

Tıkanıklık Kontrolü için Yeni Bir İletim Protokolü: DCCP

Ayşegül ALAYBEYOĞLU, Yonca BAYRAKDAR, Aylin KANTARCI

Ege Üniversitesi, BilgisayarMühendisliği Bölümü, İzmir
{aysegul.alaybeyoglu, yonca.bayrakdar, aylin.kantarci}@ege.edu.tr

Özet: Video akışlandırma gibi gerçek zamanlı uygulamalarda, verinin zamanında iletimi güvenli iletimine göre daha büyük öncelik taşır. Bu yüzden TCP gibi güvenli iletim sağlayan bir protokol bu tarz uygulamalar için uygun değildir. Geçmişte, zamanında iletimin önemli olduğu uygulamalar için UDP protokolü kullanılmış fakat zaman içinde iletim esnasındaki tıkanıklık kontrol mekanizmasının önemi anlaşılmıştır. Bu çalışmada UDP protokolünün yerine kullanılması düşünülerek tasarlanmış, yeni bir protokol olan DCCP protokolü tanıtılmıştır. Bu bildiri de DCCP protokolünün temel özelliklerinden bahsedilmiş, tıkanıklık kontrol mekanizmasında TCP'den farkları belirtilmiştir.

Anahtar Kelimeler: DCCP, TFRC, CCID2, CCID3, Tıkanıklık Kontrolü

A New Transport Protocol for Congestion Control: DCCP

Abstract: For real time applications such as video streaming, it is more important to transmit the data on time than to transmit the data without loss. For this reason, reliable protocols like TCP are not suitable for that kind of applications. In the past, firstly UDP was used for the applications that need the data to be transmitted on time, but later the importance of congestion control during transmitting was realized. In this paper, alternative protocol called DCCP is introduced. In this paper, main features of DCCP are explained and its differences from TCP on congestion control issue are specified.

Keywords: DCCP, TFRC, CCID2, CCID3, Congestion Control

1. Giriş

Çoklu ortam uygulamalarının ve bu uygulamaları destekleyen ağ türlerinin çeşitliliği ve çokluğuna rağmen, bu ağ ve uygulama kombinasyonlarıyla eşleşmiş olan protokoller ortak bir yapıya sahiptir. Paket anahtarlamalı, devre anahtarlamalı, bağlantı tabanlı veya bağlantısız gibi çeşitli türlerde çalışan farklı ağlar kendine arayüz olacak farklı bir protokol kümesine sahiptir. Protokol seçimi uygulamanın ihtiyaçları doğrultusunda karar verilecek bir konudur. Verinin iletilmesine dair güvence gerektiren ancak gecikmelere çok hassas olmayan uygulamalar için TCP (Transmission Control Protocol) protokolü tercih edilirken,

birtakım veri kayıplarına tolerans gösterilebilecek ancak gecikmelerin daha fazla önem azettiği uygulamalarda ise UDP (User Datagram Protocol) protokolü tercih edilebilir. Eğer uygulama, gerçek zamanlı ses ve video akışlandırma içermekteyse ve alıcı aldığı paketlerin senkronizasyonu için bir zamanlama bilgisine ihtiyaç duyuyorsa, RTP (Real-time Transport Protocol) ve ilişkili olarak da RTCP (Real-time Transport Protocol) protokolleri kullanılabilir.

TCP protokolü alıcı ve gönderici arasında iki yönlü ve güvenilir bir veri hattı kurulmasını gerektirir. Her bir veri paketinin alıcıya ulaşımını garantileyen bilgilendirme mesajları gön-

dericiye geri besleme hattından iletilmektedir. Herhangi bir kayıp esnasında gönderici hangi paketin kaybolduğunu bilir ve alıcıya o paketi yeniden yollar. Bu şekilde her paketin alıcı tarafla ulaşması garantilenmiş olur.

