   * DNS’nin görevleri arasında **IP adreslerini host name’e transfer etmekde** var. * Ana sistem lakapları (aliasing) * Kurallı (canonical) ve lakap (alias) adları * MX record ile hem alias hem canonical yapısı kullanılabilir. * Takma ad olarak da adlandırılabilen **CNAME (canonical name)** kayıt türü, ana bilgisayarınızın (sunucu) bir takma ad ile türünün belirlendiği kayıt türüdür. CNAME kaydı ile sunucunun FTP veya örneğin e-posta sunucusu olduğunu belirtmiş olursunuz.  30.1. DNS Sunucuları DNS sunucuları, kendilerine bağlı olan alan adlarına atanan IP adreslerini kayıt altında tutarak **Root (Kök**) **DNS sunucuları** ile bu bilgiyi senkronize edecek şekilde çalışırlar. Sisteminde alan adı bulunduran ve hosting hizmeti veren tüm firmalar da kendilerine ait bir DNS sunucusuna sahiptir ve kendi sistemlerinde barındırdıkları alan adlarını IP adresleri ile ilişkilendirerek DNS sistemine kaydederler.   1. DNS Çözümleyicisi   **DNS çözümleyici,** tarayıcıya yazdığımız alan adının IP adresinin sorulduğu ilk birimdir. DNS çözümleyicisi farklı bir yapılandırma oluşturmadığınız müddetçe **internet servis sağlayıcınız** olacaktır (Türk Telekom, Superonline, Vodafone vb).   1. Root (Kök) Server Nedir?   Tüm dünyada sadece **13 tane root name server** bulunur ve bunları **ICANN** organize etmektedir. Dünyayı zon'lara ayırır ve bölgeler halinde kontrol ederler. **DNS sunucuları,** bağlı bulundukları Root DNS sunucusuna veri akışı sağlayarak hangi alan adının (Domain) hangi sunucuda ve dolayısıyla hangi IP adresinde bulunduğunu bilgi olarak verirler.  Ayrıca Root DNS sunucularının Mirror (ayna) DNS Sunucuları denilen çok sayıdaki sunucusu da daha yaygın bir şekilde bölgelere dağıtık vaziyette konumlanmışlardır ve bu sisteme bağlı olarak çalışırlar. Sistemin yedekli olmasını, daha yaygın ve hızlı çalışmasını sağlamak üzere dünya genelinde dağılmışlardır.   1. TLD (Top Level Domain) Name Server   **TLD,** alan adlarının sonundaki uzantılara verilen isimdir. Bu uzantının çeşidine göre bütün alan adları gruplar halinde kayıt altına alınırlar. **".COM", ".NET", ".ORG", ".TR", ".UK"** gibi uzantılar bir alan adının nereye bağlı olduğunun sorgulandığı ilk adımdır. Bu uzantıların kayıtlarının tutulduğu sunuculara **TLD Name Server** denir.   1. Name Server (İsim Sunucusu)   **Name server’lar**, alan adlarının bir bütün olarak hangi IP adresinde saklandığının kaydının tutulduğu asıl sunuculardır. Alan adı barındıran her firma bir Name Server oluşturur ve o alan adına atanan IP adreslerinin kaydını tutarak bir sorgu yapıldığında tüm DNS sistemine IP adresi bilgisini verirler.  Her alan adına biri yedek olmak üzere **iki adet** Name Server adresi tanımlanır ve bu bilgi tüm DNS sistemine kaydedilir.     * **www.amazon.com’a** bağlanmak için öncelikle kök dizin sorgusu yani **.com** sorgusu yapılır.   Ardından **amazon.com** sorgusu yapılır. Son olarak **www.amazon.com** adresinin IP adresinin sorgulaması-eşleşmesi **(****authoritative (***Bu DNS istenilen sorguyu bulup başarılı bir şekilde sonuç döndürür. İstenilen adresin bulunamaması durumunda gerekli bilgilendirmeyi sağlar.***))** yapılır. 30.2. Local (Yerel) DNS Sunucuları DNS sorguları iki çeşittir:   1. Iterated Query (Yinelenen Sorgu)   Bütün işleri local server yapar.  Bu DNS Server'a gönderdiğiniz sorgular kendi içerisindeki kayıtlar içerisinden çözülebiliyorsa, bu tür sorgulara **iterative query** denilmektedir. Iterative sorguların çözümlenme işlemi kendi DNS sunucunuz üzerinde gerçekleşmektedir. Bunu bir örnekle açıklayacak olursak, benim ortamımdaki vatanci.local domaini için gerçekleştirilecek çözümleme işlemi Iterative'dir. Çünkü bu domainin Zone kaydı benim DNS sunucum üzerinde tutulmaktadır.     1. Recursive Query (Özyinelemeli Sorgu)   Yaptığımız sorguyu karşı taraftan çekmek istediğimizde **recursive query** olmuş oluyor.  Bir DNS Server'in başka bir DNS Server'a DNS sorgusu göndermesine **recursive query** denilmektedir. DNS'miz tarafından çözülemeyen domainleri web üzerindeki DNS'ler üzerinde çözümlememiz gerekebilir.   30.3. DNS Kayıt Türleri  1. CNAME (Canonical Name)   Bu tip DNS kayıtları, bir alan adını diğerine yönlendirmenizi, bir alt alanı birincil veya standart alanla ilişkilendirmenizi sağlar.   1. A   Alan adınız belirli bir IP adresine yönlendirildiği için sunucu değişikliğinde güncellenmesi en önemli olan DNS kaydı türüdür. Bir alan adını veya bir alt alan adını bir IPv4 adresine yönlendirmek için kullanılır.   1. MX (Mail Exchange)   Bir alan adını bir e-posta hizmetiyle ilişkilendirmek için kullanılır. E-postaların alan adı üzerinden nasıl teslim edileceğini belirtir.   1. NS Kayıtları   NS kayıtları, kullanıldıkları bölge için yetkili olan alan adı sunucularını tanımlamak için kullanılır.   30.4. DNS Protokol (Protocol) Mesaj |

|  |
| --- |
| 31. Peert to Peer [(P2P)](#peerToPeer) Mimarisi (256 kb)  * **Peer to peer (P2P),** üçüncü taraf olmadan kullanıcıların doğrudan birbirleri arasında gerçekleştirdiği veri paylaşımını ifade eder. Peer to peer, hem [client process’dir](#clientProcess) hem [server process’dir](#serverProcess).   **Peer to Peer Nasıl Çalışır?**  Peer to Peer ağlarının kullanıldığı programlarda insanlar kendi paylaşmak istedikleri dosyaları sunucuya kaydederler. Bu dosyaya erişmek isteyen diğer kullanıcılar dosyayı sunucu üzerinde arar fakat dosyayı kendi bilgisayarına transfer etmek istediğinde sunucu üzerinden değil, dosyanın sahibinin bilgisayarı üzerinden transfer gerçekleştirir. Sonuç olarak **sunucu** sadece dosyanın eklenmesi ve bulunması aşamasında görev alır.   * Peer’lar birbirlerine bağlanabilirler. Birbirlerine bağlanan peer to peer’ların IP adresleri birbirinden farklı olabilir. * P2P’ın en önemli özelliklerinden biriside **BitTorrent** olmasıdır. **BitTorrent,** eşten eşe dosya paylaşım protokolüdür ve merkezi olmayan bir sistemle internet üzerinden dosya paylaşımına izin verir.   Örneğin, büyük bir dosyayı bir sunucudan çok sayıda ana bilgisayara dağıtmayı hedefleyelim.  **Client-server yapısında** bir dosya paylaşılacağı vakit sunucu dosyanın bir kopyasını eşlerin her birisine yani client’lara göndermek zorundadır çünkü client’lar birbirleriyle iletişime geçemiyor sunucuyla iletişime geçebiliyor dolayısıyla bu da sunucuya büyük bir yük bindirir ve bant genişliği gereksinimini ortaya çıkarır.  **P2P’da ise** her bir eş aldığı dosyanın bir bölümünü diğer eşlere dağıtabilir böylece dağıtım sürecinde sunucuya yardımcı olur. Buna **torrent** denir.   * **Torrent** aslında bir dosya türüdür. Hatta uzantısı da **.torrent** olarak geçer. Fakat bu dosya türü, bizim asıl indirmek istediğimiz büyük boyutlu dosya değil o dosyaya ulaşabilmek için oluşturulmuş bir kaynaktır.   Boyutları büyük olan dosyaları torrent olarak indirmek için bu dosyalar, Torrent sisteminde birçok bite ayrılıyor. Daha sonra bu bitler de Torrent yazılımları (istemcileri) üzerinde birleştiriliyor.  Burada çok önemli bir nokta var. Bu dosya paylaşım türünde sabit bir sunucuya ihtiyacınız kalmıyor. Çünkü o anda var olan tüm kullanıcılardan bu bitler çekilerek birleştiriliyor. Bu sayede daha hızlı bir aktarım söz konusu oluyor.  BitTorrent alıcıların indirdiği dosya parçalarını birbirleri ile paylaşmasından dolayı ağ yükünü ciddi anlamda hafifletiyor. Tabii bu da kullanıcılara daha hızlı dosya indirme imkanı sağlıyor.  Örneğin 20 GB’lık bir veri dosyasına erişim sağlamak istiyorsunuz. Bunun için önce indirmek istemiş olduğunuz dosyanın **".torrent"** uzantılı dosyasını indirmeniz gerekiyor. Daha sonra torrent indirmek için oluşturulan bir yazılım aracılığı ile dosya aktarımına başlıyorsunuz. Bu esnada aynı dosyaya sizin gibi erişmek isteyen birçok kullanıcı mevcut. Yazılım hangi kullanıcının ne kadarlık dosya indirdiğine bakmadan herkesten dosyaları bit olarak çekiyor ve karşı tarafa aktarıyor. Yani siz 20 GB’lık veri dosyasını indirirken tek bir sunucudan dosyayı çekmiyorsunuz. O anda aktif olan kullanıcılardan parça parça dosyaları otomatik olarak çekiyorsunuz. Torrent yazılımınız da bu dosyaları birleştiriyor. Tabii bu esnada dosyanın tamamını indirmemiş olsanız bile sizin de indirdiğiniz kadarını yazılım, karşı tarafa bit bit gönderiyor.   * **Tracker,** indirme yapmak istediğiniz servera sizleri bağlayan bir Bittorent protokolü olarak karşınıza çıkmaktadır. İndirme yapmak istediğiniz dosyayı farklı kaynaklar üzerinden sizlere sunan tracker listeleri, daha hızlı indirme yapmanıza olanak sağlamaktadır.   Trackerların asıl görevi eşleri (peers) birleştirmek, birbirlerini bulmasını sağlamak.   * **Churn,** peer’ların herhangi bir zamanda gelip gitmesi demektir. * **DHT,** dosya paylaşımı yapan kullanıcıların IP adreslerini bularak tracker'a yardımcı oluyor. Bu şekilde daha çok kaynak bulunarak, dosyalar daha hızlı indirilebilir.   **Özetle DHT**, merkezi bir sunucuyla bağlantıya geçmeden peer (dosya paylaşan kullanıcı) bulabilmektedir. 32. Video Streaming and CDN  * **Video streaming,** önceden kaydedilmiş videoların (Netflix, YouTube …) yüklenmesi veyahut indirilmesidir. * Sıkıştırma algoritmaları bir videoyu herhangi bir bit hızında sıkıştırabilir. **Bit hızı ne kadar yüksek olursa** görüntü kalitesi o kadar iyi olacaktır. * **DASH (Dynamic Adaptive Streaming over HTTP),** medya içeriğini (video, ses) birçok küçük parçaya bölerek her parçayı farklı bir çözünürlükte ve kalitede aktarmayı sunar. Bu şekilde, kullanıcının internet bağlantısı hızına ve cihazının özelliklerine göre en uygun kalitedeki medya içeriği gösterilir. * **CDN (Content Distribution Networks),** dünya genelinde birçok sunucuya sahip bir ağdır. Web sitelerindeki statik içerikleri (örneğin HTML, CSS, görüntüler, videolar vb.) bu sunuculara yüklerler ve bu sunucular da kullanıcıların bu içeriklere hızlı ve güvenilir bir şekilde erişmelerini sağlarlar. Kullanıcılar, isteklerini en yakın sunucuya göndererek, içeriğin daha hızlı ve daha az gecikme ile yüklenmesini sağlarlar.   CDN, **internet trafiğini dağıtarak ağ trafiğini azaltır** ve sunucuların yüksek trafiklere dayanabilmesini sağlar.   * **Bring home,** büyük sınıfları IXP noktalarında bir araya getirmeye çalışırlar.  33. Soket Programlama  * **Soket programlama,** TCP/IP'de, veri iletişimi için gereken iki bilgi olan **IP adresi ve port numarasının yan yana yazılmasıyla oluşan iletişim kanalıdır.** * İki farklı bilgisayarda çalışan işlemlerin (process) birbirlerine mesaj göndermek istediklerini düşünelim, bunun için bazı adreslere ihtiyacımız var.   Bunlardan ilki iletişim kuracağımız bilgisayarın **IP adresi,** IP adresini biliyoruz ve karşımızdaki bilgisayarı bulduk ama bizim amacımız o bilgisayardaki belirli bi programla iletişime geçmek işte bunun için de o programın **port numarasını** bilmemiz gerekiyor.   * İletişim protokolü olarak **TCP’de,** sunucu belirli **bir portu** dinleyerek gelen istekleri karşılar, **UDP** **protokolünde** ise, tek bir soket üzerinden **birden çok istemciye** veriyi iletebilir. * **Client,** socket programlarda hizmet isteyen bir işlemcidir.   **Server ise,** socket programlarda hizmet veren sunucudur. Client server’a bağlanmak istediğinde, bağlanabilmesi için server’ın sürekli açık olup, client’ın bağlanacağı portu izlemesi gerekmektedir. 34. TCP / IP (Transmission Control Protocol) (Internet Protocol)  * **TCP/IP**, verilerin aktarılmadan önce nasıl paketleneceğini, bu paketlerin nasıl taşınacağını, nasıl adresleneceği ve hedef tarafından nasıl çözümlenerek alınacağını kontrol eden veri iletim protokolüdür. * ***IP kısmı verinin gönderileceği adresin elde edilmesinde, TCP kısmı ise IP adresi bulunduktan sonra verinin iletimi esnasında güvenilir bağlantılar kurmasından sorumludur*** ayrıca TCP, **taşıma (aktarım) katmanında** uygulanır. * TCP/IP protokol ailesinde; **application (uygulama)**, **transport (taşıma),** **ağ (network)** ve **fiziksel** **katman** **(physical katman)** olmak üzere 4 katman bulunur.  35. TCP/IP ve [OSI Modeli](#OSImodeli) Arasındaki Farklılık Nedir?  * **OSI** 7 katmandır, **TCP/IP** 4 katmandır, bu yüzden TCP/IP daha anlaşılırdır. * **TCP/IP** haberleşmeyi daha basit hale indirger.   Örneğin OSI’deki *oturum***,** *sunum* ve *uygulama katmanına* karşılık TCP/IP sadece *uygulama katmanını* kullanır.   * TCP/IP modeli UDP kullandığı zaman, iletim katmanında güvenlik kontrolü yapmaz.   OSI güvenlik kontrolünü daima yapar. |

|  |
| --- |
