**KARAMANOĞLU MEHMETBEY ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**TCP VE UDP TAŞIMA PROTOKOLLERİ**

**VİZE ÖDEVİ**

**Hakan ÇELEBİ**

**Mühendislik Fakültesi**

**Bilgisayar Mühendisliği Bölümü**

**Ders Hocası: ÖĞR. GÖR. KEREM GENCER**

İÇİNDEKİLER

**Sayfa**

[1. GİRİŞ – AĞ KATMANLARI 1](#_Toc443401149)

[1.1 OSI Modeli 1](#_Toc443401150)

1.1.1 Katmanlar ve Görevleri………………………………………………...1

[1.2 TCP/IP Modeli 2](#_Toc443401155)

[1.3 Ağ Neden Katmanlardan Oluşur? 3](#_Toc443401156)

[2. TCP/IP Taşıma Katmanı 5](#_Toc443401157)

[2.1 TCP Protokolü 5](#_Toc443401158)

[2.2 UDP Protokolü 7](#_Toc443401159)

[2.3 TCP ve UDP Protokolleri Farklılıkları 9](#_Toc443401160)

[3. IPV4 Azlığından Kaynaklanan Önlemlerde IP ve Taşıma Katmanı 13](#_Toc443401162)

[3.1 NAT 13](#_Toc443401163)

[3.2 CGN 17](#_Toc443401168)

[KAYNAKLAR 29](#_Toc443401189)

1. [GİRİŞ – AĞ KATMANLARI](#_Toc443401149)

Hostlar (Uç Cihazlar) arasında sohbetin (iletişimin) sağlanması işin modeller geliştirilmiştir. Fakat cihaz firmalarının farklı farklı model kullanmasından ötürü farklı markalar arasında ağ iletişimi sağlanmamaktadır. Bu sorun üzerine

OSI (Open Systems Interconnection – ISO 1984) ve TCP/IP (1978) gibi standart modeller geliştirilmiştir.

* 1. **OSI Modeli**

1984 yılında ISO (International Organization For Standardization) tarafından 7 katmanlı olarak geliştirilmiştir. Fakat TCP/IP Protokol Kümesi popülerleşmesinden ötürü OSI Referans Modelinin yerini almıştır. Katmanlar sırasıyla; Application, Presentation, Session, Transport, Network, Data Link ve Physical olarak isimlendirilir. Veriyi iletecek olan host veriyi gönderirken Application katmanından başlayarak alt katmanlara “header” ekleyerek gönderir. Veriyi alacak olan host ise veriyi en alt katmandan Application katmanına kadar “header” bilgilerine göre çözerek almaktadır.

* + 1. **Katmanlar ve Görevleri**

1.Application: Uygulama katmanı, host cihaz üzerinde veri gönderecek uygulama katmanıdır. Host cihaz ile iletişime geçen katmandır.

2.Presentation: Sunum katmanı, Application katmanına verileri sunar ve gelen verileri biçimlendirir. Ayrıca alıcı hostun verileri Application katmanından okunma garantisi sunar.

3.Session: Oturum katmanı, veriyi ileten host ile alıcı host Presentation katmanlarının arasındaki oturumları kurar ve yönetir.

4.Transport: Taşıma katmanı, veriyi ileten host ile alıcı host arasında iletilen veriyi segmentlere ayırmak, aktarmak ve tekrar birleştirme görevini yapar. Application katmanında kullanılan uygulamaya göre TCP veya UDP protokolleri kullanılır.

5.Network: Ağ katmanı, hostlara atanacak IP adreslerini belirler ve verilerin iletilebileceği en iyi yolu seçer.

6.Data Link: Veri Yolu katmanı, hostları ile fiziksel katman arasında bağlandı kurar. Network katmanından gelen veriyi fiziksel katman için bitlere ayırır.