Ancak çokluortam verilerinin iletiminde geri beslemenin kullanılması gecikme yaratır ve bu istenmeyen bir durumdur. Dolayısıyla çokluortam verileri için TCP protokolünün kullanımını uygun değildir. Bunun yerine daha çok UDP kullanımı tercih edilir. Çünkü güvenilir bir protokol olmayan UDP, ağ üzerinden veri paketini gönderir ve alıcıya ulaşmış olup olmadığını bilgisi ile ilgilenmez. Dolayısıyla paketin yerine ulaşmış olup olmadığını garantisini verme yetkisi yoktur. Ayrıca TCP'ye göre daha küçük başlık bilgisi içerdiği için ağ üzerinde fazla yer kaplamaz.

UDP hızlı iletişim kurulması gereken durumlarda oldukça kullanışlıdır. Çünkü TCP'de olduğu gibi, veri iletim kontrolleri içermediğinden daha basit bir yapısı vardır. UDP, veri paketlerinin sırasının önemli olmadığı uygulamalarda kullanılmak üzere tasarlanmıştır.

Yüksek kalitedeki video aktarımlarında, ağ kapasitesi aşılabılır ve yönlendiricilerde paketler birikebilir. Bu durumda yine paket kayıpları ve gecikmeler söz konusu olabilir. Çokluortam verilerinin gönderiminde hiç kontrol olmaması da istenen bir durum değildir. Yukarıda bahsi geçen durumlarda sıkışıklık kontrolleri gerekmektedir.

RTP, gerçek zamanlı ses ve görüntü gibi verilerin bir uçtan diğer uca taşınmasını sağlayan bir protokoldür. RTP UDP tabanlı bir protokol olup, RTCP (RealTime Transport Control Protocol) isimli bir protokol ile birlikte kullanılmaktadır. RTP servis kalitesi kontrolü sağlamaz, ancak RTCP servis kalitesinin izlenmesini sağlar. RTP portundan veri gönderilirken, aralıklı olarak RTCP protokolü ile göndericiye

veri trafiği hakkında geri besleme yapılır. Her RTP paketinin bir sıra numarası vardır. Alıcı eksik sıra numaralarına bakarak son RTCP gönderiminden sonraki kayıp oranı hakkında istatistikler çıkarır. Bu istatistikler bir RTCP paketine gönderilerek göndericiye gönderilir. Sıkışıklık kontrolü yaparak veri hızını düşürmek uygulamaya bırakılmıştır. Paket geliş zamanları arasındaki varyasyon gibi istatistikler de RTCP raporları ile gönderilir.

RTP başlıbaşına verinin gönderimi ile ilgilenirken, RTCP, RTP'ye sistem seviyesinde fonksiyonellik eklemektedir. RTCP, RTP'nin yanında çalışır ve aralarında ortak bilgi kullanımını söz konusudur. Fakat RTCP farklı port numarasına sahip olduğu için RTP'den bağımsız olarak da çalışabilmektedir[1].

Video akışlandırma gibi gerçek zamanlı uygulamalar için, verilerin zamanında iletimi, güvenli iletiminden daha büyük önem taşımaktadır. Bu yüzden TCP gibi iletim esnasında herhangi bir kayıpta akışın durdurulup kaybolan verinin tekrar gönderildiği bir protokol, zamana karşı duyarlı uygulamalar için uygun olmamaktadır. Bu tarz uygulamalar için geçmişte çoğunlukla içinde herhangi bir tıkanıklık mekanizması barındırmayan UDP protokolü kullanılmıştır. Son yıllarda, UDP'ye bir takım kontrol mekanizmaları eklenerek güvenilir olmayan veri iletimi sağlayan DCCP (Datagram Congestion Control) adında bir protokol önerilmiştir. Böylece DCCP ile iletim esnasındaki tıkanıklık durumları minimuma indirgenmeye çalışılmıştır.

Bu çalışmada yeni bir protokol olan DCCP protokolünün tanıtılması amaçlanmıştır. Bildirinin ilerleyen bölümlerinde DCCP'nin UDP ve TCP protokollerinden farkları, DCCP Paket yapısı, DCCP'de bağlantı kurulumu ve son olarak da DCCP'deki tıkanıklık kontrol mekanizması konularından bahsedilmiştir.