| 36. Transport Hizmetleri ve Protokoller([Taşıma – İletim – Ulaşım Katmanı](#transportLayerNedir))  * Transport protokolleri uç sistemlerde çalışır.   Gönderici Taraf (Client): Mesajları önce **segmentlere** (paketin parçalara bölünmesi) böler, sonra ağ katmanına aktarır.  Alıcı Taraf (Server): Segmentleri birleştirerek mesajı tekrar oluşturur ve uygulama katmanına aktarır.   * **Ağ Katmanı:** Hostlar arasında mantıksal bağlantıyı sağlar.   **Ulaşım Katmanı:** Prosesler arasında mantıksal bağlantı sağlar ve ağ katmanı hizmetlerini kullanır.  **İletim katmanı** uç bilgisayarlar arasında gönderilen taşıma katmanı segmentlerı için gecikme ve bant genişliği garantilerini sağlamayabilir.   * Hem TCP’de hem de UDP’de gecikme garantisi ve bant genişliği garantisi yoktur.  37. Multiplexing / Demultiplexing  * Mesajlara header bilgisinin eklenip birden fazla bağlantının eş zamanlı olarak kurulmasına **multiplexing** denir.   Taşıma katmanındaki verilerin doğru sokete iletilmesıne **demultiplexing** denir.   * Multiplexing ve demultpilexing bütün katmanlarda gerçekleşir. * **Demultiplexing nasıl çalışır?**   Host IP datagramlarını alır her bir datagram kaynak IP adresine ve hedef IP adresine gönderilir.  Her bir datagram 1 taşıma (transport) katmanı segmenti taşır.  Her bir segmentin kaynak ve hedef port numarası vardır.  Host, IP adreslerini ve port numaralarını kullanarak segmenti uygun sokete yönlendirir   * Appliaction message’a header bilgisi koyulduğu zaman **segment’e** dönüşüyor. * Gönderici tarafında multiplex, alıcı tarafında demultiplex işlemi gerçekleştirilir. * 0-1023 arasındaki port numaraları **well known** yani **iyi bilinen** port numaları olarak geçer.      * Mesela bilgisayarınızdan browser’a bir internet sitesi çağırdığınızda önce 53 portundan o sitenin IP’sini buluyor, daha sonrada HTTP veya HTTPS üzerinden siteye ulaşılıyor. Mailleri çalıştırırken 25 portunu kullanıyoruz.  37.1. Bağlantısız (Connectionless) Demultiplexing  * Port numaralarıyla soketler oluşturulur. * UDP soketi iki unsurla tanımlanır: (hedef IP adresi, hedef port numarası) * host UDP segment aldığında:   Segmentteki hedef port numarasını kontrol eder. UDP segment port numarasıyla birlikte ait olduğu sokete yönlendirir. IP datagramlar aynı sokete kaynak ve hedef IP ile port numaralarıyla yönlendirilir. Source port dönüş adresini (return address) sağlar.   37.2. Bağlantılı (Connection-Oriented) Demultiplexing  * TCP soketi 4 unsurla tanımlanır: * Kaynak IP adresi * Kaynak port numarası * Hedef IP adresi * Hedef port numarası: Client, hedef sunucuya hangi hizmetin istendiğini bildirmek için segmente bir hedef bağlantı noktası numarası yerleştirir. * Alıcı host bu 4 değer ile segmenti uygun sokete yönlendirir. Server host eşzamanlı çok sayıda TCP soket destekler:  1. Her soket kendi 4 değeriyle tanımlanır. 2. Web sunucular her bağlanan istemci için farklı bir sokete sahiptir. 3. Non-persistent (kalıcı olmayan) HTTP her istek için farklı bir sokete sahiptir.      * Aynı source olmasına rağmen farklı processlere gidebilir, P1, P2, P3 gibi …  38. URL-URN-URI  * **Uniform Resource Name (URN)** - protokole başvurmadan yalnızca kaynağın ad alanını tanımlan * **Uniform Resource Locator (URL)** - ağdaki belirli bir kaynağın ağ konumunu tanımlan * Aşağıdaki URL'de yer alan bileşenler:   **Protocol/scheme** - HTTPS veya FTP, SFTP, mailto ve NNTP gibi diğer protokoller  **Hostname** - www.example.com  **Path and file name** - lauthor/book.html  **Fragment -** #page155 |

|  |
| --- |
| 39. UDP (User Data Protocol)  * **Taşıma-ulaşım-aktarım katmanında (transport layer)** uygulanmaktadır. * **UDP ve TCP arasındaki temel fark,** UDP’nin **güvenilmez** ve **hızlı veri aktarımı** sağlamasıdır. Bağlantıların diğer ucunda **verilerin alındığını doğrulamaz**. Her UDP segmenti diğerlerinden bağımsızdır. * Uzak ana bilgisayarla bağlantı kurmaz **(el sıkışma olmaz)**, **tıkanıklık kontrolü yoktur** yani UDP istendiği kadar hızlı patlayabilir. Ayrıca **sıralı olmayan iletim** gerçekleştirilir.  39.1. Connectionless  * UDP, genellikle multimedya uygulamalarında, streaming (akış) için kullanılır. Her UDP segment diğerlerinden ayrı değerlendirilir. * Diğer **UDP kullanımları**: DNS, SNMP, HTTP/3, NFS, Routing Prokoller (RRT) .   Diğer **TCP kullananlar**: SMTP, TELNET, HTTP, FTP.   * Güvenilir olmayan bir internet transport protokolüdür.  1. Kayıp toleransı vardır. 2. Veri gönderim oranı hassastır. 3. Uygulamaya sırasız halde gidebilir. 4. UDP best effort servisini sunar.  39.2. UDP Checksum  * İletilen segmentte hata algılama için (örneğin ters çevirlmiş bitler) kullanılan bir mekanizmadır.   **Gönderici:**   1. Segment içeriklerine 16-bit integer dizisi olarak bakılır. 2. Checksum segment içeriğinin 1'e tümleyeninin toplamı alınır. 3. Gönderici Checksum değerini yani 1’e tümleyeninin toplamını UDP checksum alanına yerleştirir.   **Alıcı:**   1. Alınan segmentte checksum hesaplanır. 2. Hesaplanan checksum değerinin gelen checksum değeriyle aynı olup olmadığına bakılır. Eğer:    * HAYIR ise = hata var demektir.    * EVET ise = hata yok demektir.  * Sayıları toplarken **en soldaki bitlerdeki taşma**, sonuca eklenir.   Aşağıdaki egzersizde iki 16-bit integer değerden checksum değeri hesaplanıyor.     * **SOCK\_DGRAM** **=** **UDP Socket** * UDP’de client, her pakete IP adresi ve hedef port numarasını ekler. * UDP’de client port açmaz, işletim sistemi uygun portu belirler o port üzerinden iletişim gerçekleşir. * UDP connection’dayken, server’da port açarız yani port numarasını belirleriz.   UDP’nin segment formuna ulaşmak için [tıklayınız](#tcpUDPsegmentForm). |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 40. TCP (Transmission Control Protocol)  * TCP protokolü **kimlik doğrulaması yaparak** veriyi karşı tarafa gönderirken veya alırken **verinin bütünlüğünü sağlar**. Ayrıca **akış kontrolü**, **tıkanıklık kontrolü** sağlar ve **sıralı iletim** gerçekleştirir. * **TCP’nin Sunmuş Olduğu Servisler (Hizmetler)** * TCP, **reliable transport** (gönderilen paketlerin sırasının bozulmadan iletilmesi), * **flow control** (akış kontrolü), * **congestion control** (fazla trafik olduğunda gönderilen paketlerin azaltılmasını sağlar) * **connection-oriented** (karşı tarafla bağlantı kurulduktan sonra paketlerin gönderilmesi yani üçlü el sıkışması gibi bir şey), * **does not provide** (timing’i minimum. Throughput’u ve security’yi garanti etmez.) * TCP protokolü, **üç yönlü (üçlü) el sıkışma** **(Three Way Handshake)** denilen bir sistemi kullanarak veri aktarımı öncesinde bir bağlantı oluşturur.   **Birinci** aşamada hedefe bir bağlantı isteği gönderilir.  **İkinci** aşamada bağlantının gerçekleştiği onaylanır ve veri transferi başlar.  **Üçüncü** aşamada ise veri transferinin tamamlandığı taraflara iletilerek bağlantı sonlandırılır.  Bu üç aşamanın gerçekleşmesi [**state**](#stateStatelessNedir) **işlemi** olarak tanımlanır. Veriler gönderildikten sonra, bağlantı **dört yönlü benzer bir el sıkışma** ile sökülür.  TCP'de bu üç ana aşamanın (üçlü el sıkışması) gerçekleşmesi için bazı ara durumlar gerçekleşmektedir. Bunlar:   |  |  | | --- | --- | | **LISTEN** | Sunucu tarafından bir TCP bağlantı isteğinin beklenildiği durumdur. Dinleme modu olarak adlandırılır. | | **SYN-SENT** | Karşı tarafa TCP bağlantı isteği gönderildikten sonra karşı taraftan bağlantı isteğine cevap verilmesi beklenilen durum olarak adlandırılır. | | **SYN-RECEIVED** | SYN bayrağı ile yapılan bağlantı isteğine sunucunun SYN-ACK bayrağı ile cevap vermesi aşamasından sonraki bekleme durumu olarak adlandırılır. | | **ESTABLISHED** | Bağlantı kurulduktan sonra gelen veri transferinin yapıldığı durumdur. | | **FIN-WAIT-1** | Sunucu ve istemci tarafındaki bekleme durumudur. | | **FIN-WAIT-2** | Karşı taraftan TCP bağlantısının bitirilme isteğinin beklendiği durumdur. | | **CLOSE-WAIT** | Sunucu ve istemci tarafından bağlantı kapatma talebinin beklendiği durumudur. | | **CLOSING** | Karşı tarafa bağlantının bitirilmesine dair bir ACK bayrağı gönderildikten sonra bağlantının bitmesini bekleme durumu olarak adlandırılır. | | **LAST-ACK** | Sunucu ve istemci tarafında ACK beklendiği durumdur. | | **TIME-WAIT** | Bekleme durumudur. | | **CLOSED** | TCP bağlantısının tamamen bittiği durum olarak adlandırılır. |  * TCP bağlantısı kurulurken; yani iki bilgisayar birbiriyle TCP protokolü üzerinden bağlantı kurmak istediğinde aşağıdaki sıralama ile işlemler gerçekleşmeye başlar:  1. X bilgisayarı Y bilgisayarına bir **SYN** (synchronzation-senkronizasyon-bağlantı isteği) mesajı yollar. Bu sayıya **sequence number** denir. 2. Y bilgisayarı X bilgisayarının isteğini aldığına dair bir **SYN+ACK** (isteğin kabul edildiği) mesajı yollar. Ayrıca sequence number 1 artar. 3. X bilgisayarı Y bilgisayarına **ACK** (acknowledgement-bilgilendirme-bağlantı isteğinin kabul edildiğini onaylama) mesajı yollar.   Eğer bağlantı aşamasında, client karşı taraftaki sunucunun dinlenmeyen (açık olmayan) bir portuna bağlantı isteği iletirse -ki bu işlem sırasında **LISTEN** modda olması gerekir.  Sunucudan ***SYN-ACK*** paketi cevabı yerine ***RST-ACK*** cevap paketi gönderilir. Bu noktada sunucu bağlantıyı reddetmiş olur ve bağlantı kurulmaz. ***RST paketi* "reset"** olarak tanımlanır ve açık bir bağlantıyı kesmek için de kullanılır.  metin içeren bir resim  Açıklama otomatik olarak oluşturuldu    **Server tarafında gerçekleşen** olaylar şöyledir:   1. ***socket*** ile haberleşme alt yapısı oluşturulur. Bu durumu telefon cihazının kendisine sahip olmaya benzetebiliriz. 2. ***bind*** ile soket bir IP adresi ve port ile ilişkilendirilir. Bu durumu, telefon numaramıza benzetebiliriz. 3. ***listen*** ile soket üzerinde dinleme işlemi yapılacağı belirtilir. Bu durumu telefonun şebekeye bağlanmasına ve başında cevap vermek üzere beklemeye benzetebiliriz. 4. ***accept*** ile gelen bağlantı isteği kabul edilir. Bu durumu telefon çaldığında cevap verilmesine benzetebiliriz. 5. ***read*** ve ***write*** fonksiyonlarıyla çift yönlü okuma ve yazma işlemleri yapılır. Bu durum karşılıklı konuşmaya denk gelmektedir. 6. ***close*** ile bağlantı sonlandırılır. Bu durum telefonun kapatılmasına karşılık gelmektedir.   **Client tarafında gerçekleşen** olaylar şöyledir:   1. Sunucu tarafında olduğu gibi ilk olarak ***socket*** ile haberleşme alt yapısı oluşturulur. 2. ***connect*** ile karşı tarafa bağlanma isteği gönderilir. Bu durum birini telefon numarasını çevirerek aramaya denk gelmektedir. **SOCK\_DGRAM** türünde kullanılması zorunlu değildir. 3. ***read*** ve ***write*** fonksiyonlarıyla çift yönlü okuma ve yazma işlemleri yapılır. Bu durum karşılıklı konuşmaya denk gelmektedir. 4. ***close*** ile bağlantı sonlandırılır. Bu durum telefonun kapatılmasına karşılık gelmektedir.   Sunucunun yaşam döngüsü boyunca birden çok istemciye yanıt verebilmek için bir döngü içerisinde yeniden **accept** fonksiyonunu çağırdığını görmekteyiz. Genel işleyişi gösterdiğimiz bu örnekte, sunucu aynı anda bir tek istemciye hizmet vermektedir.  IP numarasını bir şirketin telefon numarasına benzetirsek, port numarası konuşmak istediğimiz kişinin dahili numarasına denk gelmektedir. Port numaraları sistemdeki servisleri birbirinden ayırmak için kullanılmaktadır.  İstemci tarafında herhangi bir port numarası belirtmememiz dikkatinizi çekmiş olabilir, işletim sistemi bizim için geçici bir port ayarlayacaktır. Bu tür portlar **ephemeral port** olarak isimlendirilmektedir. 40.1. TCP Yaşam Döngüsü    * Bağlantı sonlandırmayı TCP istemci başlatır.   **Adım 1:** İstemci bağlantı sonlandırma isteğini içeren segmenti sunucuya iletir.  TCP istemci özel bir segmenti sunucuya iletir.  Bu özel segmentin **FIN biti 1** yapılır.  **Adım 2:** TCP sunucu FIN biti 1 olan segmenti alır ve ACK gönderir.  Sunucu bağlantıyı kapatır ve FIN biti 1 olan özel bir segmenti istemciye gönderir.  **Adım 3:** TCP istemci FIN biti 1 olan segmenti alır ve ACK gönderir.  İstemci bir süre bekler ve bağlantıyı sonlandırır.  **Adım 4:** Sunucu ACK segmentini alır ve bağlantıyı sonlandırır.   * TCP, iletim esnasında veriye, içinde bazı kontrol bilgilerinin (datagram, packet vb.) bulunduğu bir başlık (header) ekler. Gönderilen veriler için özel bir **TCP kabul paketi (TCP ACK)** gönderilir ve gelmiş olan paketlerin doğruluğu kontrol edilir.   Başlık bilgisi ve veri parçasının kendisi ile birlikte oluşan TPDU’ya **TCP segmenti** adı verilir. Her segmente bir sıra numarası verilir. Bu segmentler belli sayılarda gönderilir. Alıcıya segmentler ulaştıkça bunlar alıcının tampon belleğine yerleştirilir. İki ardışık segment tampon belleğe yerleşince alıcı, gönderilen en son segment için bir onay mesajını göndericiye yollar.   * TCP’nin en önemli özelliği veri akışını **sürekli** ve **iki yönde (full-duplex)** sağlamasıdır.   Gönderilen veriler 8 bitlik (oktet) gruplar halinde gönderilir. Mesaj parçaları **(segmentler)** çeşitli uzunluklarda  olabilir. Örneğin bir TCP uygulaması 1024 oktetlik veri yollaması gerekiyorsa bu bilgiyi 1024 adet 1 oktetlik veya 256 adet 4 oktetlik parçalar halinde gönderebilir.  Bu parçalar gönderilecek en büyük parça değerini **(maksimum segment size - MSS)** aşmayacak uzunlukta olmalıdır. **MSS,** link layer'daki maksimum transmission unit'e (MTU) göre belirlenir. 40.2. UDP ve TCP Segment Formatı  |  |  |  |  | | --- | --- | --- | --- | | **Source Port**  Gönderilen işlemin portunu gösterir. Eğer gönderen host bir kaynak numarasına sahip değilse bu alan "0" ile doldurulur. | | **Destination Port**  Alıcı tarafın port numaralarını belirtir. 16 bitlik bir alana sahiptir. | | | **Sequence Number**  TCP oturumu işlenirken ne kadar veri gönderildiğini gösteren alandır. Alıcı tarafından bu sıra numarasını kullanarak alındı mesajı verecektir. **Özetle,**  gönderilen veri segmentinin ilk byte’ının numarasıdır. | | | | | **Acknowlendgment** [**Number**](#Acknowledgements)**(Onay Numarası)**  Alıcı tarafından bir sonraki TCP segmentini istemek için kullanılır. **Özetle,** sequence number’dan sonraki byte’ının numarasıdır. | | | | | **Header Length**  UDP veri ve UDP başlığının byte cinsinden toplam uzunluğudur. | **Unused**  Ayrılmış alanları belirtir ve kullanılmazlar. | **Flags (Bayraklar) (FIN, SYN vb.)