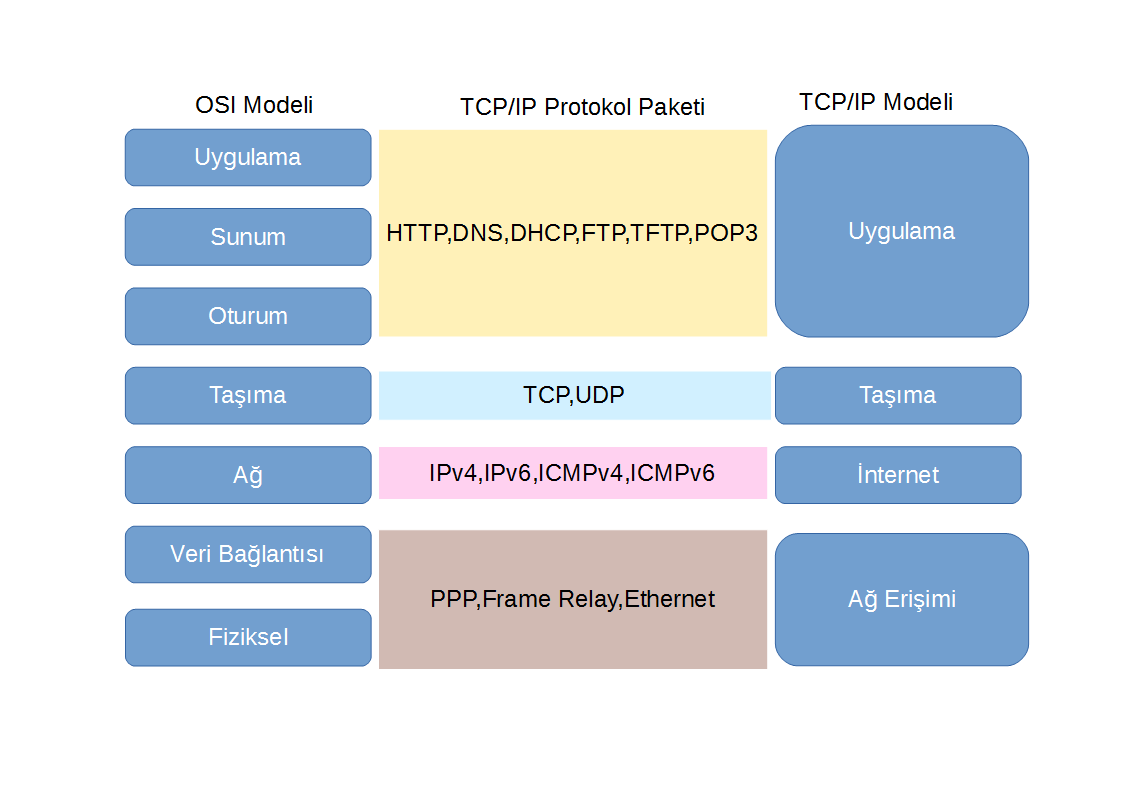
7.Physical: Fiziksel katman, hostlar arasında kullanılan fiziksel medyalar yani fiber optik kablolar, bakır kablolar ile verinin iletilmesini sağlar.

* 1. **TCP/IP Modeli**

Rovert Kahn ve Vinton Cerf 1973 yılında Amerikan Savunma Bakanlığı’nın projesi olan ARPANET üzerinde kullanılması için TCP protokol kümesini geliştirdiler. 1978 yılında ise IP protokol kümesi ile birleşerek TCP/IP protokol kümesini oluşturdular. Ayrıca bu modele Amerikan Savunma Bakanlığı tarafından geliştirildiği işin Depertment Of Defense modeli de denir.

TCP/IP modeli içerisinde 100’den fazla protokol kümesi oluşturmaktadır. Bunlardan en önemlilerinden ikisi ise TCP (Transmision Control Protocol) ve IP (Internet Protokol)’dür.