2. Datagram Congestion Control Protocol (DCCP)

DCCP [2], çoklu ortam trafiğini desteklemek üzere IETF topluluğu tarafından önerilmiş bir iletim protokolüdür. Güvenilir olmayan veri iletimi gerçekleştirilmektedir. Bağlantı kurulumu, sonlandırımı ve uygun tıkanıklık kontrol mekanizmasının seçimi için anlaşmalar yapılmaktadır. Herhangi bir paket kaybı olduğunda, Explicit Congestion Notification (ECN) [3] olarak adlandırılan bir bilgilendirme mekanizması kullanmakta, bu mekanizma ile de göndericiye, alıcının hangi paketleri alabildiğine ve hangi paketlerin kaybolduğuna dair bilgiler gönderilir. ECN mekanizmasında, tıkanık durumda olan yönlendiriciler (router), paketleri atmak yerine paketleri işaretleme işlemini gerçekleştirirler.

DCCP protokolünün daha önce bahsedilen UDP ve TCP protokolleri ile karşılaştırıldığında, UDP protokolünden tıkanıklık kontrolü, tokalaşma ve bağlantı kurulumu gibi ek özellikleri taşıması ile farklılıklar gösterdiği, TCP protokolünden ise güvenilirlik, sıralı paket iletimi ve bayt akışı özelliklerini taşımaması ile farklılık gösterdiği görülür.

2.1. DCCP'de Paket Yapısı

DCCP protokol fonksiyonları on adet paket türü ile gerçekleştirilmektedir. Bu paketlerden sekizi bağlantı kurulumu için kullanılırken diğer ikisi de senkronizasyonu sağlamak için kullanılır. DCCP-Request paketi, istemci tarafından bağlantının kurulmasının başlaması için gönderilir. DCCP-Response paketi, sunucu tarafından DCCP-Request paketine cevap olarak gönderilir. DCCP-Data paketi ile uygulama verisi, DCCP-Ack paketi ile yalnız onay bilgisi, DCCP-DataAck paketi ile de uygulama verisi ile onay bilgisi birlikte gönderilmektedir. DCCP-CloseReq paketi sunucunun istemciye bağlantıyı kapatması için gönderdiği bir istek paketi iken DCCP-Close paketi sunucu ya da

istemcinin bağlantıyı kapatması için kullandığı bir pakettir. DCCP-Reset paketi, bağlantının beklenmeden bir durumda sonlandırımı için kullanılmaktadır. DCCP-Sync ve DCCP-SyncAck paketleri paket iletiminde senkronizasyonu sağlamak için kullanılır. Her bir DCCP paketi birer sıra numarası taşır böylece paket kayıpları kolayca belirlenip rapor edilebilir. TCP protokolünde sıra numaraları bayt tabanlı olarak artış gösterirken, DCCP protokolünde sıra numaraları her bir paket için bir artırılır. Bir DCCP paketi, başlık ve içerilmesi zorunlu olmayan uygulama verisinden oluşur[4].

0		7		15		31	
Source port				Dest Port			
Data Offset		ccval		CsCov		Checksum	
Res	Type	X	Res		Sequence Number(high bits)		
Sequence Number(low bits)							

Şekil1. DCCP Paket Genel Başlığı
(Generic Header)

Şekil1'deki tabloda görülen alanlar incelenecek olursa ccval alanı, tıkanıklık kontrolü tanımlayıcı değerini, type alanı paket tipini, X alanı genişletilmiş sıra numarası bitini içermektedir. Bu X değerine bağlı olarak iki adet genel başlık türü vardır. Eğer X değeri 0 ise sıra numarası alanı 24 bit, eğer bu değer 1 ise sıra numarası alanı 24 bit daha esner ve toplamda 48 bitlik bir alan olur [5].