**  Bağlantılar kurmak, sonlandırmak verileri göndermek için kullanılır. | **Receive Window**  Alıcını kaç bayt veri alacağını gösterir. | | **Checksum**  Hata kontrol mekanizması sağlar. Eğer hata kontrolü yapılmayacaksa bu alan "0" ile doldurulur. | | **Urgent Pointer (Acil İşaretçi)**  Acil verilerin nerede bittiğini gösteren yapıdır. | | | **Options (Seçenekler)**  İsteğe bağlı ayarlamaların yapıldığı alandır. | | | |       ÖRNEK: **Gönderilen ilk veri (ilk yeşil çizgi – ilk byte’ın sıra numarası) yani sequence number 42 ise acknowlendgment number 43 (ilk sarı çizgi) olacaktır**.    **SampleRTT,** bir segmentin gönderilip cevabının (ACK) gelmesi için geçen süredir.  SampleRTT değerleri ile ortalama **EstimatedRTT** hesaplanır.    **tahminEdilenRTT = ((1-α) \* birOncekiTahminEdilenRTTdegeri) + (α \* enSonAlinanDeger)**  **α = 0.125**    **DevRTT = Güven Aralığı**   40.3. TCP Fast Retransmit  * Gönderici aynı segment için 3 ACK alırsa segmentin kaybolduğunu varsayar ve timeout süresi uzun olduğu için timeout’u beklemeden retransmit eder yani tekrar gönderir. * Timeout süresi çok uzun olursa kayıp paketleri retransmit etmek çok gecikir.  40.4. TCP Flow Control (Akış Kontrolü)  * Gönderilen paketlerin buffer’a gelme hızı çıkma hızından yüksek olursa buffer’da paketler birikmeye başlar.   TCP alıcı kendi buffer’ı dolmadan göndericiyi uyarır (flow control).  Akış denetimi hız uyumlama servisidir. Akış denetimini alıcı başlatır ve yönetir.   * **Tıkanıklık denetimini (congestion control)** gönderici kendisi başlatır ve yönetir.   **Last Byte Rcvd:** Buffer’a son eklenen byte.  **Last Byte Read:** Process’in buffer’dan son okuduğu byte.  **rwnd:** Segmentin içindeki receive window alanına yazılır.   40.5. TCP ve UDP Karşılaştırılması  |  |  |  | | --- | --- | --- | | **SERVİS** | **TCP** | **UDP** | | **Bağlantı Kurulumu** | Zaman alır ancak güvenilir şekilde bağlantı kurulur. | Bağlantıya gerek yoktur. | | **Teslim Garantisi** | Gönderildiğini onaylar. (ACK) | UDP’de onay mekanizması yoktur ve kaybolan paketler tekrar iletilmez. | | **Paket Ardışıklığı**  **(Paketlerin Doğru Sırası Hakkında Bilgi)** | Paketler ardışık numaralandırılmıştır. | UDP, paketlerine numara verilmez. Paketlerin ulaştığı veyahut kaybolduğu düşünülür. | | **Akış Kontrolü** | Alıcı, göndericiye yavaşlaması için sinyal gönderebilir. | Paket akış kontrolü için TCP’de kullanılan onay UDP’de geri dönmez.  Yani, akış kontörlü yoktur. | | **Tıkanıklık Kontrolü** | Network cihazları TCP onayları sayesinde göndericilerin tavrını kontrol edebilir. | Onay olmadan network tıkanıklık sinyali gönderemez. |  * **AF\_INET** **= IPV4**  **AF\_UNIX** **= Unix dosyası AF\_INET6** **= IPv6** * **SOCK\_STREAM** **= TCP Socket** * TCP connection’da server’ın açık olması gerekmektedir. * **serverSocket.listen(1**) **=>** Aynı anda kaç tane client’ın dinleneceğini gösterir. * TCP’de **welcoming socket** ve **connection socket** var. |

|  |
| --- |
| 41. Güvenilir Veri Transferi Prensipleri (Reliable Data Transfer - RDT)  * Güvenilir veri transferi protokolün gönderici ve alıcı tarafları incelenmelidir. Veri transferin tek yönlü (unidirectional) olduğunu düşünülürse, sadece **kontrol (checksum**) bilgisi iki yönlü gitmektedir. * Güvenilir veri aktarımı **transport layer’da, network layer’da** veya **link layer’da** oluşturulabilir. * Sonlu durum makineleriyle **(Finite State Machines)** modellenebilir.    42. Stop and Wait Protocol  * Gönderici bir paket gönderir, ardından alıcının yanıtını bekler. Stop and wait, half-duplex modda çalıştığından TCP tarafından kullanılmamaktadır. Çünkü stop and wait durumunda alıcıdan bir kabul gelmeden, TCP yeni bir iletim yapamaz.  42.1. Rdt 1.0: Güvenilir Kanal Kullanarak Güvenilir Transfer  * Altyapıdaki kanal tümüyle güvenilirdir.  1. Bit hatası yoktur. 2. Kayıp paket yoktur. 3. Gönderici ve alıcı sonlu durum makineleriyle modellenebilir. 4. Gönderici üst katmandan gelen veriyi pakete dönüştürüp alt katmandaki kanala gönderir. 5. Alıcı alt katmandan aldığı paketten veriyi çıkarıp üst katmana gönderir  42.2. Rdt 2.0: Bit Hatası Olan Kanal ile Çalışma  * Alt katmandaki kanalda paket içindeki bitler bozulabilir. Bu bit hatalarını kontrol etmek için **checksum** kullanılır. Bozulan paketler alıcı tarafından tekrar istenir **(Automatic Repeat reQuest (ARQ) protokolleri).**   Hata düzeltilmesi:   1. **ACKnowledgements (ACKs):** Alıcı göndericiye aldığı paketin hatasız olduğunu iletir. 2. **Negative ACKnowledgements (NACKs):** Alıcı göndericiye aldığı paketin hatalı olduğunu bildirir. 3. Gönderici NAK ile bildirilen paketi tekrar gönderir **(retransmission)**.  42.3. Rdt 2.0’daki Yenilikler (Rdt 1.0’a göre):  1. Hata algılama eklendi. 2. Alıcı geri bildirimi için kontrol mesajları(ACK, NACK).  42.4. Rdt 2.0’da Karşılaşılan Problemler  * ACK/NACK bozulursa:  1. Gönderici, alıcı tarafından doğru alınıp alınmadığını bilemez. 2. Retransmit (Yeniden İletim) yapılamaz ve duplicate olabilir.  * Duplicate’lerin (Çiftlerin) seçilmesi için:  1. Göndericide her pakete sıra numarası eklenir. 2. Pakete yeni bir alan eklenir (sequence number). 3. Gönderici ACK/NACK bozulursa retransmit yapar. 4. Alıcıda aynı sıra numarasına sahip tekrarlı gelen paketler atılır.  42.5. Rdt 2.1 **Gönderici**   1. Sıra numarası alanı pakete eklenir. 2. Minimum bir bit ile (0, 1) sıra numarası verilebilir **(alternating-bit protocol)**. 3. Alınan ACK/NACK paketin bozuk olup olmadığı kontrol edilir.   **Alıcı**   1. Gelen paket çift mi kontrol edilir. Çift gelen paketler atılır. 2. Bulunulan durum gelen paket için sıra numarası olur. (0 veya 1 olarak bekler)  42.6. Rdt 2.2 NACK Kullanılmayan Protokol  * ACK kullanarak rdt 2.1 ile aynı işlevi yapar.  1. NACK yerine, alıcı en son doğru alınan paket için ACK paket gönderir. 2. Alıcı paketin sıra numarasını bilmelidir. 3. Alıcıdaki duplicate ACK paketin retransmit edilmesini sağlar.  42.7. Rdt 3.0: Kanallarda Hata ve Kayıp  * Alt kanallarda paketler kaybolabilir (Data veya ACK). * Paketin kaybolduğunun anlaşılması için gönderici tarafa timer eklenir. Gönderici bir süre bekler ACK gelmezse retransmit yapar. * Eğer paket (veya ACK) kaybolmamışsa ve gecikmiş olarak gelirse:  1. Retransmit ile duplicate oluşur ancak sıra numarasıyla belirlenir. 2. Alıcı ACK gönderdiği paketin sıra numarasını belirlemelidir. 3. Bekleme süresi sonunu belirlemek için aşağı doğru sayıcı gereklidir.      * rdt 3.0 protokolü **stop-and-wait** çalışır ve performansı çok düşüktür. * Stop-and-wait yerine **pipelining** yapılarak kullanım oranı artırılır.  43. Pipeline (Boru Hattı) Mimarisi  * Pipeline, birden fazla işin paralel olarak aynı anda yürütülmesidir. Pipeline, bir işin süresini değiştirmiyor yapılan iş miktarını arttırıyor. * Pipelining ile birden fazla paket ACK alınmadan art arda gönderilir.   Sıra numarası aralığı artırılmak zorundadır.  Gönderici ve alıcıda sırasız gelen paketler için buffer gereklidir.  İki temel pipelining protokolü vardır: [Go-Back-N](#goBackN) ve [Selective Repeat](#selectiveRepeat) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 44. Go-Back-N (Go Back N) Protokolü  * TCP herhangi bir nedenden dolayı gönderdiği segment için hata algıladığında yeniden iletim yapmaktadir. TCP yeniden iletimi 2 sebepten dolayı yapar. Ya paketler kaybolur ya da paketlerin bozulur.   Paketlerin kaybolmasını veyahut bozulmasını engellemek için alıcı ve gönderici taraf birbirlerine çeşitli mesajlar gönderir ve iki tarafta da çeşitli kontroller yapılır.  Alıcı aldığı paket için göndericiye **Acknowledgements =** [**ACK,**](#ACK) **gönderir.**  hatalı paketi aldığına dair de **Negative Acknowledgements = NAK** mesajı gönderir.   * Go-Back-N protokolü, **sürgülü pencere protokolüdür**. Veri bağlantı katmanındaki hatayı algılayan ve kontrol eden bir mekanizmadır. * Bu protokolde alıcı, göndericinin verilerini sıralı bir şekilde almak ister. Yani göndericinin 10 paket gönderdiğini ve gönderim sırasında 5. paketin kaybolduğunu düşünelim. Alıcı ilk 4 paket için ACK onay mesajını gönderecektir. 5.paketi almadan  6, 7, 8, 9 ve 10.'cu paketler alıcıya geldiğinde TCP bu paketleri için onay mesajı yani ACK göndermeyecektir. Bu paketler elimine edilecektir (yok edilecek) ve  alıcı, 5.paket için göndericiye sürekli ACK gönderecektir. Ve ben 5. pakete kadar olanı aldım bana 5 ve sonrasını tekrar gönder diyecektir. 5. paket gönderici tarafından yeniden gönderildikten sonra  gönderici daha önce göndermiş olduğu 6, 7, 8, 9 ve 10.'cu paketleri yeniden gönderecektir. Bu durumun hem avantaj hem de dez avantajları bulunmaktadir. **Avantajı,** alıcı düğüme her paket sıralı bir şekilde gelmiş olacağından ve sıralı gelmeyen paketler elimine edileceğinden dolayı, alıcının tamponu şişmeyecektir ve esnek bir şekilde kullanılmış olunacaktır.   **Dezavantajı** **ise,** gönderici gönderdiği paketleri tekrar gönderdiğinden dolayı iletim ortamı etkin kullanılmamış olunacaktır.   * Aşağıdaki şekillde, alıcı 5. paketi almadığından 4 ve 4'ten sonra gelen her paket için göndericiye ACK 'si 5 olan paketi gönderiyor. Ben 5'e kadar olanı aldım bana 5 ve sonrasını gönder demek istiyor şeklimiz.    45. Selective Repeat (Seçici Tekrar) Protokolü  * Bu algoritmada alıcı, göndericinin göndermiş olduğu paketleri sıralı bir şekilde alma koşulu aramaz. Alıcı, arada kaybolan bir paketin oluşması durumunda sadece ilgili paketin iletimini ister. Fakat sıralı bir şekilde tam alım olmadığı için alıcının tamponu 1 veri çerçevesini depolayabilecek boyuttan büyük olmalıdır. * Örnek olarak  göndericinin 10 paket gönderdiğini ve gönderim sırasında aradaki 5. paketin kaybolduğunu düşünelim. Alıcı, aldığı ilk 4 paket ve 6, 7, 8 ,9 10'uncu paketler için ACK onayını gönderecektir. Gönderici 5. paketin ACK onayını almamıştır. Yeniden iletim zaman aşımına gelindiğinde halen göndericiye 5. paket için onay gelmediğinden, gönderici 5. paketi yeniden gönderecektir. Fakat yalnızca kaybolan paket olan 5. paketi gönderilecektir. Bu durumun hem avantaj hem de dezavantajlari bulunmaktadır. **Avantajı,** yalnızca kaybolan paket yeniden gönderileceğinden iletim ortamı esnek bir şekilde kullanılmış olunacaktır.   **Dezavantajı ise,** alıcıya paketler sıralı bir biçimde gitmediğinden alıcının tamponu esnek kullanılmamış olacaktır. Ve çok kayıp paket olduğu düşünülürse tampon zamanla şişecek ve yeni paketler için yer kalmayacaktır.   * Alıcı 5. paketi almadığından 4 ve 4'ten sonra gelen her paket için göndericiye ACK'si 5 olan paketi gönderiyor. Bana 5. paketi gönder demek istiyor.      |  |  |  | | --- | --- | --- | |  | **Go-Back-N** | **Selective Repeat** | | **Temel** | Zarar gördüğü veya kaybolduğundan şüphelenilen kareden sonra gönderilen tüm paketleri yeniden iletir. | Yalnızca kaybolduğundan veya hasar gördüğünden şüphelenilen paketleri yeniden iletir. | | **Bant Genişliği** | Hata oranı yüksekse, çok fazla bant genişliği boşa harcar. | Daha az bant genişliği israf edilir. | | **Karmaşa** | Daha az karmaşık. | Daha fazla karmaşık. | | **Pencere Boyutu** | N-1 | ≤ (N + 1) / 2 | | **Sınıflandırma** | Sıralama, gönderici tarafında veya alıcı tarafında gerekli değildir. | Alıcı, çerçevelerin sırasını koruyacak şekilde sıralayabilmelidir. | | **Depolama** | Alıcı, hasar görmüş paket yeniden iletilinceye kadar hasarlı paketten sonra alınan paketleri saklamaz. | Alıcı, hasarlı paketi değiştirilinceye kadar hasarlı çerçeveden sonra alınan çerçeveleri tamponda saklar. | | **Arama** | Gönderici tarafında veya alıcıda paket aranmasına gerek yoktur. | Gönderenin yalnızca istenen paketi arayabilmesi ve seçebilmesi gerekir. | | **ACK Numaraları** | NAK numarası için, bir sonraki beklenen paketin numarasına bakılır. | NAK numarası, kaybedilen pakete karşılık gelir. | | **Kullanım Sıklığı** | Daha sık kullanılır. | Karmaşıklığı nedeniyle pratikte daha azdır. |   Selective repeat’te, pencere boyutunun **(yani** **window size:** **gönderilecek paket sayısı)** sıra numarasının yarısından eşit veya küçük olması lazım. Sıra numarası dediğimiz yukarıdaki örnek için 10 paketin göderilmesi pencere boyutu ise gönderilen mangası yani 5’er 5’er gönderilmesi gibi.  Selective repeat, uygun şekilde alınan çerçeveler için bant genişliğini boşa harcamadığı için daha verimli bir protokoldür, ancak karmaşıklığı ve masrafı, geri dönüş protokolünün kullanılmasını desteklemektedir. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 46. Tıkanıklık – Sıkışıklık Kontrolü (Congestion)  * Ağa gönderilen paket sayısı ağın kapasitesinden fazla ise de tıkanıklık meydana gelir. Bu sebeple trafik oluştuğunda performansda düşüşler meydana gelir ve aşağıdaki durumlar oluşur:   **Kayıp paketler** oluşur. (Router’da arabellek taşması olduğunda paketler kaybolabilir.).  **Uzun gecikmeler** meydana gelir. (Paketler, router arabelleklerinde sıraya girerler.)   * Tıkanıklık meydana geldiğinde; öncelikle sistem izlenir, tıkanıklığın meydana geldiği yer tespit edilir. Tıkanıklığın giderilebileceği yerlere bilgi iletilir.  46.1. Tıkanıklığı Önleme Politikaları  |  |  | | --- | --- | | **KATMAN** | **POLİTİKA** | | **Taşıma (Transport) Katmanı** | Yeniden gönderim Sırasız ön belleğe alma  ACK geri bildirim Akış kontrolü  Timeout belirlenmesi | | **Ağ (Network) Katmanı** | Alt ağlarda sanal devreler oluşturma  Paketleri kuyruğa alma ve hizmet verme  Paket görmezden gelme  Yönlendirme Algoritması  Paket yaşam süresi belirleme | | **Veri Bağı (Data Link) Katmanı** | Yeniden gönderim Sırasız ön belleğe alma  ACK geri bildirim Akış kontrolü |  46.2. Tıkanıklık Denetimi  * TCP protokolü timer kullanarak kayıp paketlerin tekrar gönderilmesini sağlar.   Paketlerin tekrar gönderilmesi ağda tıkanıklık olduğunun bir göstergesidir.   * Paket kayıpları genellikle router buffer’ının dolu olmasından kaynaklanır. * Ağdaki tıkanıklığın nedenini ortadan kaldırmak için göndericilerde yeni mekanizmalara ihtiyaç vardır.  46.3. Tıkanıklığın Nedenleri ve Maliyeti **Senaryo-1: İki gönderici, bir router ve sınırsız buffer**   * İki host bir router’ı (sınırsız buffer) paylaşarak kullanmaktadır.   [Retransmit](#retransmissionNedir), akış denetimi ve tıkanıklık denetimi yapılmıyor.  Host A ve Host B ağa  **(gönderilen veri miktarı)** byte/s veri göndermektedir.  Alıcıya byte/s veri ulaşıyor.   * Soldaki grafik her bağlantı için alıcıdaki [throughtput’u](#throughput) **()** gösterir.   Sağdaki grafik **(gönderilen veri miktarı)** değerine göre gecikmeyi gösterir.  Veri gönderim oranı R/2’den fazla olursa buffer’a alınır.  Buffer’daki veri arttıkça gecikme süresi sonsuza doğru artar.  Tıkanıklığın maliyeti gecikme süresindeki artıştır.    **Senaryo-2: İki gönderici, bir router ve sınırsız buffer**   * Router sınırlı buffer’a sahiptir.   Buffer dolu ise paketler atılıyor, gönderici retransmit yapıyor.  Uygulama katmanı byte/s ulaşım katmanına gönderir.  ’nin orijinal veri ve retansmit yapılan verinin toplamıdır.   * Kayıp paket hiç olmazsa  **=** olur (soldaki grafik).   Kaybolduğu kesin olanlar tekrar gönderilebilir (ortadaki grafik).  Retransmit oranı arttıkça oranı azalır.  Burada tıkanıklığın maliyeti kayıpların retransmit edilmesidir.     * Erken timeout ile paketler retransmit edilebilir (sağdaki grafik).   Tekrar alınan paketler alıcıda atılır ve oranı azalır.  Router’da ve diğer bileşenlerde gereksiz iş yapılır.  Burada tıkanıklığın maliyeti erken timeout’tan dolayı gereksiz retransmit yapılmasıdır.  **Senaryo-3: Dört gönderici, router’lar (sınırlı buffer), multihop yol**   * Tüm host’lar iki [hop](#HOPS) yola sahiptir.   Tüm host’lar byte/s veri göndermektedir.  Tüm linkler R byte/s bant genişliğine sahiptir.  Host A, Host C ile R1 ve R3 üzerinden haberleşmektedir.  A, C, R1 linkini D B ile R2 linkini B D ile paylaşmaktadır.   * Küçük değerleri için buffer taşması nadiren olur **( = )**   arttıkça artar.  ve arttıkça atılan paket artar ve sıfıra doğru düşer.   46.4. Tıkanıklık Denetimi Yaklaşımları  * Tıkanıklık denetimi ağdaki cihazlar tarafından geri bildirim kullanılarak yapılırsa ağ trafiğini artırır.   Tıkanıklık denetimini uç birimler tarafından yapılırsa kayıp paket ve gecikmelere göre analiz yapılır.   1. Ağ Destekli Tıkanıklık Denetimi (Network-Assisted Congestion Control)  * Ağ katmanı bileşenleri (router) göndericiye **doğrudan veya dolaylı bir şekilde ağdaki tıkanıklığı bildirebilir**. * Geri bildirim bir bit kullanılarak yapılabilir. * Router doğrudan göndericiye kendisinin çıkış linkinin desteklediği gönderim oranını da bildirebilir. * Günümüzde IBM SNA, DEC, DECnet ve ATM ağlarda kullanılmaktadır. * Tıkanıklık bilgisi ağdaki cihaz tarafından iki yolla göndericiye iletilebilir.  1. **Alıcı Üzerinden Bildirim**   Alıcı üzerinden bildirimde, router üzerinden geçen paketi tıkanıklık bitini işaretleyerek gönderir.  Alıcı geri bildirimle göndericiye tıkanıklık bildiriminde bulunur (akış denetimine benzer).   1. **Doğrudan Bildirim**   Doğrudan bildirimde, router doğrudan göndericiye bir paket ile bildirim yapar **(choke packet)**.  Doğrudan router tarafından bildirimde geri bildirim süresi daha kısadır.   1. Uçtan Uca Tıkanıklık Denetimi (End-End Congestion Control)  * Ağ katmanının, ulaşım katmanına doğrudan destek sağlamadığı durumlarda kullanılır. * Ağdaki **tıkanıklık uç birimlerde paket kaybı veya gecikme kullanılarak algılanır**.   Ağdaki tıkanıklık denetimi için **tıkanıklık penceresi (congestion window (cwnd))** kullanır.   * TCP, uçtan uca tıkanıklık denetimi yapar.   TCP segment kaybı ağdaki tıkanıklığa işaret sayılır ve gönderim hızı azaltılır.  TCP cwnd ile ağa veri gönderme hızını sınırlar.   * IP, uç sistemlere ağdaki tıkanıklıkla ilgili geri bildirim yapmaz. * RTT süresi arttıkça tıkanıklığa işaret sayılır ve gönderim hızı azaltılır. * ACK alınmadan gönderilecek maksimum veri miktarı **min(cwnd, rwnd)** olmalıdır **(receive window (rwnd))**   **LastByteSent – LastByteAcked ≤ min(cwnd, rwnd)** 46.5. TCP Tıkanıklık Denetimi  * TCP gönderici timeout olduğunda veya 3 kez art arda ACK aldığında tıkanıklık olduğunda karar verir. * TCP aldığı ACK’lara göre cwnd boyutunu düzenler **(self clocking)**: * Hiç paket kaybı olmaksızın ACK almaya devam ederse, RTT süresine göre cwnd boyutunu artırır. * ACK alma süresi uzarsa, cwnd boyutunu düşürür. * ACK alma süresi aniden çok kısalırsa, cwnd boyutunu hızla artırır; aniden çok uzarsa cwnd boyutunu hızla azaltır.   TCP, cwnd boyutunu değiştirerek veri gönderme hızını ayarlar.  TCP, ACK aldığı sürece gönderme hızını sürekli artırır.   * TCP tıkanıklık kontrol algoritması üç bileşene sahiptir:   + Yavaş başlatma (slow start)   + Tıkanıklıktan kaçınma (congestion avoidance)   + Hızlı toparlanma (fast recovery)  46.6. Yavaş Başlatma (Slow Start)  * TCP başladığında cwnd değeri 1 MSS (maximum segment size) alınır.   **MSS = MTU – TCPHeader – IPHeader**   * MTU datalink layer payload boyutudur. * Hiç ACK almadan veri gönderim oranı MSS/RTT alınır   MSS = 500 byte RTT = 200 ms cwnd = 20 kbps olur.   * TCP, her ACK aldığında cwnd değerini 1 MSS artırır. * Slow start ilk paket kaybında sona erer ve cwnd 1 MSS yapılarak başa alınır. * **Ssthresh** (slow start threshold) cwnd/2 yapılır ve TCP congestion avoidance moduna geçer.  46.7. Tıkanıklıktan Kaçınma (Congestion Avoidance)  * Congestion avoidance moduna geçildiğinde, cwnd değeri en son kayıp olduğundaki değerinin yarısına eşitlenir. * TCP her ACK aldığında, cwnd değerini MSS/RTT kadar artırır. * Her alınan RTT değerine göre gönderme hızı yeniden hesaplanır. * MSS = 1460 byte, cwnd = 14600 byte ise, 10 segment RTT süresinde gönderilebilir. * TCP, kayıp paket olduğunda cwnd=1, sstresh=cwnd/2 yapar. * TCP, art arda üç ACK geldiğinde (fast retransmit yapar), cwnd=cwnd/2 yapar ve fast recovery durumuna geçer.  46.8. Hızlı Toparlanma  * Fast recovery durumunda iken;   Kayıp paket için negatif ACK geldiğinde, cwnd değeri 1 MSS artırılır ve fast recovery durumu devam eder.  Yeni bir pozitif ACK geldiğinde, cwnd=ssthresh yapılır ve congestion avoidance durumuna geçilir.  Timeout olursa ssthresh=cwnd/2 ve cwnd=1 yapılır ardından slow start durumuna geçilir.   * TCP, cwnd recovery için ilk versiyonlarında ilk kayıp algılamada cwnd=1 yapılmaktaydı. **(Tahoe)** * TCP yeni versiyonda ilk kayıp algılamada cwnd=cwnd/2 yapılmaktadır. **(Reno)** * TCP Tahoe ve Reno’da ssthresh değerine kadar cwnd üssel artar, ssthresh değerinden sonra doğrusal artar. * İlk kayıp pakette, Tahoe cwnd değerini 1, Reno ise cwnd/2 yapar.      * TCP, cwnd boyutunu çalışma süresince toplama yaparak artırır, her kayıp algılamasında yarı yarıya düşürür (additive increase, multiplicative decrease AIMD)    47. QUIC Protokolü (HTTP/3 Protokolü)  * Kısaca QUIC Protokolü, HTTP protokolünün UDP üzerinden taşınarak hızlandırılmasını amaçlamaktadır. |

|  |
| --- |
| 48. Ağ Katmanı (Network Layer)  * Ağ katmanının görevi, gönderici host’tan alıcı host’a paket gönderiminin en uygun düğümlerden geçirilip yönlendirilmesini sağlamaktır. * Ağ katmanında iki temel işlev yapılır:  1. Forwarding (İletim)   Router’ın giriş linkine gelen paketin belirlenen çıkış linkine gönderilmesidir.  Her router bir forwarding tablosuna sahiptir.  Paket, adres başlık bilgisine göre gönderilir. Aşağıdaki resimde örneği verilmiştir.   1. Routing (Yönlendirici   Router’a gelen paket için alıcıya gidebileceği yol oluşturulur ve yönlendirme yapılır.  Yolun maliyetini routing algoritmaları hesaplar.   * **Traditional routing algoritmalar,** router’larda implement edilir.   **Software defined networking (SDN),** farklı server’da implement edilir yani fiziksel olarak [control plane](#controlPlane), [data plane’den](#dataPlane) ayrılmıştır.   * Bir paket başlık bilgisine göre tabloda belirlenen çıkış linkine gönderilir. Örneğin 0111 başlık bilgisini içeren paket 2 numaralı router’a gider.      * Forwarding tablosu **merkezi** veya **dağıtık** olarak güncellenebilir.   **Merkezi güncellemede** bir algoritma tüm router tablolarını günceller.  **Dağıtık güncellemede** her router kendi tablosunu günceller.   * Bazı paket anahtarlar forwarding kararını **link layer frame adres alanlarındaki** değerlere göre yapar (link layer switch veya layer 2 switch).   Bazı paket anahtarlar forwarding kararını **network layer paket alanlarındaki** değerlere göre yapar (router veya layer 3 switch). Layer 3 anahtarlar (router) layer 2 servislerini bulundurmak zorundadır. 48.1. Ağ Katmanı Servis Modelleri  * Ağ katmanı servis modeli **end to end (uçtan uca)** paketlerin nasıl taşınacağını belirler. * Ağ katmanı, ulaşım katmanına farklı servisler sağlayabilir:   **Garanti Edilmemiş Gönderim:** Paketin alıcıya ulaşmasını garanti etmez.  **Garanti Edilmiş Gönderim:** Paketin alıcıya ulaşmasını garanti eder.  **Sınırlı Gecikme İle Garanti Edilmiş Gönderim:** Paketin alıcıya sınırlandırılmış sürede (Örn. 100 ms içinde) ulaşmasını garanti eder.   * Ağ katmanı, paketlerin taşınması sırasında aşağıdaki servisleri sağlayabilir:   **Sıralı Paket Gönderimi:** Paketlerin çıktıkları sırayla alıcıya ulaşması sağlanır.  **Garanti Edilmiş Minimum Bant Genişliği:** Gönderici ve alıcı arasında bit rate garanti edilir. Belirli bir bit oranının altında (Örn. 1Mbps) gönderim yapılırsa, kayıp paket olmaz ve belirlenmiş host to host gecikmeyle alıcıya ulaşır.  **Garanti Edilmiş Maksimum Jitter:** İki ardışık paketin göndericiden çıkış süreleri ile alıcıya varış süreleri eşittir veya belirlenen değerin üzerinde olamaz.  **Güvenlik Servisleri:** Paketler şifrelenmiş bir şekilde gönderilebilir.   * İnternet ağ katmanı **best effort servis’i** (en iyi çabayla iletim servisi) amaçlar yani minimum sürede maksimum gönderimi amaçlar. * ATM ağ katmanında temel olarak iki tür servis sağlar:  1. CBR (Constant Bit Rate)   CBR, sanal bir yol oluşturarak tüm paketleri (ses – video) (ATM hücreleri) aynı yoldan taşır.  End to end delay, jitter, kayıp paket oranı belirlenen değer için garanti edilir.   1. ABR (Available Bit Rate)   İnternet’in [best effort servisine](#bestEffortService) benzer.  ATM hücreleri kaybolabilir. ATM hücreleri alıcıda tekrar sıralanmaz.  Minimum cell transmission rate **(MCR)** garanti edilir.  Ağ belirli bir süre yeterli kaynağa sahipse, gönderici MCR’den yüksek gönderim yapabilir.  ATM ABR servisi, göndericiye tıkanıklık bildirimi biti gönderir., 48.2. Datagram Ağlar  * Ulaşım katmanı gibi, ağ katmanı da host’lar arasında connectionless (bağlantısız)veconnection (bağlantılı) servis sağlayabilir. Ağ katmanı connection servisi ile handshake yapar. * Ağ katmanı ve ulaşım katmanındaki connectionless ve connection servisleri arasında önemli farklılıklar vardır:   Ağ katmanı **host to host** iletişim yapar. Ulaşım katmanı **process to process** iletişimini, ağ katmanının host to host servisini kullanarak yapar.   * Tüm önemli bilgisayar ağ mimarileri (ATM, Frame Relay, İnternet) ya connection servis ya da connectionless servis sağlar. İkisini birlikte aynı anda sağlayamaz.   Ağ katmanında connection servis sağlayanlar **sanal devre (virtual circuit)** ağları olarak adlandırılır.  Ağ katmanında connectionless servis sağlayanlar **datagram ağlar** olarak adlandırılır.  Ulaşım katmanı connection oriented servisi **uç sistemlerde**, ağ katmanı connection servisi **router’larda** oluşturur.   * Datagram ağlarda, uç sistem; paketi hedef adresiyle birlikte ağ katmanına gönderir. * Paket, göndericiden alıcıya bir grup router üzerinden ulaşır.   Her router, paketi hedef adresine göre **forwarding** yapar.  Her router bir forwarding tablosuna sahiptir.   * Router üzerinde [**prefix**](#prefix) değere göre gönderme yapılır.      * **Datagram ağlarda** router’lardaki forwarding table routing algoritmaları tarafından güncellenir.   **Virtual circuit ağlarda** router’lardaki forwarding table her yeni bağlantı kurulumunda veya bağlantı sonlandırmada güncellenir (tier-1 router’da 1μs sürede).  *Datagram ağlarda router forwarding tablosu her an değişebileceğinden paketler hedefe farklı sırada ulaşabilir.* 49. IP (Internet Protocol) Nedir ve Nasıl Çalışır  * **IP,** iki bilgisayar arasındaki paketlerin yönlendirilmesini sağlayan bir protokol olarak tanımlanır diğer bir adıyla **yönlendirme protokolüdür**. IP’de verinin niteliği önemli değildir sadece **verinin gideceği adresi belirler**. * IP, veriyi karşı tarafa yönlendirirken, alıcının bu veriyi kabul edeceği veya etmeyeceği konusunda bir doğrulama yapmaz. Hata kontrolü bir üst katmanın işidir ve bu sayede IP kendi başına çalışabilen bir protokoldür. * IP protokolü: Adresleme ve datagram formatını belirler. IP (Internet Protocol), IPv4 ve IPv6 olarak iki versiyona sahiptir. * **Routing protokolleri:** Kaynak ve hedef arasında yolu belirler. * **Rapor servisleri:** Datagram hatası ve ağ katmanı bilgilerine yönelik servisleri sağlar.    50. IPv4 Datagram Formatı **Version:**   * 4 bit ile IP protokol versiyonunu belirler. * Router datagram’ın kalan kısmını yorumlarken kullanır.   **Header length:**   * Payload kısmın nereden başladığını belirlemek için kullanılır. * IPv6’da kaldırılmıştır.   **Type of service:**   * Düşük gecikme, yüksek throughput veya reliability gibi datagram türlerini belirler.   **Datagram length:**   * Datagram’ın toplam uzunluğunu 16 bit ile belirler. Maksimum 65.535 byte.   **Identifier, flags, fragmentation offset:**   * IP datagram’ların parçalanıp birleştirilmesinde kullanılır. * IPv6 parçalamaya izin vermez. * Identifier her datagram için artırılarak atanır. * **Flag bit** (0) datagram’ın son parçasını gösterir. * **Offset,** parçanın datagram içinde offset’ini gösterir (8 byte chunk).   **Time to live (Yaşam Süresi):**   * Datagram’ın sonsuza kadar ağda dolaşmasına izin verilmez. * TTL her router’da 1 azaltılır ve 0 olunca datagram atılır.   **Protocol:**   * TCP için 6, UDP için 17 değerine sahiptir.   **Header checksum:**   * Router’ın datagram’daki bit hatalarını algılamasını sağlar (2-byte). * Hatalı datagram’lar router’da atılır. * TTL değiştiği için her router’da tekrar hesaplanır. * IPv6’da kaldırılmıştır.   **Source and destination IP addresses:**   * Kaynak ve hedef IP adresleridir. * Hedef IP adresi kaynak tarafından DNS protokolü ile elde edilir.   **Options:**   * IP header’ı genişletmek için kullanılır. * IPv6’da kaldırılmıştır.   **Data (payload):**   * Genellikle TCP/UDP segmentini taşır.    51. IPv6 Datagram Formatı **Version:**   * IP versiyon numarasıdır. IPv6 için değeri 6’dır.   **Traffic class:**   * 8 bit ile trafik önceliklendirilir.   **Flow label:**   * 20 bit ile datagram’ları tanımlamak için kullanılır.   **Payload length:**   * 16 bit ile başlıktan sonraki veri boyutunu belirler.   **Next header:**   * Üst katman (ulaşım katmanı) protokolünü (TCP veya UDP) belirler.   **Hop limit:**   * Datagram’ın ağdaki yaşam süresini (hop count) belirler. * Değeri 0 olduğunda atılır.   **Source and destination addresses:**   * Kaynak ve hedef host içi 128 bit adreslerdir.   **Data:**   * Payload datayı gösterir. * IPv6, temel özelliklerde birtakım önemli değişiklikler getirmiştir.   **Expanded addressing capabilities**   * IPv6 adres boyutu 128 bit’tir. * IPv6, *unicast* ve *multicast’in* yanı sıra *anycast* adresleme yapabilir.   **A streamlined 40-byte header**   * 40 byte sabit header boyutu vardır.   **Flow labeling and priority**   * IPv6 trafikte önceliklendirme yapabilir (real time ses ve video)   **Fragmentation/reassembly**   * IPv6 parçalamaya izin vermez. * Parçalamayı kullanan DoS saldırılarına dayanıklıdır.   **Header checksum**   * IPv6’da çıkartılmıştır.   **Options**   * IPv6’da çıkartılmıştır. |

|  |
| --- |
| 52. IP Datagram Fragmentation  * Tüm link layer (bağlantı katmanı) protokolleri network layer paketlerini aynı boyutta taşıyamaz.   Link layer’daki maksimum veri boyutu **maksimum transmission unit (MTU)** olarak adlandırılır.   * Kaynak host ile hedef host arasındaki link layer protokolleri farklı olabilir. * IP datagram, link layer çerçeve veri boyutuna göre parçalanır **(fragmentation).** * Parçalanan datagram’ın parçaları alıcı host’ta, identification, flag ve fragmentation offset alanları kullanılarak birleştirilir.    53. IPv4 Adresleme  * Her host ve router arayüzü (ucu) bir IP adresine sahiptir. * **3. katman protokolüdür. Connectionless (bağlantısız) bir protokoldür. Paketin hedefe ulaşmasını garanti etmez.** * IPv4, 32 bitlik adresden oluşur. Bu 32 bitlik adresi 4’e bölüp 8’erli parçalara ayırırız bunlara da **oktet** denir.   193.32.216.9 = 11000001 00100000 11011000 00001001   * ISP’ye IP adresi ataması **ICANN (Internet Corporation for Assigned Names and Numbers)** tarafından yapılır.   Türkiye’deki IP adreslerini ise **RIPE** şirketi dağıtır.   * Aşağıdaki örnekte 2 mantıksal ağ, 1 fiziksel ağ (kırmızılı kısım şimşek gibi olan) bulunmaktadır.      * Her host için IP adres ataması [alt ağa (subnet mask)](#subnetMaskNedir) göre belirlenir. Bir ağdaki tüm arayüzler **aynı alt ağ (subnet)** adresine sahiptir. * Sol üstteki alt ağlar "223.1.1.x" sağ üstteki alt ağlar "223.1.2.x" ve alttaki alt ağlar "223.1.3.x" adreslerine sahiptir. Sol üstteki alt ağ için adres 223.1.1.0/24 olarak ifade edilir.   /24 **subnet mask** (router’a gitmeden IP ve MAC adresini bilerek ağ içinde haberleşme) olarak adlandırılır.     * İnternet, [**CIDR**](#CIDR) adres atama yöntemini kullanır.   **CIDR,** 32-bit IP adresini [ağ adresi](#networkIDnedir) ve [host adresi](#hostIDnedir) olarak ikiye böler.  a.b.c.d/x için x değeri adresin **öneki (prefix)** olarak adlandırılır.  32-bit’in (32-x) biti host adresidir.     * Bir firma ISP değiştirdiğinde, ISP subnet mask adresini ekler. * CIDR adreslemeden önce **classfull** adresleme yapılıyordu.   **Classfull** adreslemede **prefix 8 (A sınıfı), 16 (B sınıfı)** veya **24 (C sınıfı)** olabilir.   * Class A - (0.0.0.0 / 8 - 127.0.0.0 / 8)   16 milyondan fazla ana bilgisayar adresine sahip son derece büyük ağları desteklemek üzere tasarlanmıştır. Sabit / 8 öneki (255.0.0.0) kullanır.   * Class B - (128.0.0.0 / 16 – 191.255.0.0 / 16)   Yaklaşık 65,000 ana bilgisayar adresine kadar orta ve büyük boyutlu ağların ihtiyaçlarını desteklemek üzere tasarlanmıştır. Sabit / 16 öneki (255.255.0.0) kullanır.   * Class C - (192.0.0.0 / 24 - 223.255.255.0 / 24)   En fazla 254 ana bilgisayara sahip küçük ağları desteklemek üzere tasarlanmıştır. Sabit / 24 öneki (255.255.255.0) kullanır.  Bir firma 2.000 hosta sahipse ve B sınıfı bir adres kullanıyorsa, 65.534-2.000 = 63.534 adres kullanılmadan boş kalır. 53.1. Statik ve Dinamik IPv4 Adresleme  * IPv4 adresleri statik veya dinamik olarak atabilir. * **Statik IPv4 Adres Atama** * Kullanıcı tarafından el ile girilen IP adresidir. IP adresi sabit kalması istenen sistemler için kullanılmaktadır. * Statik bir atama ile cihaza IPv4 adresinin yanısıra alt ağ maskesi, varsayılan ağ geçidi ve DNS sunucusu adres bilgileri manuel olarak verilir. * Statik IPv4 ataması genellikle, yazıcılar, sunucular ve belli bir hizmeti sağlamak üzere sabit bir IP adresine sahip olması istenen diğer ağ aygıtları için tercih edilir. * Manuel adres ataması sırasında hata yapma olasılığı yüksektir.   How Many IP Addresses a Computer Can Have? - GeeksforGeeks   * **Dinamik IPv4 Adres Atama (Dinamik IP** * IPv4 adresleri, **DHCP (Dynamic Host Configuration Protocol)** servisi tarafından, ağdaki istemcilere bir IP havuzundan otomatik ve belirli bir süreyle dağıtılan IP adresleridir. Böylece IP adresleri merkezi bir yerden kontrol edilir, her bir istemciye tek tek IP bilgisi girilmesi engellenir. * Ağdaki bir bilgisayarın ağdaki diğer bilgisayar ve cihazlarla iletişimde bulunabilmesi için IP adresinin, alt ağ maskesinin ve varsayılan ağ geçidi gibi bilgilerinin olması gerekir. * **DHCP;** IPv4 adresi, alt ağ maskesi, varsayılan ağ geçidi ve diğer yapılandırma bilgileri gibi adres bilgilerini otomatik olarak atar.   Issue with configuring IPV4. - Microsoft Community  **DHCP kullanmanın yararları şunlardır:**   * Ağ destek personeli üzerindeki yükü azaltır ve giril hatalarını neredeyse ortadan kaldırır. * Dinamik bir IP adresi cihazlara kalıcı olarak atanmaz, yalnızca belirli bir süre için kiralanır.   **DHCP Sunucuları**   * Broadcast yayın yaptıkları için DHCPv4 sunucuları genellikle tek bir LAN'a bağlanır. Bu yerel ağ dışında IP dağıtan DHCPv4 sunucularının router üzerinden geçiş hizmeti alması gerekir. * Bir ev ağında veya Soho'da kablosuz yönlendirici cihaz, ağdaki diğer aygıtlara DHCP hizmeti sunar, onlara private IP dağıtır; yani iç ağda bir DHCP server olarak çalışır. Aynı zamanda da ISS'den public IPv4 adresi almak üzere IP talebinde bulunan bir DHCP istemcisidir.   **DHCPv4 Yapılandırması İşlemi**   * İstemci broadcast yayını yaparak DHCPDISCOVER iletisi gönderir. * DHCP sunucusu, IPv4 adresleme bilgileri kümesine sahip bir DHCPOFFER iletisiyle bu isteğe cevap verir. * İstemci daha sonra DHCP sunucusu tarafından kendisine teklif edilen IPv4 adresleme bilgilerini kullanmayı istemek için bir DHCPREQUEST iletisi gönderir. * DHCP sunucusu, istemciden gelen bu isteği de bir DHCPACK iletisiyle cevaplar.      * Bir ev kablosuz yönlendiricisini yapılandırmak için tarayıcıyı açarak ve IP Adresi alanına yönlendiricinin varsayılan IPv4 adresini: 192.168.0.1 girerek grafiksel web arayüzüne erişelim. * 192.168.0.1 IPv4 adresi ve 255.255.255.0 alt ağ maskesi, iç yönlendirici arabiriminin varsayılanlarıdır. Bu, yerel ağdaki tüm cihazlar için varsayılan ağ geçididir. DHCP sunucusunun etkin olduğundan emin olun. * DHCP IPv4 adres aralığını başlangıç adresini (yönlendiriciye bu adres atandığından 192.168.0.1 kullanmayın) ve atanacak adres sayısını belirtin.   How To Manage Your Roadrunner Router's WiFi – CertSimple.com 54. IPv6 Adresleme  * IPv4 adreslerinin azalmaya başlamasıyla birlikte **IETF (Internet Engineering Task Force)** tarafından IPv6 geliştirilmeye başlanmıştır. * Bir IPv6 adresi 128 bit uzunluğundadır. * IPv4, broadcast'leri bol miktarda kullanır. Bu da birçok probleme sebep olur. Bunlardan en kötüsü, aşırı miktarda iletilen broadcast'in tüm ağı çalışamaz hale getirmesi ve tüm bant genişliğini tüketmesine sebep olan, güçlü broadcast fırtınasıdır. IPv6'de broadcast gibi bir şey yoktur çünkü o, multicast trafiği kullanmaktadır. * Ayrıca iki farklı haberleşme yöntemi daha vardır: Bunlar **unicast** ve **anycast**. * **Anycast** iletişim aynı adresin birden fazla makineye yerleştirilmesine izin verir. Böylece trafik, bu yolla adreslenmiş tek cihaza gönderildiğinde aynı adresi paylaşan en yakın host'a yönlendirilir. * İlk olarak, dört yerine sekiz adres grubuna sahip olduğuna ve bu grupların, nokta yerine iki nokta üst üste ile ayrıldıklarına dikkat edin. * Aynı MAC adresinde olduğu gibi, hexadecimal formatta gösterilmektedir. Bu nedenle, bu adresin sekiz adet 16-bit hexadecimal iki noktayla ayrılmış bloklarının olduğunu söyleyebiliriz.      * **GUA (Global Unicast Adsress),** yönlendirilebilir adreslerimizdir ve IPv4’dekilerle aynıdır.   DIŞ AĞDA! 2000-3FFF 2001-ODB8::/32 REZERVEDİR  **Link Local Address,** route edilemezler. Onları, route edilmeyecek, fakat hala lokal olarak dosya ve servisleri  paylaşması ve erişmesi gereken küçük bir LAN oluşturmak veya toplantılar için geçici bir LAN gönderme kabiliyeti sağlayan faydalı bir araç alarak düşünün. **Diğer bir deyişle,** DHCP üzerinden ani otomatik bir IP atayan cihazdan IP adresi alamadığımız zaman bilgisayara atanmış IP adresi olarak görürüz.  [Link Local Adress FE80-FEBF]   * **SLAAC (Stateless Address Autoconfiguration),** bir bilgisayarın DHCP sunucusuna ihtiyaç duymadan kendi varsayılan ağ geçidi olan yönlendirici cihazdan IPv6 adresi talep etmesine olanak sağlar.   Link-local adresi, aynı IPv6 yerel ağındaki cihazlarla iletişim kurmak için kullanılır. 54.1. IPv6 Adres Gösterimi  * 16 Bitlik herhangi bir bölümdeki baştaki sıfırları atlayın.   0DB8 = DB8 0000 = 0 0200 = 200  Herhangi bir sayıdaki "tüm sıfır" grubu atlanır ve onun yerine iki nokta üst üste (::) eklenir ancak bu bir kereye mahsus yapılabilir. Aşağıdaki örnekte ardışık 3 adet sıfır var bunları kaldırıp yerine :: koyduk. Peki biz sıkıştırılmış halinden kaç tane sıfır olduğunu nasıl anlayacağız? Eksikleri grupları sayarak .   54.2. IPv4’ten IPv6’ya Geçiş  * IPv6 alt versiyon (IPv4) ile uyumludur, ancak IPv4 üst versiyonla (IPv6) uyumlu değildir. * İki versiyonun birlikte çalışabilmesi için iki farklı yaklaşım vardır: * **Dual-stack (Çify Yığın)** * Aynı ağ segmentinde bulunan cihazların hem IPv4 hem IPv6 adres yapılarını bir arada kullanmalarına izin veirr. * Bu düğümler IPv6 veya IPv4 datagram’larını gönderip alabilir. * Giriş ve çıkış düğümündeki IP versiyonuna göre kullanacağı datagram türünü belirler. * Bir düğüm IPv6 protokolü çalıştırıyorsa, DNS’den IPv6 adresi döner, aksi takdirde IPv4 adresi döner. * IPv6’dan IPv4’e dönüşümde bazı alanlar kaybolur (flow).      * **Tunneling (Tünel Oluşturma)** * **Tunneling**, IPv6 paketini IPv4 ağı üzerinden taşıma yöntemidir. IPv6 paketi, diğer veri türlerine benzer şekilde bir IPv4 paketinin içinde kapsüllenir. * Tunneling, IPv6’da dual-stack ile oluşan alan kaybını önler. * IPv6 datagram’ı IPv4 datagram’ının payload’u yapılır. * Kaynak ve hedef adresler IPv4 tünelin ilk ve son düğümü alınır. * E düğümü IPv4 datagram’ı payload’undan IPv6’yı çıkarır.      * **NAT64** kurulumu en az 2 arayüzü olan bir ağ aygıtı (yönlendirici) gibi düşünülebilir. Bu arayüzlerden bir tanesi IPv4 ağına bağlı diğeri de IPv6 ağına bağlıdır.   Paketler, router cihaz tarafından IPv6 ağından IPv4 ağına yönlendirilecek şekilde yapılandırılmalıdır. Bu yönlendirici, IPv4 ağından IPv6 ağına ya da tam tersi dönüşümler için gerekli bütün işlemleri yapmalıdır.  Genel olarak, NAT64 ile iletişim IPv6 makinalar tarafından başlatıldığı zaman kullanılmak üzere tasarlanmıştır. Bu yüzden önce 6 rakamını sonra 4 rakamını yazarız. Bu işlemin tersini yapacak bazı mekanizmalar (sabit adres eşleme dahil) hala vardır.       * Hedefin Yerel Mi Yoksa Uzak Mı Olduğunu Hesaplama   **Ana bilgisayarlar IPv4 adresinin hangi bölümünün ağ olduğunu ve hangisinin ana bilgisayar olduğunu nasıl bilebilir?**  Cevap - Alt ağ maskesi  Alt ağ maskesi IPv4 adresiyle soldan sağa, bit bit karşılaştırılır.  Alt ağ maskesindeki 1'ler ağ bölümünü temsil eder; 0'lar ana bilgisayar bölümünü temsil eder.  Bir ana bilgisayar bir paket gönderdiğinde, alt ağ maskesini kendi IPv4 adresiyle hedef IPv4 adresiyle karşılaştırır.  Ağ bitleri eşleşirse, hem kaynak hem de hedef ana bilgisayar aynı ağdadır ve paket yerel olarak teslim edilebilir.  Ağ bitleri eşleşmezse, gönderen ana bilgisayar paketi diğer ağa gönderilmek üzere yerel yönlendirici arabirimine iletir.   54.3. Loopback IP Address  * 127.0.0.1 ile 127.255.255.255 aralığında bulunan IP adreslerdir. 