TCP/IP modeli 4 katmandan oluşur. Bu katmanları OSI Referans Modelindeki bazı katmanların birleşimi olarak düşünebiliriz. Bu katmanlar sırasıyla; Application, Transport, Internet ve Network Acces dir. Application katmanı OSI Referans Modelindeki Application, Presentation ve Session katmanlarının görevini yürütür. Network Acces katmanı ise OSI Referans Modelindeki Data Link ve Physical katmanları görevini yürütür.



TCP/IP modelinde veriler iletilirken veri ileten host ise sırasıyla Application katmanından başlayarak en alt katman Network Acces katmanına kadar her katman sırasında belirli “header” eklenerek iletilir. Bu işleme “encapsulation” yani kapsulleme denir. Veriyi alan host ise Network Acces katmanından başlayarak Application katmanlarına kadar gelen verileri “header” larına göre çözerek Application katmanına kadar iletir. Bu işleme ise “deencapsulation” yani kapsül açma işlemi denir.

Uygulama katmanından gelen veriye PDU (Protoc Data Unit) yani Protokol Veri Ünitesi ismi verilir. Bu PDU Transport katmanına iletilir ve gerekli protokollere göre kapsüllenerek Transport katmanında segment haline getirilir. Daha sonra bu segment ise Internet katmanında gerekli protokollere göre kapsüllenerek Packet kaline getirilir. Network Acces katmanında ise yine protokollere göre kapsüllenerek Frame haline getirilir.

TCP/IP modelinde aynı zamanda katmanlar arasında iletişim devam etmektedir. Örneğin Transport katmanında veriler sana ulaştımı sırası tam mı şeklinde iletişimler kurularak gerekli işlemler yapılır.

* 1. **Ağ Neden Katmanlardan Oluşur?**

Ağ yapısı katmanlardan oluşur çünkü katman modeli tek katmanlı modele göre daha avantajlıdır.

Örnek olarak Bir katmandaki değişiklik veya geliştirilme diğer katmanlardan bağımsızdır. Kullanıcı uygulama üzerinden yorulmadan ve bilmeden ağ’a bağlanarak işlem yapar ve uygulamalar verilerin nasıl iletileceği ile ilgilenmez gibi verilebilir.

1. **TCP/IP Taşıma Katmanı**

Verilen veriyi iletmek veya almak için gerekli protokoller ile düzenleyerek ileten katmandır. Taşıma katmanı iletilen veriyi segmentlere ayırır. Alınan veri ise segment den PDU ya çevrilir. Bu işlemler TCP (Transmission Control Protocol) ve

UDP (User Datagram Protocol) olmak üzere iki protokol ile gerçekleştirilir.

Taşıma katmanının özelliklerinden birtanesi de birden fazla uygulamayı aynı anda kullanabilmemize olanak sağlamaktadır. Yani hostlar arası sohbet sürekli olarak gerçekleştirilebilmektedir.

Peki ya aynı router’a bağlı hostlar arasındaki veri iletiminde host ayrımı nasıl yapılabilmektedir? Burada ise segmentlere atanan bir bilgi sayasesinde host ayrımı yapılabilmektedir. Bu bilgiye port ismi verilir. Taşıma anında hedef port belirlenir ve o IP adresine gönderildikten sonra port bilgisine göre host belirlenir.

* 1. **TCP Protokolü**

TCP protokolü taşıma katmanında segmentlere ayırmada kullanılan protokollerden birisidir. Bu protokol güvenlir bir protokoldür. Bilgisayarımız üzerinden herhangi bir web sitesine erişmek istediğimizi düşünelim ve web sitesini barındıran sunucu tarafından TCP/IP Protokol Kümesi Modeli ile verilerin iletimini inceleyelim.