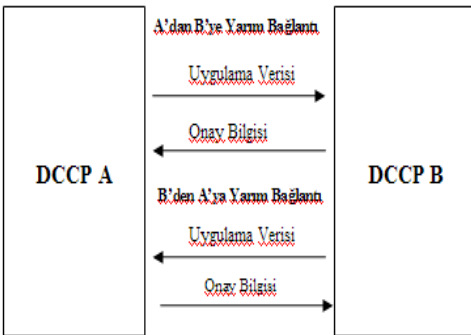
2.2. DCCP'de Bağlantı Kurulumu

DCCP bağlantısı kurulmadan önce istemci ve sunucu arasında hangi tıkanıklık mekanizmasının ve parametrelerinin kullanılacağına dair bir anlaşmaya varılmış olması gerekir. Her bir uç noktanın uygulayacağı mekanizmalar ve bu uygulamaların da bir takım gereksinimleri vardır. Bu yüzden bağlantı kurulumu yapılmadan önce sunucu ve istemci arasında bir takım öngörüşmeler yapılmaktadır.

DCCP, güvenilir olmayan bir veri iletimi gerçekleştirirse de, onay bilgilerinin gönderiminin güvenilir olması beklenir. Bunun içinde gön-

dericinin, alıcının göndermiş olduğu onay bilgilerinden birisini onaylaması gerekmektedir. DCCP’de, göndericinin bu onaylama işlemini gerçekleştirebilmesi için kullanılan yöntem, alıcının gönderdiği onay bilgisinde paket sıra numarası bilgisinin bulunmasını sağlamaktır. Böylece gönderici onay bilgisini aldığı anda, alıcıya hangi pakete ait onay bilgisinin alındığını rapor edebilmektedir. DCCP’de TCP’den farklı olarak onay bilgilerine de tıkanık kontrolü yapılabilmektedir.

DCCP’de bağlantılar iki yönlüdür yani veri ve onay bilgisi eş zamanlı olarak her iki yönde iletelebilmektedir. Bu durum video akışlandırma uygulamalarda kullanılan DCCP gibi bir protokol için uygun değildir. Çünkü sunucudan istemciye video gibi büyük bir boyutta veri iletilirken, istemciden sunucuya sadece dosya ismi gönderilebilmektedir. Yani tek bir bağlantıda asimetrik bir durum oluşmaktadır. Bu duruma çözüm olarak, DCCP bağlantısının iki yarım bağlantıya ayrılması önerilmiştir. Alıcıdan göndericiye, göndericiden alıcıya olan bağlantıların herbirine yarım-bağlantı (half-connection) denmektedir ve bu yarım bağlantıların her biri tıkanıklık kontrol mekanizmasına sahiptir [2].



Şekil2. DCCP’de Bağlantı Durumu

Şekil2’de de görüldüğü gibi A ve B arasındaki bağlantıda, ilk yarım bağlantı A’dan B’ye veri paketlerini ve B’den A’ya onay bilgisini

si içerirken, ikinci yarım bağlantı ise B’den A’ya veri paketlerini ve A’dan B’ye de onay bilgisini içerir. Bu iki yarım bağlantı birbirlerinden bağımsız bir şekilde eş zamanlı olarak gerçekleşebilmektedir. Her bir yarım bağlantı için kullanılan tıkanıklık kontrol mekanizması farklı olabilmektedir.

2.3 DCCP’de Tıkanıklık Kontrol Mekanizması

DCCP, birçok tıkanıklık kontrol mekanizmasını içermekte ve bu mekanizmalardan en uygun olanını dinamik olarak seçebilmektedir. Her bir tıkanıklık kontrol algoritması birer tıkanıklık kontrol tanımlayıcısı (CCID) ile belirtilir. CCID2 ve CCID3, IETF tarafından standartlaştırılan kontrol tanımlayıcılarıdır. CCID2 [6], TCP gibi pencere tabanlı bir tıkanıklık kontrol algoritmasıdır ve daha çok online oyunlar gibi uygulamalar için uygundur. CCID3 [7] ise, TCP Friendly Rate Control (TFRC) algoritmasıdır ve daha çok IP telefonculuğu gibi iletim oranında ani değişimler yerine hassas değişimlerin olduğu uygulamalar için uygundur.