127.0.0.1 adresi yerel hostu tanımlayan Loopback adrestir. Bu adres her bilgisayarın kendisini ifade eder. TCP/IP adresinin düzgün çalışıp çalışmadığını kontrol etmek amacıyla kullanılır. Mesela bu adres bilgisayarınızda bir web servisi çalıştırıyorsanız ve çalışırlığından emin olmak istiyorsanız browser'a http://127.0.0.1 yazarak onu kontrol edebilmenizi sağlar.  55. Yazılım Tanımlı Ağ (SDN - Software Defined Networking) Nedir?  * **SDN,** ağ kontrol düzleminin sevk düzleminden fiziksel olarak ayrılması ve bir kontrol düzleminin çeşitli aygıtları denetlemesini sağlar. SDN, ağı sanallaştıran bir ağ mimarisidir. * **SDN,** ağ kontrol düzleminin ağ aygıtından farklı bir öğe olarak incelenerek, her ağ aygıtının uyguladığı hesaplamanın, merkezi bir denetleyici ya da denetleyiciler topluluğuna atanabileceği temeline dayanan, sadece bir noktadan programlama yapılarak ağ aygıtlarının davranışlarının tespit edilmesini sağlayan bir yapıdır. Yapısında OpenFlow iletişim protokolünü bulundurmaktadır. * SDN’de ağ yöneticisi, bütün ağ aygıtlarını ayrı ayrı oluşturmaya ihtiyaç duymadan, merkezi bir arayüzde gerçekleştirebilir, ağ trafiğini güncelleyebilirler. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ****56. Alt Ağ Maskesi (Subnet Mask)****  * **Subnet mask,** bir IP adresinin hangi bölümünün Network hangi bölümünün host yani kullanıcı olduğunu belirlemek için kullanılır. IP ile beraber, o ağda toplamda kaç bilgisayar olabileceğini ve hangi IP aralığında olursa cihazların aynı ağda sayılabileceklerini belirler.   **Diğer bir deyişle;** IP adresleri her evin adresi ise subnet mask’de bu evlerin hangi mahallede olduklarını belirten yapılardır. Her subnet mask sadece kendi subnet mask’indekilerle konuşabilir.   * Bir bilgisayarın başka bir bilgisayarla iletişime geçebilmesi için öncelikle aynı networkt’te olması gerekiyor. Bu iki bilgisayar sadece IP adresini kullanarak aynı networkt’te olduklarını anlayamazlar aynı zamanda **aynı subnet mask’te** olmaları da gerekir. Aşağıda A sınıfı, B sınıfı ve C sınıfı adreslerin ilk kaç bitinin network biti olduğu yer almaktadır.     Eğereki IP adresleri farklıysa router’lar veyahut gateway’ler aracılığıyla haberleşebililrer. Doğrudan kendi aralarında haberleşemezler.     * Alt ağ maskesi, alt ağın boyutunu ve kaç tane IP'ye sahip olacağını belirler. Alt ağ maskesi kullanıcı sayısına ve istenen IP sayısına göre seçilmelidir. * Aynı alt ağda bulunan bilgisayarları temsil eden sınıf adresine **ağ adresi (Network ID)** denir. Bu adresler, IP olarak herhangi bir cihaza atanamazlar ve oluşturulan alt ağların ilk adresleridir.   **Host adresi (Host ID),** server, bilgisayar, yazıcı vb. gibi cihazların kendi ağları içerisinde ayırt edilebilmesi için kullandıkları adresdir.   * Herhangi bir ağda bütün adresleri temsil etmek için kullanılan adreslere **Broadcast Adres (Yayın Adresi)** denir. Bu adresler de ağ adresi gibi ağdaki herhangi bir bilgisayara IP adresi olarak atanamazlar ve oluşturulan alt ağların son adresleridir.   **Alt Ağ Maskelerinin  ikilik taban (binary)  ve ondalık taban (decimal)** gösterimi:  Örneğin; alt ağ maskesi 255.0.0.0 olan maskenin gösterimleri aşağıda verilmiştir.   |  |  | | --- | --- | | Maske Gösterim Şekilleri | Alt Ağ Maskesi | | İkilik Taban | |  |  |  |  | | --- | --- | --- | --- | | 11111111 | 00000000 | 0000000 | 0000000 | | | Ondalık Taban | |  |  |  |  | | --- | --- | --- | --- | | 2555 | 0 | 0 | 0 | |  * Bir alt ağ maskesindeki 1 bitleri IP adresinin [**ağ adresi kısmını**](#networkIDnedir), 0 bitleri ise IP adresinin [**host** kısmını](#hostIDnedir) gösterir. * Host sayısını bulmada genel formül ’dir. Buradaki n, alt ağ maskesindeki 0 olan bitlerin sayısıdır. * Alt ağ maskesindeki 0 olan bitlerin, IP adresindeki karşılığı olan bitlerin tamamı 1 yapılarak **broadcast adresi** hesaplanır. * Alt ağ maskesi ile IP adresi mantıksal AND (VE) işlemine tabi tutulduğunda **ağ adresi** bulunur. * Her bir alt ağdaki istemci sayısı **-2** formülünden bulunabilir. Kullanıcı sayısını veren değerinden 2 çıkarılması ise ağ adresi (Network ID) ve yayın adresinin (Broadcast ID) IP olarak kullanıcılara verilememesinden kaynaklanmaktadır.   <https://www.udemy.com/course/ip-adresi-ve-alt-aglara-bolme-ip-addressing-and-subnetting-2/learn/lecture/20922390#overview>  **Örneğin;** IP adresi 194.134.60.2 ve 194.134.60.110 , alt ağ geçidi ise 255.255.255.0 olan IP'lerin  ağ adresini hesaplarsak;   |  |  |  | | --- | --- | --- | | IP Adresi | Alt Ağ Geçidi | Ağ Adresi | | 194.134.60.2 | 255.255.255.0 | 194.134.60.0 | | 194.134.60.110 | 255.255.255.0 | 194.134.60.0 |   **Örneğin;** IP adresi 132.44.237.15 ve Alt Ağ Maskesi 255.255.248.0 olan alt ağın Network ve Broadcast adresini hesaplrsk;   |  |  | | --- | --- | | IP Adresi | İkilik Tabanda Gösterim | | 132.44.237.15 | |  |  |  |  | | --- | --- | --- | --- | | 1000100 | 00101100 | 11101**101** | **00001111** | | | 255.255.248.0 | |  |  |  |  | | --- | --- | --- | --- | | 11111111 | 11111111 | 11111**000** | **00000000** | |   Alt Ağ Maskesinin 0 bitlerine karşılık gelen IP adresleri koyu renkle gösterilmiştir. Bu bitlerin hepsi 0 yapılırsa Ağ Adresi, 1 yapılırsa Broadcast Adresi bulunur. Bu durumda;   |  |  |  | | --- | --- | --- | | Adres Tipi | İkilik Tabanda Gösterim | Ondalık Tabanda Gösterim | | Ağ Adresi | |  |  |  |  | | --- | --- | --- | --- | | 1000100 | 00101100 | 11101**000** | **00000000** | | 132.44.232.0 | | Broadcast Adresi | |  |  |  |  | | --- | --- | --- | --- | | 1000100 | 00101100 | 11101**111** | **11111111** | | 132.44.239.255 |   AND (VE) işlemi ile bulmak istenirse;   |  |  |  | | --- | --- | --- | | IP Adresi | Alt Ağ Geçidi | Ağ Adresi | | 132.44.237.15 | 255.255.248.0 | 132.44.232.0 |     **Örneğin;** 212.45.64.0 IP adresini 32 IP’lik alt ağlara bölmek için önce alt ağ maskesi bulunmalıdır. 32, olduğu için son 5 bit host adres için ayrılmalıdır. Geriye kalan 27 bit ise network kısmını oluşturur.   |  |  | | --- | --- | | Gösterim Şekilleri | Alt Ağ Geçidi | | İkilik Taban | |  |  |  |  | | --- | --- | --- | --- | | 11111111 | 11111111 | 11111111 | 111**00000** | | | Ondalık Taban | |  |  |  |  | | --- | --- | --- | --- | | 255 | 255 | 255 | 224 | |  * Her alt ağ için ayrılan 32 IP’lik kısmın sırasıyla network adresleri (network ID): 212.45.64.0/27   212.45.64.32/27 212.45.64.64/27  212.45.64.96/27 212.45.64.128/27  212.45.64.160/27 212.45.64.192/27  212.45.64.224/27  Örnek: Müdürünüz size 10.100.100.0/24 IP’sini aşağıdaki departmanlara uygun olarak dağıtmanızı söyledi. İlk departmanda 50 host, ikincisinde 20 host, üçüncüsünde ise 10 host bulunması gerektiğini bildirmiş. Broadcast IP’lerini de bularak çözümünü yapalım.    Bir alt ağa 50 host verebilmek için 2n-2 ≥ 50 olacak şekilde olmalıdır. Burada n=6 olması gerekmektedir. İlk networkümüz 64’lük bir parçadan oluşacaktır. n burada host bitimiz olduğu için ve IP C class olduğundan 8 tane 0'dan 6 tane 0 çıkarırsak network bitimiz yani 1’lerin sayısı 2 olacaktır. İlk networkümüzün subneti de /26 olacak.  **network 1 = 10.100.100.0/26**  20 hostluk bir network için 2n-2 ≥ 20 burada n sayımız 5 olacaktır. İkinci networkümüz ise 32’lik bir parçadan oluşacaktır. Subnet ise 8-5'ten 3 tane network bitimizden oluşacak. Subnetimiz de /27 olacak. İlk network’e 10.100.100.0/26 yani 64’lük bir parça verdiğimizden ikinci networkümüz 64’den başlayacak.  **network2 = 10.100.100.64/27**  10 hostluk bir network için 2n-2 ≥ 10 burada n sayımız 4 olacaktır. 16’lık bir parça gerekmektedir. Subnet 8-4’ten 4 tane network bitimizden oluşacak. Subnet /28 . ikinci network 10.100.100.64/27 yani 32’lik bir parça verildiğinden üçüncü network:  **network3 = 10.100.100.96/28**  Örnek: Sağ tarafta bulunan networkleri uygun yerlere yerleştirelim.    RouterA’ya bağlı 7 host bulunacak. Burada küçük bir detay göze çarpıyor genelde, CCNA sınavlarında da bu şekilde şaşırtmaca sorabilirler. 2n-2 bizim host sayımızdı bu kuralı asla unutmayacağız yani 7 host görüp, 8’lik alt ağa bölersek yanlış yapmış oluruz. Çünkü tekrar hatırlatalım hatırlatalım "network IP’si ve broadcast İP’si kullanılamaz".  Bu durumda routerA’ya bağlı alt ağımız 16‘lık parça olmak zorunda. n = 4 olduğundan en az 4 tane "0" yani host bitimiz olacak bu durumda hali hazırda /24 olan network için /28’lik bir para ayıracağız.  RouterB’ye bağlı 90 host için 128’lik bir parça ayırmamız gerekiyor. 27-2 = 120 n=7 olduğundan 7 tane "0" yani host bitimiz olacak. /24 networkümüz için ayıracağımız parça /25 olacak bu durumda.  RouterC’ye bağlı 23 host için en küçük 32’lik bir parça gerekiyor. 27-2 = 30 n=5 olduğundan 5 tane "0" yani host bitimiz olacak. /24 networkümüz için ayıracağımız parça /27 olacak bu durumda. 56.1. Subnetting (Alt Ağlara Bölme)  * Network yapısını alt ağlara bölme işlemine **alt ağlara bölme (subnetting)** denir. IP yönetimini daha verimli yapmak için kullanılır. Bu işlem ile IP uzayları alt sınıflara ayrılır.   Subnetting Neden Yapılır ?   * Mevcut IP adresleri sınırlı olması nedeniyle ve 32 bitlik IP adres alanının verimliliğini arttırmak için yapılır. Bu işlem sayesinde boşa harcanmış IP adresi sayısı azaltılmış olur. * Alt ağlara bölme işlemi broadcast (yayın) etkisini yani tüm kullanıcılara paket gönderilmesini azaltır. Bu da ağ performansını arttırır. * Alt ağlara bölünmüş bir network daha kolay yönetilebilir. * Güvenlik gerekçeleri ile de alt ağlara bölme işlemi yapılabilir. Bu alt ağlara erişim kısıtlanabilir.   Classfull IP Adresi Sınıfları  IP adreslerinin sınıflarına göre alt ağ maskeleri belirlenmiştir. Sınıfına uygun alt ağ maskesine sahip olan IP adreslerine **classfull IP adresi** denir.  **A sınıfı:** İlk okteti    **1**.0.0.0 – 126.255.255.255 *arasında* olanlar ---> Varsayılan Ağ Maskesi: 255.0.0.0  **Binary olarak** 0000000**1** - 01111111 **B sınıfı:** İlk okteti **128**.0.0.0 – 191.255.255.255 *arasında* olanlar --> Varsayılan Ağ Maskesi: 255.255.0.0  **Binary olarak 1**0000000 - 10111111 **C Sınıfı:** İlk okteti **192**.0.0.0 – 223.255.255.255 *arasında* olanlar --> Varsayılan Ağ Maskesi: 255.255.255.0  **Binary olarak 11**000000 - 11011111 **D sınıfı:** İlk okteti **224**.0.0.0 – 239.255.255.255 *arasında* olanlar --> Varsayılan Ağ Maskesi: 255.255.255.255 **Binary olarak 111**00000 - 11101111  **E sınıfı:**  İlk okteti **240**.0.0.0 – 254.255.255.255 *arasında* olanlar ---> Araştırma için ayrılmıştır.  **Binary olarak 1111**0000 – 11111111  **Özel (Private) IP Adresleri**  10.0.0.0 - 10.255.255.255 1A Sınıfı Network 172.16.0.0 - 172.31.255.255 16B Sınıfı Network 192.168.0.0 - 192.168.255.255 256C Sınıfı Network Bu IP adresleri yerel alan ağlarında kullanılmak üzere tahsis edilmiştir ve internete çıkacakları zaman gerçek IP adreslerine NAT yapılarak kullanılabilirler. 56.1.1. IP Subnetting (Binary (İkili) Sayı Sistemini Kullanarak Bulma) **Örnek-1:**  [**https://www.udemy.com/course/ip-adresi-ve-alt-aglara-bolme-ip-addressing-and-subnetting-2/learn/lecture/20917050#overview**](https://www.udemy.com/course/ip-adresi-ve-alt-aglara-bolme-ip-addressing-and-subnetting-2/learn/lecture/20917050#overview)     * CIDR, subnet mask’teki 1’lerin toplamını verir. * Yukarıdaki 1 bitlerin sayısı yani prefix’lerin sayısı IP adresinin **network kısmını** gösteriyor. Yani IP adresini binary olarak yazdığımızda soldan başlayarak subnet mask’daki 1 sayısı kadar olan kısım bize network kısmını gösteriyor. Subnet mask’daki 0’ların olduğu kısım ise IP adresinin **host kısmını** gösteriyor.         **Örnek-2:**  [**https://www.udemy.com/course/ip-adresi-ve-alt-aglara-bolme-ip-addressing-and-subnetting-2/learn/lecture/20917070#overview**](https://www.udemy.com/course/ip-adresi-ve-alt-aglara-bolme-ip-addressing-and-subnetting-2/learn/lecture/20917070#overview)       57. Gateway  * Önceki konularda anlattığımız bir client kendi networkünde olan bir başka client ile haberleşeceği zaman [ARP sorgusu](#ARPsorgusu) yapıp MAC adresini öğrendikten sonra haberleşiyordu. Farklı bir networkle haberleşeceği zaman ise istekler gateway denen IP’ye doğru yönlenir. Farklı networkten kastımız ise kendi networkü dışındaki bütün networkler. Bu gateway’e örnek evinizdeki modemleri verebilirsiniz. İnternet sizin bilgisayarınızın bilmediği bir network ve internete yaptığınız her istek modem’e çıkacaktır. Genelde gateway'lere networklerin kullanılacak ilk IP adresini verirler, ama kullanılır IP aralığından istediğinizi de kullanabilirsiniz. * Örneğimizde bizden gateway IP adreslerini istemiş, Router A için /28 olacağını söylemiştik. Yani gateway 192.168.55.29/28 olacak. Router B için /25 olacağını bulmuştuk. Yani gateway 192.168.55.132/25 verebiliriz. Router C için /27 olacağını bulmuştuk 192.168.55.57/27 verebiliriz.  Ve son olarak RouterA'nın RouterB ile bağlı olan networkünü inceleyelim. RouterA tarafına 192.168.55.2/30 verilmiş. 30 maskımız demek IP adresimiz toplam 32 bit olduğundan geriye 2 tane host bitimiz kaldığını belirtiyor. 22-2 = 2 olmaktadır. Yani iki cihaz birbirine bağlanacağı zaman verebileceğimiz en küçük network /30 olmaktadır. 4’lük bir parça olsa bile 1 network 1 broadcast IP adresi kullanmayacağımız için geriye 2 tane kalıyor. Sağ üst köşede 2 tane verilmiş biri 192.168.55.0/30 ve 192.168.55.1/30 kuralımızı bildiğimizden “network IP si kullanılamaz” bu durumda 192.168.55.0/30 un kullanılamayacağını biliyoruz. RouterB’nin bacak IP'si bu durumda 192.168.55.1/30 olacaktır. |

|  |  |  |
| --- | --- | --- |