Öncelikle web uygulaması olduğu için http uygulaması anlamına gelmektedir. Veri uygulama katmanına gönderilir PDU olarak Application Katmanında bulunur. Application katmanı PDU’yu Transport katmanına iletir. Transport katmanında bunun bir web uygulaması olduğunu ve bu yüzden TCP protokolü seçilmesi gerektiğini belirler ve TCP protokolüne göre PDU’yu Segment’e ayırır. Segment içerisinde; Hedef MAC,

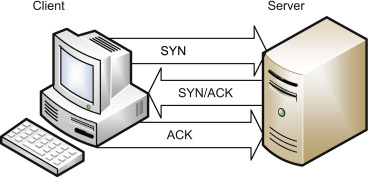
Local MAC, Local Port, Kendi Portu, Hedef IP, Local IP bilgileri yer alır. Daha sonra ise alt katmana Segment iletimi yapılır.

TCP protokolü güvenilir, bağlantı odaklı ve segment sıralamalı bir protokoldür. Uygulama katmanından gelen veriyi belirli byte lara bölerek veriyi sıralı olarak numaralandırır ve alt katmana gönderir. Hedef hosta giden veri ise bu sıraya göre verileri açarak veri kaybı olmadan verilerin iletimi sağlanır. Eğer yolde veri kaybı olursa hedef host eksik numaralı bilgiyi tekrar ister ve iletimi yapan host bilgiyi tekrar gönderir. Eğer tekrar bilgi eksik olursa TCP Protokolünün önemli bir özelliği ile TCP bilginin yolda kaybolduğunu tıkanıklık olabileceğine karar verir ve veri iletimini yavaşlatır. Yani TCP tıkanıklıkların ve veri kayıplarının önüne geçer. Peki tıkanıklık varken veriler nasıl kaybolur. Network’ün performanslı çalışması için veriler bir noktadan sonra yok edilmektedir. Böylece bir verinin Network içerisinde sürekli gezinmesi engellenmiş olmaktadır.

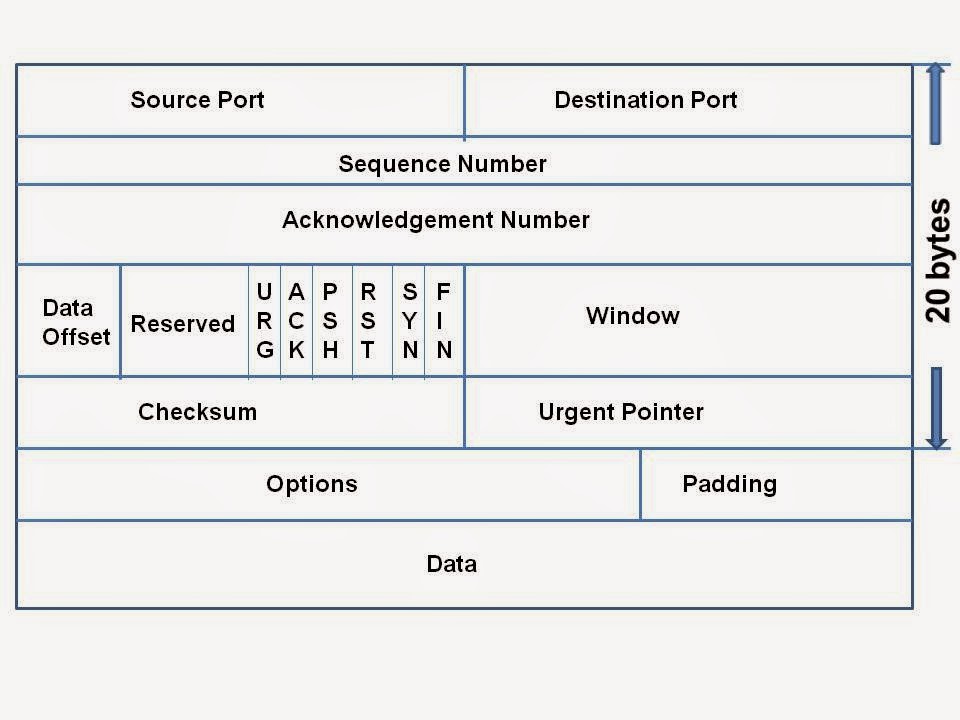
TCP Protokolü verileri sıralı ve düzenli göndererek güvenilir bir protokoldür fakat bu güvenlik önlemlerinden ötürü performansı düşük bir protokoldür. Yani TCP Protokolü; canlı yayın, VoIP (IP üzerinden sesli sohbet) gibi anlık iletim gerekli olduğu yerlerde kullanılmamaktadır.