1.1.1TCP-like Congestion Control (CCID2)

CCID2, daha çok sürekli değişen koşullara sahip ortamlarda maksimum bant genişliğinden yararlanabilecek ve tıkanıklık penceresindeki ani değişimlere uyum sağlayabilecek uygulamalar için uygundur. CCID2, TCP protokolünün de kullandığı artımsal artış çarpımsal azalış (Additive Increase Multiplicative Decrease) tıkanıklık kontrol mekanizmasına benzer bir mantıkta çalışır. Buna göre, gönderici bir tıkanıklık penceresi oluşturur ve bu pencere doluncaya kadar paket iletimine devam eder. Paket alıcı tarafından onaylanarak alınır. Alıcı tarafından ECN bilgisinin gönderilmesi tıkanıklık olduğunu belirtir. Bu durumda tıkanıklığın giderilmesi için tıkanıklık penceresi yarıya düşürülür. CCID2, uzun vadede maksimum bant genişliği kullanımı sağlarken, herhangi bir tıkanıklık durumunda tıkanıklık pencere-

sini direk yarıya düşürmesi ile veri gönderim oranında ani değişimlere neden olur. Bu özellikler de CCID2'yi video akışlandırma gibi gerçek zamanlı uygulamalardan ziyade online oyunlar için daha uygun kılar.

Birçok benzerliğe rağmen CCID2'deki tıkanıklık kontrolü ile TCP protokolündeki tıkanıklık kontrolü arasında bir takım farklılıklar vardır. Bunlardan birisi güvenilir olmayan DCCP protokolünde kaybolan paketlerin yeniden gönderiminin olmamasıdır. Bir diğer farklılık da TCP'deki iletim, bayt akışı halinde olurken DCCP'de paket iletimi söz konusudur.

1.1.2 TCP Friendly Rate Control –TFRC (CCID3)

TFRC, bir eşitlik tabanlı (equation-based) tıkanıklık kontrol mekanizmasıdır. Daha önce bahsedilen AIMD tıkanıklık kontrol mekanizması, herhangi bir tıkanıklık olduğunda iletim oranını direk yarıya düşürürken, eşitlik tabanlı tıkanıklık kontrol mekanizması ile iletim oranı en son birim zamandaki kayıp sayısına bağlı olarak belirlenmektedir. Gönderici, alıcıdan tıkanıklık durumu ile ilgili geri bildirim almaktadır ve (1)'de verilen tıkanıklık kontrol denklemine göre de iletim oranını belirlemektedir.

$$T = \frac{s}{R\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{2p}{s}})p(1 + 32p^2)} \quad (1)$$

Bu denklemde T maksimum gönderim hızına karşılık gelmektedir. s parametresi paket büyüklüğünü, R parametresi gidiş dönüş zamanını, p parametresi kayıp olayı sayısını ve t parametresi de yeniden gönderim zaman aşımını ifade etmektedir. Tıkanıklık kontrolü için (1)'de belirtilen denklemin kullanılabilmesi için R ve p parametrelerinin tanımlanması gereklidir. Burada kayıp olay sayısını ifade eden p parametresi alıcıda hesaplanırken, gidiş dö-

nüş zamanını ifade eden R parametresi de alıcı yada göndericide hesaplanabilir. Alıcı p parametre değerini göndericiye geribildirim yapar. Gönderici de bu geribildirimi yorumlayarak, (1) denklemine göre bir gönderim hızını hesaplar. Eğer mevcut gönderim hızı, hesaplanan gönderim hızından düşükse artırılır, yüksekse de azaltılır.

Eşitlik tabanlı tıkanıklık kontrol mekanizmasında, uygulama için en uygun kontrol denkleminin seçilmesi çok önemlidir. CCID3, CCID2 ile karşılaştırıldığında, CCID3'teki iletim oranında daha az değişimin olduğu görülür. Bu özellikler CCID3'ü, iletim oranının önem taşıdığı video akışlandırma gibi uygulamalar için daha uygun hale getirir. TFRC, pencere tabanlı değil hız tabanlıdır. Göndericinin alıcıdan olumlu bir geribildirim almaması durumunda, iletim oranının düşürülmesi için mekanizmaya ihtiyaç vardır. Tıkanıklık durumunda iletim oranının düşürülmesi ile ilgili bir takım çalışmalar yapılmıştır. [8]'de herhangi bir paket kaybı olduğunda iletim oranı en sonki iletim oranı değerinin 7/8'ine düşürülmüştür. [9]'da ise Van Jacobson, iletim oranının 7/8 yerine 1/2 oranında düşürülmesi gerektiğini belirtmiştir.