| 58. ICMP (Internet Control Message Protocol – İnternet Kontrol Mesaj Protokolü)  * ICMP, İnternet Katmanında IP paketinin veri bölümünde çalışıp, sorunları haberleşen birimlere bildirerek bir geri besleme mekanizması oluşturur. * ICMP genel olarak; TTL süresi dolduğu zaman paketin sahibine bildirim yapma, herhangi bir durumda yok edilen paket hakkında geribildirim sağlama, hata oluşumlarında geribildirim sağlama, paket başka bir yoldan gideceği zaman geribildirim sağlama gibi görevler üstlenir.  58.1. ICMP Paket Yapısı  * ICMP mesajları IP datagramının veri alanında taşınır. IP başlığındaki protokol alanı 1'e ayarlanarak ICMP'nin kullanıldığı gösterilir.  |  |  | | --- | --- | |  | Tip Alanı: Mesaj tipini tanımlar.  Kod Alanı: Hata veya durum bilgisi tipini tanımlar.  Toplamsal Hata(Checksum): ICMP mesajının 16 bit'lik 1'e tümleyenini hesaplar.  Parametreler: Parametrelerin daha uzun halinin belirlenmesinde kullanılır.  Bilgi: Mesajla ilgili bilgidir. |  58.2. Mesaj Tipinin Alabileceği İfadeler  * ICMP paketinde durum bilgisi tip bölümünün aldığı parametreler ile tanımlanır.   0  -  Yankı(Echo) yanıtı (Echo Reply)-ping yanıtı  3  -  Hedefe Erişilemedi (Destnation Unreachable)  4  -  Kaynak Kapatmak (Source Quench)  5  -  Yeniden Yönlendirme (Redirection Required)  8  -  Yankı isteği (Echo Request)-(ping isteği)  9  -  Yönlendirici (Router)  tanıtımı  10 – Yönlendirici (Router) istemi  11 - Zaman aşımı (Time to Live Exceeded)-traceroute tarafından kullanılır.  12 - Parametre Problemi (Parameter Problem)  13 - Tarih Bilgisi İstemi (Timestamp Request)  14 - Tarih Bilgisi Yanıtı (Timestamp Reply)  15 - Bilgi İstemi (Information Request)  16 - Bilgi Yanıtı (Information Reply)  17 - Addres Maskesi istemi (Address Mask Request)  18 - Addres Maskesi yanıtı (Address Mask Reply) |

|  |
| --- |
| 59. TELNET  * **TELNET**, internet ağı üzerindeki çok kullanıcılı bir makineye uzaktaki başka bir makineden bağlanmak için geliştirilen bir TCP/IP protokolü ve bu işi yapan programlara verilen genel isimdir. * TELNET’i, komut istemi (cmd) kullanarak çalıştırabilirsiniz. Eğer aktif değilse, TELNET’i aktif hale getirmeniz gerekir. **Varsayılan olaral aktif değildir.**   Uygulamanız bittikten sonra tekrar pasif konuma getirmeniz tavsiye edilir.  metin içeren bir resim  Açıklama otomatik olarak oluşturuldu   * Telnet sunucuları, TCP bağlantı noktası 23'te istemci isteklerini dinler. * Telnet kullanan bir bağlantıya sanal terminal (VTY) oturumu denir. * İstemci, komutları sunucuya fiziksel olarak bağlıymış gibi yürütebilir.  59.1. Telnet'teki Güvenlik Sorunları  * Telnet bağlantısı kurulduktan sonra kullanıcılar, sunucunun kendisinde bir komut satırı oturumu kullanıyormuşçasına, sunucudaki herhangi bir yetkili işlevi gerçekleştirebilirler. Bu da her şeye erişebileceği anlamına gelir. * Telnet protokolü bir kullanıcının oturum açmasını gerektirse de şifrelenmiş verilerin taşınmasını desteklemez. * Telnet oturumları sırasında değiş tokuş edilen tüm veriler ağ üzerinden düz metin olarak taşınır. Yani her şey apaçık ortadadır. * Buna çözüm olarak, **güvenli kabuk (SSH) protokolü,** sunucu erişimi için alternatif ve güvenli bir yöntem sunar. SSH, güvenli uzaktan oturum açma ve diğer güvenli ağ hizmetleri için yapı sağlar.      * Güvenli Kabuk (SSH) protokolü, sunucu erişimi için alternatif ve güvenli bir yöntem sunar. SSH, güvenli uzaktan oturum açma ve diğer güvenli ağ hizmetleri için yapı sağlar. * SSH, Telnet'ten daha güçlü kimlik doğrulaması sağlar ve şifreleme kullanarak oturum verilerinin taşınmasını destekler. |

|  |
| --- |
| 60. Kablosuz Ağ Ayarları  * **Network mode** - Desteklenmesi gereken teknoloji türünü belirler.   Örneğin, 802.11b, 802. 11g, 802.11n veya Mixed Mod   * **Network Name (SSID)** - WLAN'ı tanımlamak için kullanılır. WLAN’a katılmak isteyen tüm cihazların aynı SSID'ye sahip olması gerekir. Yani özetle modemin bize gözüken Wi-Fi’daki adıdır SSID. * **Standard Channel** - iletişimin gerçekleşeceği kanalı belirtir. Varsayılan olarak, bu, AP'nin kullanılacak en uygun kanalı belirlemesine izin vermek için Auto olarak ayarlanır. * **SSID Broadcast** – SSlD’nin kapsam içindeki tüm cihazlara yayınlanıp yayınlanmayacağını belirler. Varsayılan olarak, Enabled durumdadır.  61. ISS  * **ISS,** Internet Servis Sağlayıcısıı, ev ağı ile internet arasındaki bağlantıyı sağlar.   Bir ISS, müşterilerine e-posta hesapları, ağ depolama, web sitesi barındırma ve otomatik yedekleme veya güvenlik hizmetleri gibi ek hizmetler sunabilir.  Her ISS, tüm dünyadaki kullanıcıları birbirine bağlayan bir bağlantı ağı oluşturmak için diğer ISS'lere bağlanır.  **Ev kullanıcıları için diğer ISS bağlantı seçenekleri şunlardır:**  Cellular (Hücresel) - Hücresel internet erişimi bağlanmak için bir cep telefonu şebekesi kullanır. Performans, telefonun ve bağlı olduğu hücre kulesinin yetenekleriyle sınırlı olacaktır.  Satellite - Uydu servisi, DSL veya kabloya erişimi olmayan evler veya ofisler için iyi bir seçenektir. Uydu antenleri uyduya net bir görüş hattı gerektirir ve bu nedenle yoğun ormanlık alanlarda veya diğer hava şartları zor olan yerlerde problemli olabilir.  Dial-up Telephone - Herhangi bir telefon hattını ve modemi kullanan ucuz bir seçenek. ISS'ye bağlanmak için bir kullanıcı ISS erişim telefon numarasını arar. Çevirmeli modem bağlantısı tarafından sağlanan düşük bant genişliği, genellikle büyük veri aktarımı için uygun değildir. 62. Bulut Türleri (Veri Depolama Türleri)  1. Public clouds - Genel bulutta sunulan uygulamalar ve hizmetler genel popülasyona açıktır. Hizmetler ücretsiz olabilir veya kullanım başına ödeme modelinde sunulur. 2. Private clouds - Özel bir bulutta sunulan uygulamalar ve hizmetler, hükümet gibi belirli bir kuruluş veya kuruluşlara yöneliktir. 3. Hybrid clouds - Karma bulut, her bir parçanın ayrı bir nesne olarak kaldığı, ancak her ikisinin de tek bir mimari kullanılarak birbirine bağlandığı iki veya daha fazla buluttan (örnek: kısmen özel, kısmen genel) oluşur. 4. Community clouds - Belirli bir topluluk tarafından özel kullanım için bir topluluk bulutu oluşturulur. İşlevsel ihtiyaçlar toplum için özelleştirilir. Örneğin, sağlık kuruluşları.   **Bulut Servisleri**   * National Institute of Standards and Technology (NIST) tarafından yayınlanan 3 ana bulut bilişim hizmeti vardır:  1. Software as a Service (SaaS) - Bulut sağlayıcısı, internet üzerinden gönderilen e-posta, iletişim ve Microsoft 365 gibi uygulamalara ve hizmetlere erişimden sorumludur. 2. Platform as a Service (PaaS) - Bulut sağlayıcısı, kullanıcılara uygulamaları sunmak için kullanılan geliştirme araçlarına ve hizmetlerine erişim sağlamaktan sorumludur. 3. Infrastructure as a Service (laaS) - Bulut sağlayıcısı, BT yöneticilerine ağ donanımına, sanallaştırılmış ağ hizmetlerine erişim sağlama ve ağ altyapısını desteklemekten sorumludur.  63. Ağ Sanallaştırma  * **Bulut bilişim,** verilere erişmenin ve depolamanın yollarından biridir.   **Sanallaştırma,** bilgi işlem kaynaklarının sanallaştırılmış sürümünü oluşturmak için teknolojiyi kullanıyor.  Sanallaştırma teknolojisi, bir ana işletim sisteminin bir veya daha fazla istemci işletim sistemini desteklemesini sağlar.   * **Hipervizör,** fiziksel donanımın üstüne bir soyutlama katmam ekleyen bir program, bellenim veya donanımdır.   Soyutlama katmanı, CPU'lar, bellek, disk denetleyicileri ve NIC'Ier gibi fiziksel makinenin tüm donanımlarına erişimi olan sanal makineler oluşturmak için kullanılır.     * Sanallaştırma, işletim sistemini (OS) donanımdan ayırır.   Sunucu sanallaştırması boşta kalan kaynaklardan yararlanır ve gerekli sunucu sayısını birleştirir. Tek bir donanım platformunda birden fazla işletim sisteminin bulunmasına izin verir.  Ağ sanallaştırması, geleneksel ağ donanımı ve yazılım ağ kaynaklarını yazılım tabanlı bir varlık olan sanal bir ağda birleştirir.     * Bir ağ aygıtı iki düzlem içerir:  1. Control plane (Kontrol Düzlemi)   Veri iletme kararları vermek için kullanılır. Denetim düzleminde IPv4 ve IPv6 yönlendirme tabloları ve ARP tablosu gibi Katman 2 ve Katman 3 rota iletme mekanizmaları bulunur. Kontrol düzlemine gönderilen bilgiler CPU tarafından işlenir.   1. Data plane (Veri Düzlemi)   Yönlendirme düzlemi olarak da adlandırılan bu düzlem, tipik olarak bir aygıttaki çeşitli ağ bağlantı noktalarını bağlayan anahtar dokusudur. Her cihazın veri düzlemi trafik akışlarını iletmek için kullanılır. Yönlendiriciler ve anahtarlar, gelen trafiği uygun çıkış (giden) arabiriminden iletmek için kontrol düzlemindeki bilgileri kullanır- Veri düzlemindeki bilgiler genellikle CPU'nun devreye girmeden özel bir veri düzlemi işlemcisi tarafından işlenin |

|  |
| --- |
| 64. Komut Satırı Network Kontrol İşlemleri  * **ping,** küçük bir veri kümesinin cihazınızdan internetteki bir sunucuya ve tekrar cihazınıza aktarılması için geçen süreye denir.   **‌Ping Nasıl Çalışır?**  ‌Ping komutu iki ‌IP adresi arasında çalıştırılır. Hedef ‌IP adresine ‌ping komutu verildiğinde karşı tarafa ‌ECHO\_REQUEST istek paketi gönderilir. İstek kendisine ulaşan hedef ‌IP adresi, kaynak noktasına ‌ECHO\_REPLY cevap paketini yollar. İstek mesajıyla cevap mesajı arasındaki süre gecikme süresi olarak bilinir. Gecikme süresi ne kadar uzunsa bilgisayar ağının hızı o kadar yavaş anlamına gelir.  TTL(time to live) değeri IP packetinde bulunan bir değerdir. Hedefe ulaşana kadar geçtiğiniz networklerde 1 düşer. 0 olunca hangi network cihazına geldiyse o cihaz bu paketi drop eder     * **tracert** sayesinde, bilgisayarınızdan bir başka hedef noktaya ulaşırken, paketlerimizin hangi yolları takip ettiği kontrol edilir. Bu komut ile hedef bilgisayar ve sizin aranızdaki en yakın ve en kısa yolu belirleyen routerların bir listesi gösterilmiş olur.      * **arp -a,** IP adresini bildiğimiz bilgisayarın MAC adresini öğrendikten sonra onunla haberleşebiliriz. Peki, o bilgisayara tekrar bilgi göndermek istesek ne olacak? Tekrar bir ARP broadcast mesajı mı yayınlanacak? Hayır, çünkü ARP üzerinden elde edilen bilgiler arp tablosunda saklanır ve bir MAC adresi gerekli olduğu zaman, ilk önce buraya bakılır. Eğer IP adresine karşılık gelen MAC adresi bulunuyorsa, broadcast yapmadan bu adres kullanılarak iletişime geçilebilir.      * **ipconfig /flushdns,** İşletim sistemleri ve DNS istemcileri işlemleri hızlandırmak için IP adresleri ve diğer DNS sonuçları gibi sonra tekrar kullanılabilecek bilgileri önbelleğe alır. Alınan bazı bilgiler probleme sebep olabilir ve güncellemeleri almanızı engeller. Bu da doğru IP’yi almanızı engeller. Bu komut, alan adınızla tekrar doğru bir şekilde iletişim kurabilmeniz için önbelleğin temizlenmesi sağlar.     **ipconfig /all**: DHCP’den aldığımız IP, GATEWAY, DNS IP’lerini görebiliriz. ipconfig /release: DHCP’den aldığımız IP’yi serbest bırakırız. **ipconfig /renew**: DHCP’den yeni bir IP alır.   * **netstat -an,** komutu ile sunucudan dışarıya eriştiğimizde kullanılan source portlarını da görmüş oluruz. Mesela aşağıda 95.168.168.213 IP’sinin 80 portuna yani web sitesine erişim sağlanmış. Dikkat edersek 80 portuna giderken aynı zamanda 59 binli random porttan erişim sağlanıyor.      * **nslookup** komutu, bir servisin TCP/IP adresinin bulunmasına yardımcı olur. Domain adları ve ip adresleri ile ilgili olarak sorgulama desteği sunar.   **NsLookup Nasıl Çalışır ?**  Bunun için Win + R tuşları ile birlikte basılır ve çalıştır ekranı açılır. Çalıştır ekranına cmd yazılarak komut istemcisi açılır.  Komut istemcisine **nslookup** yazarak default server IP adresimizi görüntüleyebiliriz.    **nslookup alanadınız**: web servisinizin IP adresini sorgular.    **nslookup -query=mx alanadınız**: mail servisinizin IP adresini sorgular.    **nslookup -type=ns alanadınız**: Nameserver bilginizi sorgular.    **nslookup -type=any alanadınız**: web, mail, nameserver bilgilerinizin ayrıca txt kayıtlarınızın tümünü görüntüleyebildiğiniz sorgu çeşididir. |

|  |
| --- |
| 65. WIRESHARK ip**.**dst\_host contains **"atauni"** *// atauni iceren trafik izlenir.*  ip**.**addr**==**10.106.3.2 *// IP adresi 10.106.3.2 olan trafik izlenir.*  *// (Belirtilen IP Source veya Destination olabilir.)*  ip**.**src**==**10.106.3.2 *// Source IP adresi 10.106.3.2 olan trafik izlenir.*  ip**.**dst**=**10.106.3.2 *// Destination IP adresi 10.106.3.2 olan trafik izlenir.*  https *// https trafigi izlenir*  smtp *// smtp trafigi izlenir.*  ftp *// ftp trafigi izlenir.*  tcp**.**port**==**443 *// TCP portu 443 olan trafik izlenir.*  tcp**.**port eq 443 *// eq-equals anlamina gelir.*  tcp**.**srcport**==**443 *// TCP Source portu 443 olan trafik izlenir.*  tcp**.**dstport**==**443 *// TCP Destination portu 443 olan trafik izlenir.*  udp**.**port**==**53 *// UDP portu 53 olan trafik izlenir.*  udp port eq 53  ip**.**src**==**192.168.1.0**/**24  *// 255.255.255.0 subnet’i için(256 IP) trafik izlenir.*  eth**.**addr**[**0**:**6**]==**FF**:**FF**:**FF**:**FF**:**FF**:**FF  *// Ethernet karti MAC adresiyle trafigi izlenir.*  frame**.**len **>** 20 *// Frame boyutu 20'den buyuk trafik izlenir.*  frame**.**len **<** 30 *// Frame boyutu 30'dan kucuk trafik izlenir.*  frame**.**len gt 20 *// gt-greater than anlamina gelir.*  frame**.**len lt 30 *// lt-less than anlamina gelir.*  frame**.**len **>=**0x100  frame**.**len **<=**0x20  ip**.**src**==**10.0.0.5 **and** tcp**.**flags**.**fin  *// Source IP adresi 10.0.0.5 olan ve TCP bayragi fin olan trafik izlenir.*  ip**.**scr**==**10.0.0.5 **or** ip**.**src**==**192.1.1.1  *// Source IP adresi 10.0.0.5 ya da 192.1.1.1 olan trafik izlenir.*  **not** http *// http olmayan trafik izlenir*  **!** http *// http olmayan trafik izlenir*   * Herhangi bir paket seçilerek o pakete ait Transmission Control Protocol alanıyla ilgili bilgiler gösterilir.      * Ayrıca bu alanda Internet Protocol ile ilgili bilgilerde gösterilir.      * Wireshark paketleri Paket Listesi, Paket Detayları ve Paket Byte alanları şeklinde listeler.      * Herhangi bir paketi takip etmek için [Follow] seçeneği takip edilmelidir. Takip edilen paket ile ilgili detaylar farklı bir pencerede sunulurr. |