TCP Protokolü bağlantı odaklı bir protokoldür. Bu bağlantıyı sağlamak için

“Üç Yönlü Doğrulama Kullanır”. Bu yöntem ile veriyi ileten host önce karşıda gerçek bir kişinin olup olmadığını, veri yollarının sağlıklı olup olmadığı gibi çeşitli kontrolleri sağlar. Veriyi ileten host öncelikle iletişim kurma istedeği mesajını SYN gönderir. Bu isteği alan veriyi alan host ise bağlantı istediğini kabul eden mesajını SYN ACK gönderir. Veriyi ileten host ise tamam bağlantıyı kuruyorum demek için ACK mesajını gönderir. Bu şekilde hem yollar arasındaki veri iletim sorunları test edilir. Hem de verinin iletileceği hostun gerçek olduğu doğrulanmış olur.



TCP Segmentinin yapısı ek güvenlik önlemleri çok olduğu için büyük boyutludur.



Görüldüğü üzere TCP segmenti birçok bilgi barındırmaktadır.

Bu bilgiler;

Source Port: Veriyi ileten hostun portudur.

Destination Port: Verinin iletileceği portdur.

Sequence Number: Sıra numarası. Verilerin TCP Protokolü ile segmentlere ayrılırken sıralama verildiğini söylemiştik.

Acknowledgement Number: Verinin başarılı ile iletildiğini ve gelecek veri var olduğunu bildirlen bilgidir.

Data Offset: Segment başlığı her zaman 4’ün katları olmak zorundadır. Bunun için boşluk kullanılır bu Data Offset bilgisi ise kaç boşluk olduğunu yani boşluklardan sonra verinin uzaklığını bildirir.

Reserved: Daha sonra kullanılmak için ayrılmış 6 bitlik bir alandır.

Control Bits: Üç yönlü el sıkışmada da gönderilen SYN gibi bitleri barındırır.

Window: Veriyi iletenden alabileceği veri boyutunu belirten bilgidir

CheckSum: Veri bütünlüğünü doğrulamak için 16 bitlik sağlamadır. Verilerin doğru iletilip ve dağıtıldığını doğrulamak için kullanılır.

Urgent Pointer: Control Bits deki URG biti ile birlikte kullanılır. Acil olan bir veri olduğunu bildirir ve Urgent Pointer içerisinde bu verinin son segmentinin numarası bildirilir bu sayade öncelikli veri sonuna kadar iletilir.

Option: TCP Segmentinin maximum boyutunu belirler.

Data: Host cihazdan gönderilen veriyi tutar.

**2.2 UDP Protokolü**

UDP protokolü daha çok anlık veri iletiminde kullanılır. Örneğin internet yayıncılığı yapacağımızı ve kamera ile görüntümüzü ileteceğimizi düşünelim. TCP Protokolü ile bu işlemi yaparsak çok fazla beklememiz gerekmektedir. Bunun yerine UDP Protokolü kullanılır. Application katmanından gelen PDU UDP protokolüne göre segmentlerine ayrılır ve sadece verinin uzunluğu ve sağlaması yapılır fakat kayıp verilerin peşine düşülmez.

UDP Üç Yönlü Doğrulama kullanmaz yani doğrulama yönelimli bir mekanizmaya sahip değildir.

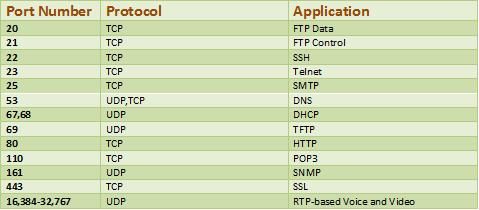
Segmentlerinde sıralama yoktur böylece herhangi bir veri kayıp olduğunda kaybolan veri tekrar gönderilmez ve veri kaybı önlenemez.