Eşitlik tabanlı tıkanıklık kontrolünde temel hedef en uygun bantgenişliğini kullanmaktan ziyade en uygun gönderim hızını belirlemektir. Bu tıkanıklık kontrol yapısının TCP'deki tıkanıklık kontrol mekanizmasından temel farklılıklarından biri, iletim oranını tek bir kayıp için değil ardıl kayıplar olduğunda yarıya düşürmesidir. Bir diğer farklılık da kayıp oranı düşünce iletim oranının TCP'den daha yavaş artırılmasıdır.

Alıcının en az bir gidiş-dönüş (round-trip time) zamanı için göndericiye paket alıp almadığına dair geri bildirimde bulunması ve göndericinin bir kaç gidiş-dönüş zamanı içinde herhangi bir geribildirim alamaması durumunda iletim ora-

nını düşürmesi gibi özellikler, eşitlik tabanlı tıkanıklık kontrol mekanizmasının tasarlanmasındaki temel amaçlardan birisidir [10].

3. Sonuç

Son yıllarda video akışlandırma, online oyunlar, internet telefonculuğu gibi uygulamalara olan ilgideki artış, bu uygulamalar için kullanılan protokollerde gelişmelerin olmasında tetikleyici olmuştur. UDP, bu tarz uygulamalar için çok sık kullanılan fakat herhangi bir tıkanıklık kontrol mekanizması içermeyen bir protokoldür. Zaman içinde tıkanıklık kontrolünün önemi anlaşılmış ve UDP protokolü temelleri üzerine bir takım tıkanıklık kontrol mekanizmaları eklenerek DCCP protokolü oluşturulmuştur. Bu çalışmada da bu önerilen yeni protokolün özelliklerinden bahsedilmiş, en önemli özelliği olan tıkanıklık kontrol mekanizmasına değinilmiştir. DCCP, TCP ve UDP'nin avantajlarını bir araya getirerek tıkanıklık kontrolü mekanizmasında başarılı olmuştur ve ileride çokluortam uygulamaları için çok tercih edilen bir standart olması beklenmektedir.

4. Kaynaklar

[1] Fred Halsall, Multimedia Communications, Addison Wesley, 2001

[2] E. Kohler, M. Handley, S. Floyd, "Datagram Congestion Control Protocol", <http://www.icir.org/kohler/dc>

[3] K. K. Ramakrishnan, S. Floyd, D. Black, "The addition of explicit congestion notification(ECN) to IP. RFC 3168", Internet Engineering Task Force, Sept. 2001

[4] E. Kohler, M. Handley, S. Floyd, "Designing DCCP: Congestion Control Without Reliability", <http://www.icir.org/floyd/papers/kohler06designing.pdf>

[5] C. Xu, J. Lu, C. Zhao, "Performance analysis of transmitting H.263 over DCCP", IEEE Int. Workshop VLSI Design & Video Tech. Suzhou, China, May 28-30, 2005, s. 328-331.

[6] Sally Floyd, Eddie Kohler, "Profile for DCCP Congestion Control ID2: TCP-like Congestion Control", <http://www.icir.org/kohler/dccp>

[7] Sally Floyd, Eddie Kohler, "Profile for DCCP Congestion Control ID3: TFRC Congestion Control", <http://www.icir.org/kohler/dccp>

[8] R. Jain, K. Ramakrishnan, and D. Chiu. Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Tech. Rep. DEC-TR-506, Digital Equipment Corporation, August 1987.

[9] V. Jacobson. Congestion Avoidance and Control. SIGCOMM Symposium on Communications Architectures and Protocols, pages 314–329, 1988. An updated version is available via <ftp://ftp.ee.lbl.gov/papers/congavoid.ps.Z>.

[10] Floyd, S., Handley, M., Padhye, J., and Widmer, J., "Equation-Based Congestion Control for Unicast Applications," Proc. of SIGCOMM '00, 2000