|  |
| --- |
| 66. Fiziksel Bağlantı Çeşitleri  * Ağa bağlanan cihazlar Ethernet teknolojisini kullandığını belirtmiştik bu bağlantıları da belli bir kablo düzeneği ile yaparlar. Bu kablolara RJ-45 denir.   Normalde kablo sıramız straight düzendir. Turuncu-beyaz, turuncu, yeşil-beyaz, mavi, mavi-beyaz, yeşil, kahverengi-beyaz, kahverengi. Dünyada bütün son bağlantılar bu sıralamayla yapılmaktadır.     * Bu kabloların renk sıralamasına göre bağlantı çeşitliliği yapılır. Günümüzde bu şekilde bir ihtiyaç bulunmamaktadır ama eskiden iki bilgisayarı birbirine direk rj-45’leri takarak konuşturamıyordunuz. Kabloyu cross yaparak konuşturabiliyorduk. Aynı durum network cihazlar içinde geçerli. Aşağıda belirli network cihazları için bağlantı çeşitleri verilmiştir.      * Rollover kablo network cihazlarına bağlanmak için kullanılan console kablosunda kullanılır.   Basit bir yöntem olarak aynı mantıkta çalışan cihazlar birbirine cross (router-PC hariç), farklı mantıkta çalışan cihazlar birbirine straight bağlanır. |

|  |
| --- |
| 67. TERİMLER SÖZLÜĞÜ  * FDM (Frequency Division Multiplexing) (Frekans Bölmeli Çoğullama) Nedir ?   **FDM,** hücresel ve analog sistemlerde çoğullama tekniklerinden biridir. Bu teknik, radyo vericilerinin aynı frekans spektrumunda veri göndermesi için kullanılır.  Bir iletişim kanalı var 4 tane kullanıcı bu iletişim kanalını kullanabiliyor (daha dar bant genişliklerine bölünür.)   * TDM (Time Division Multiplexing), zaman olarak bölünmeler gerçekleşir. * ISP (Internet Service Provider) Nedir ?   **İnternet servis sağlayıcıları** internete erişmenizi sağlayan bir aracıdır.  Türkiye’de bulunan ISP firmaları arasında BİMcel, TTNET, Turk.net, Turkcell, Vodafone bulunur.   * IETF (Internet Engineering Task Force) Nedir ?   İnternet uygulamasında standart çalışmaları ve önerileri yapan, kırktan fazla çalışma grubundan oluşan **uluslararası bir birliktir**.   * RFC (Request For Comment) Nedir ?   İnternette kullanılan protokoller ile ilgili **standartları tanımlayan dokümanlar** **dizisidir**.   * HFC (Hibrit Elyaf Koaksiyel) Nedir ?   HFC, **optik fiber ve koaksiyel kabloyu birleştiren** geniş bantlı bir telekomünikasyon ağını ifade eder.   * CMTS (Kablo Modem Sonlandırma Sistemi) Nedir ?   **CMTS,** genellikle bir kablo şirketinin merkez bölgesinde, bir kablo ağında dijital modemlerin kablo modemlerle değiştirilmesine izin veren bir cihazdır. Kablolu modem sonlandırma sistemi, abonelere İnternet Protokolü (IP) üzerinden Ses (VoIP) veya kablolu internet gibi yüksek hızlı veri hizmetleri sağlamak için kullanılır.   * Veriler kablosuz ağda yayılırken **reflection’a** mağruz kalırlar yani yayılan sinyalin bir yere çarptıktan sonra geri yansıması mesela dağa çarpıp geri yansıması. * **Obstruction by Objects,** yayılan sinyalin duvarlar, evler vs. nesnesler vasıtasıyla yavaşlaması **interference/noise,** iki telefonun aynı baz istasyona bağlanmaya çalışması. * Her bir düğüm, birbirine bağlı ve birbiriyle konuşabiliyorsa bu ağa **mesh ağı** denir. * ISP'lerin yani internet sağlayıcı şikretlerin birbirleriyle iletişim halinde olmasına **peering link** denir.   ISP'ler için oluşturulan bağlantı yerlerine yani altyapı sağlanmasına **"Internet Exchange Point (IXP)"** denir.   * **Burst-Bursty,** ard arda bir yığın halinde paketlerin gelmiş olmasıdır. * **RTT (Round Trip Time),** bir ileti veya bilgi paketinin bir kaynaktan gidip karşı taraftan geri gelmesine kadar geçen süredir. * **TTL (Time to Live – Yaşam Süresi),** bir sorgulamanın ağ üzerinde ne kadar süreyle önbellekte kalacağını belirleyen bir kavramdır. * **HOPS,** bir paketin (verilerin bir kısmı) kaynağından hedefine geçtiği yönlendiricilerin sayısını ifade eden bir bilgisayar ağ terimidir. Özetle adım sayısı denilebilir. Adım sayısı farklı router'ların kullanılmasına göre değişiklik gösterebilmektedir. Aslında adım sayısı router sayısını verir diyebiliriz. * **Darboğaz,** veri akışı, çeşitli sistem kaynaklarının bant genişliğine göre kontrol edilir. Bir ağ üzerinde çalışan sistem, ağın mevcut kapasitesinin desteklediğinden daha yüksek miktarda veri sağlıyorsa, bir ağ darboğazı meydana gelir. * **Client process** iletişimi başlatan olarak görev yapmaktadır.   **Server process** ise karşı taraflı iletişim haline geçecek bir vaziyette bekler.   * **Header,** paketin gitmesi gereken yeri söyler.   **Local forwarding,** tabloya bakarak yönlendirme işlemidir. **Global routing,** kaynaktan hedefe gidecek olan paketlerin izleyeceği yolların belirlenmesidir, **routing algoritmasıyla** gerçekleştirilir.   * Fiziksel adres **(MAC adresi)** değişmez. Ağ Arayüz Kadı (NIC)'na bu adres bilgisi yakılır.   Mantıksal adres **(IP adresi)** değişebilir ve genellikle bir ağ yöneticisi (Admin) tarafından atanın   * **Subnet mask**, ana bilgisayarın bağlı olduğu ağı tanımlamak için kullanılır.   **Default gateway,** ana bilgisayarın Internet’e veya başka bir uzak ağa erişmek için kullandığı ağ aygıtını tanımlar.     * **Bant genişliği,** ağ sinyalleri arasında kullanılan en yüksek ve en düşük frekanslar arasındaki aralık. Yaygın olarak, bir ağ protokolü veya ortamının, ölçülen throughput (yapılan iş) kapasitesine işaret eder. * **Broadcast Domain:** Gruptaki herhangi bir cihazdan başlatılan broadcast frame’lerini alan cihazların bir grubudur. Router’lar broadcast frame’lerini iletmediklerinden, broadcast domain’leri bir broadcast’ten diğerine iletilmezler. * **Collision:** Ethernet’te aynı anda aktarım gönderen iki düğümün etkisi. Fiziksel ortamda karşılaştıklarında, her bir düğümden frame’ler çarpışacak ve hasar görecektir. * **Connection-Oriented (Bağlantı Tabanlı):** Herhangi bir veri transfer edilmeden önce sanal bir devre oluşturan, veri transfer yöntemi. Güvenli veri transferi için onay ve akış kontrolü kullanır. Connectionless ile zıttır. * **Crossover Kablo:** Bir switch’i switch’e, kullanıcıyı-kullanıcıya, hub’ı hub’a veya switch’i hub’a bağlayan Ethernet kablo çeşididir. * **Data Link Katmanı:** OSI referans modelinin 2. katmanı, fiziksel bir hat üzerinde güvenli veri aktarımından emin olur ve öncelikle, fiziksel adresleme, hat disiplini, ağ topolojisi, hata uyarısı, istenilen frame’lerin taşınması ve akış kontrolü sağlar. IEEE, bu katmanı, MAC alt katmanı ve LLC alt katmanı olarak daha çok sayıda bölümlemiştir. Aynı zamanda Link Katmanı olarak da bilinir. * **Deencapsulation:** Bir katmanın, alt katmandan gelen Protocol Data Unit’deki (PDU) başlık bilgisini sildiği, katmanlaşmış protokoller tarafından kullanılan teknik. * **Default Route:** Routing tablosunda bir sonraki hop’un belirtilmediği frame’leri yöneltmek için kullanılan static routing tablo girişi. * **Enkapsülasyon:** Bir katmanın, üzerindeki bir katmandan Protocol Data Unit’e (PDU) başlık bilgisini eklediği, katmanlı protokol tarafından kullanılan teknik. Örneğin, internet terminolojisinde, bir paket, Network katmanından (IP) bir başlık, Transport katmanından (TCP) bir başlık, uygulama protokol bilgisi ilave edilen bir Data link katmanı başlığı içerebilir. * **Frame:** Bir aktarım ortamına Data Link katmanı tarafından gönderilen bilginin mantıksal bir birimi. * **FTP File Transfer Protocol:** Ağ düğümleri arasında dosya aktarılması için kullanılan bir TCP/IP protokolüdür. Geniş bir dosya çeşidi aralığını destekler ve RFC 959’la tanımlıdır. * **LAN Switch:** Genel olarak, özellikle bir Ethernet switch olarak belirtilen, veri hattı segmentleri arasındaki paketleri aktaran, yüksek-hızlı, çoklu interface’i olan bir köprüleme mekanizmasıdır. LAN switch’ler, MAC adres tabanlı trafiği transfer ederler. * **Maksimum Hop Sayısı:** Sonlandırılmadan önce, bir paketin aktarılmasına izin verilen router’ların sayısı. Bu, bir paketin bir ağda sonsuza kadar dönmesini engellemek için oluşturulmuştur. * **Physical Katman:** OSI referans katmanındaki en alttaki katman (katman 1). Veri frame’lerini, Data Link katmanından (katman 2) elektrik sinyaline çevirmekten sorumludur. Physical katman protokolleri ve standartları, örneğin, pin atamaları ile 0 ve 1 değerinde sinyalleşme için şifreleme planlaması içeren, kablo ve konnektörleri tanımlar. * **Switch (1):** Ağ kurulumunda, frame filtreleme, yayma ve gönderme gibi çoklu fonksiyonlardan sorumlu bir cihazdır. Özel frame’lerin hedefini kullanarak çalışır. Switch’ler OSI modelinin Data Link katmanında çalışır. (2) Yaygın olarak, ihtiyaç duyulduğunda kurulan ve artık ihtiyaç olmadığında sonlandırılan bağlantılara izin verilen elektronik/mekanik bir cihaz. * **Telnet:** TCP/IP protokol ailesinde standart terminal emülasyon protokolü. Uzak terminal bağlantı yöntemi, kullanıcıların uzak ağlarda login olmalarını mümkün kılar ve lokal olarak bağlanır gibi, bu kaynakları kullanırlar. * **Public IP adresi,** internette çıkan cihazlara ISP’ler tarafından yönlendiriciye atanan IP adresleridir. Aynı modeme bağlı cihazlar aynı public IP adresine sahiptir. * **Private IP adresi,** kapalı yerel ağlarda her bir sistemin sahip olduğu IP adresleridir. Belirtilen IP adresleri kapalı yerel ağ içerisinde benzersizdir. * **Authorization,** bir kaynağa erişmek isteyen kişinin kaynak üzerindeki yetkilerinin tanımlanmasıdır. * **Authentication,** bir kaynağa erişmek isteyen kişinin kimliğinin doğrulanmasıdır. * **Topoloji,** ağa bağlı cihazların ağda hangi cihazlara nasıl bağlandıklarını gösteren yerleşim şemalarıdır. * **Bus topoloji,** iletişim omurga denilen tek bir hat üzerinden gerçekleşir. Koaksiyel kablolar kullanılır. Omurganın sonlandırılması için sonlandırıcılar kullanılır. Veri iletilmeden önce hattın boş olup olmadığı kontrol edilir. Ağ kullanıldıysa hattın boş olması beklenir. İletilen veri hedefe ulaşıncaya kadar veya sonlandırıcıya gelene kadar tüm cihazlara uğrar. Ağ performansı düşüktür, omurganın çökmesi durumunda hat düşmektedir. * **Halka topoloji,** halka biçiminde bir ağ üzerinde veri alııcya ulaşıncıya kadar bütün hedeflere uğrar. Veri 3 farklı jeton denilen bir kılavuzla iletilir. Topolojideki bir cihazın arızalanmasıyla hat düşmektedir. * **Yıldız topoloji,** merkezde bulunan bir ağ cihazı üzerinden iletişim gerçekleşmektedir. En yaygın kullanılan topolojidir. Merkezdeki cihazın bozulması sonucu ağ kullanılamaz hale gelir. Cihaz eklemek kolaydır. * **Mesh topoloji,** ağa bağlı bir cihaz ağdaki diğer cihazlara da bağlıdır. Veri kaybını minimalize etmek için güvenilir bir ağ oluşturmak için kullanılır. * **MQTT,** makineden makineye iletişim için kullanılan, standart tabanlı bir mesajlama protokolü veya kurallar dizisidir. Akıllı sensörler, giyilebilir cihazlar ve başka nesnelerin interneti (IoT) cihazları, veri alışverişinde bulunmak için genelde sınırlı bant genişliğine sahip, kısık kaynaklı ağlar kullanmak zorundadır. MQTT, otomotiv, lojistik, üretim, akıllı ev ve ulaşım sektörlerinde kullanılmaktadır. Sınırlı bant genişliğiyle haberleşen IoT cihazları için tasarlanmışlardır. TCP üzerinde çalışan sıralı, kayıpsız ve çift yönlü iletişim sağlayan aktarım protokolüdür. Web soketi üzerinden de çalışabilmektedir.   Protokolün uygulanması basit olduğundan ve IoT verisini kolayca iletebildiği için IoT cihazları bu tarz veri iletimi için MQTT'yi kullanır. MQTT, cihazların birbirleri aracılığıyla bulutla ve bulutun cihazla mesajlaşmasını destekler.   * **CIDR (Classless Inter-Domain Routing),** genel IPV4 adreslerinin tükenmesi ile başa çıkmak için ortaya çıkmıştır. * **InterNIC:** Ağ bilgi merkezidir yani alan adı kayıt hizmetleri hakkında bilgi sunar.   **IANA:** IP adresi verme ve diğer IP kaynaklarından sorumludur.   * **Network Component:** Host’ların bulunduğu ağı yani subnet’i ifade eder.   **Host Component:** O ağa bağlı olan her bir cihazı ifade eder.   * **NAC (Network Access Control),** kurumsal ağlardaki görünürlüğü arttırarak, uç noktalarda ağa erişen cihazların belirli politikalarla sınıflandırılıp riski azaltmak adına ağa erişimin kontrollü sağlandığı bir güvenlik çözümüdür. Erişim kontrolü ile kişinin erişeceği network üzerinde hangi işlemleri yapacağı konusunda yönetim sağlar. * **Synchronization (SYN):** İki bilgisayar arasında gerçekleştirilecek olan bağlantıda kullanılan ilk bayrak "SYN” bayrağıdır. İki bilgisayar arasındaki senkronizasyon (sequence number) işlemini gerçekleştirir.   **Acknowledgement (ACK):** Başarıyla alınan paketleri onaylamak için kullanılan bilgilendirme bayrağıdır.  **Finish (FIN):** Anlaşmalı olarak bağlantının sonlandırılması için kullanılan bayraktır. (Bağlantıyı sonlandırma)  **Reset (RST):** TCP bağlantısında bir sorun olduğunu veya iletişimin olmaması gerektiğini düşündüğünde bağlantıyı sonlandırmak için kullanılır. (Kopma-iptal etme)  **Urgent (URG):** Urgent acil anlamına gelmektedir. Urgent bayrağına sahip bir veri diğer verilerden daha öncelikle olarak uygulama katmanına dahil edilir.  **Push (PSH) :** Paketin hemen gönderilmesi ve işlenmesi için kullanılır. |