En hızlı teslim etmek için geliştirilmiştir. Daha çok anlık iletişimlerde kullanılır.

Ayrıca UDP ile gönderiler verilerin karşı tarafa gidip gitmedi veya karşı tarafın gerçek bir alıcı olup olmadığını öğrenemeyiz bu nedenle güvenliksizdir.

TCP Segmenti yapısında sadece; kaynak port, hedef port, veri uzunluğu ve sağlama bulundurur.

En çok kullanılan TCP/UDP Portları aşağıdadır.



Uygulamada server-host olmamız gerektiğinde router üzerinden port forwarding yapmamız gerekmektedir. Alıcı host olarak tüm portlardan gelen istekler kabul edilmektedir fakat veri ileten host olmak istediğimiz taktirde port forwarding yapmak gerekmektedir. Örneğin windows remote desktop ile bilgisayarımıza online olarak bağlanmak istiyoruz veya kameramıza online olarak bağlanmak istiyoruz bunun gibi tüm işllemler için port forwarding gerekmektedir. Yani bu porttan veri göndereceğim alınmasını talep ediyorum gibi düşünülebilir.

Port Forwarding tetikleme olmadığı zaman yani veri iletimi olmadığı zaman pasif moda geçer. Veri iletim isteği olduğu anda aktif olmaktadır.

1. **IPV4 Azlığından Kaynaklanan Önlemlerde IP ve Taşıma Katmanı**

İnternete erişimimizi IPV4 yani Internet Protocol Version 4 üzerinden yapıyoruz. IPV4 4’er tane 8 bit den toplamda 32 bitlik numaralardan oluşmaktadır. İnternete erişim sağlayan kullanıcı sayısı arttığı için de IPV4 adreslerinde kıtlık çekilmeye başlandı. Bunun üzerine çeşitli şekillerde çözümler üretilmeye başlandı.

* 1. **NAT (Network Acces Translation)**

NAT IPV4 sınırlılığını aşmak için geliştirilen bir yöntemdir. Normalde internete ortak modem veya herhangi bir yol ile bağlanan bir kullanıcıya bir IPV4 adresi atanmaktaydı. NAT ile aynı router a bağlı hostların tümüne aynı ip atanmaktadır.

Bu yöntem TCP/UDP Protokolünün çalışmasında da sorun çıkarmaktadır. TCP/UDP Protokolü port numarası ile yolunu bulmaktadır demiştik. Dolayısı ile aynı router a bağlı hostların kullanabileceği TCP/UDP Protokol port sayısı sınırlandırılmış oldu. Örneğin yeni bir girişimde bulunmak istiyoruz ve internet yayıncılığı yapılması için ekipmanları ile bir işletme açtık. İnternet sağlayıcısı NAT sistemi ile internet sağlıyor bu yüzden UDP portlarınızın tamamı tükendi ekstra olarak işletme büyüyemez hale geldi. Fakat yinede bu port sayısı gün geçtikçe büyütülmeye devam etmektedir. Bu sorunun yinede üstesinden gelinmektedir.

**3.2 CGN (Carrier Grade NAT)**

CGN sistemi bir IPV4 adresini portlar yardımı ile birden fazla router a atanmasını sağlayan sistemdir. Bu sistemin dezavantajlarından en önemlisi kişisel cihazını server-host haline getiremiyorsunuz. Bunun anlamı IOT kullanamıyorsunuz. Remote Desktop veya Evdeki kameranıza erişemiyorsunuz. Türkiye’de tüm internet sağlayıcıları CGN ile internet sağlamaktadır. Telefon operatörleri ise uzun süredir CGN kullanıyor.

**KAYNAKÇA**

<http://www.tcpipguide.com/>

<https://tr.wikipedia.org/>

<https://www.youtube.com/BilgisayarKavramlari>

Ağ Yöneticiliğinin Temelleri Cemal TANER Kitabı