# AÇIK KAYNAK KODLU YAZILIM TABANLI FİREWALLARIN PERFORMANS DEĞERLENDİRMESİ

**Devrim Seral**
Uluslararası Kıbrıs Üniversitesi
dseral@ciu.edu.tr

## ABSTRACT

Everyday large number of hosts being connected to Internet, securing and screening these hosts become more important. For this reason firewall systems should be used. The cost of hardware based firewall systems are high and alternatively ordinary PC with Open Source Software Based firewall systems may be used. GNU/Linux netfilter and OpenBSD PF are most known and widely used Open Source Software based firewall systems. This paper evaluates, UDP and TCP protocol data forwarding performance by considering various rule numbers of these systems. For performance benchmarking netperf application unidirectional TCP and UDP tests used. Also under attack performance behaviour of these firewalls provided.

**Key words:** Open Source, Software Based Firewall, Firewall, Firewall Performance

## 1. GİRİŞ

İlk ağ bağlantısının yapılmasının üzerinden kırk yıldan fazla zaman geçti [1]. Bu bağlantı belkide ileride yaşadığımız çağa adını verecek Internet'in ilk adımının atılmasına yol açmıştır. Internet artık hayatımızın her evresine giren ve milyarlarca insanın bağlı olduğu [2] sınır tanımayan bir ağ haline gelmiştir. Internet bağlantısı sayesinde birçok alandaki işlemler hızlı ve pratik şekilde yapılabilmektedir. Bundan dolayı gerek bireyler, gerekse kurumlar, Internete bağlanarak avantajlarından yararlanmaya çalışmaktadırlar. Ancak Internet bağlantısı, getirdiği avantajlar yanında özellikle ağ ve veri güvenliği tarafında bazı sorunlara yol açabilmektedir.

Bilgisayar ağlarındaki güvenliği artırmak ve başka ağlardan gelebilecek saldırıları önlemek için kullanılan en önemli araçlardan biri de Firewall'lardır. Firewall'lar ağlar arasında akan trafiği izleyerek, belirlenen kurallar dahilinde paketlerin geçişini kontrol eden sistemlerdir [3]. Paket geçiş kontrolü paket filtreleme olarak adlandırılan işlem ile yerine getirilir [3]. Firewall sistemleri yazılım veya donanım temelli olabilmektedir. Firewall sistemleri içerisinde yazılım tabanlı olanlar, fazladan sundukları saldırı önleme (IDS), proxy ve paket işaretleme gibi özelliklerden dolayı önemli bir yer tutarlar [4,5].

Yazılım tabanlı firewall sistemleri, genelde PC sınıfı bilgisayarlardan ve sıradan ağ iletişim kartları ile yapılandırılırlar ve Linux, FreeBSD yada OpenBSD gibi Açık Kaynak Kodlu (AKK) işletim sistemleri üzerinde çalışırlar [5,6]. Bu yüzden toplam sahip olma maliyetleri, donanım temelli firewall sistemlerine göre çok daha düşük olmaktadır.

Günümüzde Internet bağlantı hızlarının artmasıyla birlikte özellikle kurumlar için yüksek hızlarda çalışabilecek firewall sistemleri önem taşımaya başlamıştır. Birçok kurum düşük maliyetli ve yüksek performanslı olduğu için AKK işletim sistemleri üzerinde çalışan yazılım tabanlı Firewall sistemlerini tercih etmektedir.

Yazılım tabanlı firewall sistemlerinin performansı etkileyen iki temel nokta bulunmaktadır. Bu noktalardan biri donanımsal sınırlamalar diğeri ise işletim sisteminin ve firewall'ın yazılımsal sınırları olarak verilebilir. Donanımsal sınırlamalara örnek olarak firewall sisteminde kullanılan ağ bağlantı kartlarının bağlı olduğu veri yolunun hızı verilebilir. PCI veriyolunun hızı 1064Mb/s (32 bit, 33Mhz) ve 4256Mb/s (64 bit, 66Mhz) aralığındadır [6]. Yani teorik olarak 32 bit ve 33 MHz bir ağ kartı kullanarak en fazla 1064Mb/s hız elde edilebilir. Ancak veriyollarının ortak kullanıldığı göz önüne alındığında bu hızlara erişmek mümkün olamamaktadır. Yazılım tarafında ise, yazılım tabanlı firewall'ın tasarımı ve paket filtreleme işleminin hızı performansı etkileyen önemli bir etkendir [7,8]. Diğer yandan işletim sisteminin ağ katmanındaki sınırlamalarıda başka bir etken olarak verilebilir [9].

Yazılım tabanlı firewall sistemlerinin performansı ile ilgili birçok çalışma yapılmıştır [10,11,12]. Yapılan çalışmalar, firewall sisteminin performansını çeşitli test yazılımları ile ölçmeye dayanmaktadır. Bu çalışmada ise Açık Kaynak Kodlu (AKK) işletim sistemlerinde en fazla kullanılan iki yazılım tabanlı firewall sisteminin performans değerlendirmesi yapılacaktır. Bunlardan biri GNU/Linux işletim sisteminde çalışan netfilter [13] diğeri ise temel olarak OpenBSD işletim sistemi üzerinde geliştirilen ancak diğer BSD işletim sistemi türevlerinde de çalışabilen PF yazılımıdır [14].

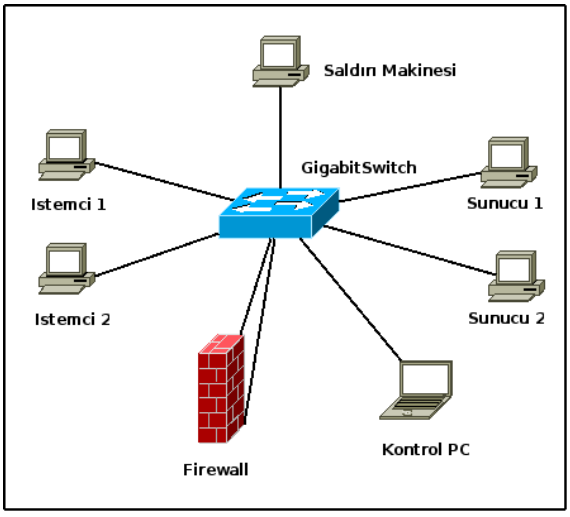
Bu makalenin geriye kalanı şu şekilde organize edilmiştir. 2. Bölümde test ortamı ile ilgili ayrıntılı bilgi verilmiştir. 3. Bölümde Linux netfilter ve OpenBSD PF yazılım tabanlı firewalların performans değerleri verilmiştir. 4. Bölümde ise AKK yazılım tabanlı firewall sistemlerinin performans değerlendirmeleri ve ileride yapılacak çalışmalar için öneriler sunulmuştur.

## 2. TEST ORTAMI

### 2.1 Ağ Ortamı

Yazılım tabanlı firewall sisteminin performansını ölçmek için Şekil 1'de gösterildiği gibi bir ağ ortamı oluşturulmuştur. Bu ortamdaki ikisi istemci, ikisi sunucu ve biri'de saldırı amaçlı kullanılmak üzere yapılandırılmış makineler Intel Core 2 Duo, 1Gb Ram ve gigabit ağ arabirim kartına sahiptir. Gigabit ağ anahtarı olarak Cisco Catalyst 3560G kullanılmıştır. Firewall donanımı ise AMD Athlon 3000+ CPU, 1Gb Ram ve iki adet anakarta dahil Gigabit ağ arabirimine sahip bir makinedir. Test ortamını kontrol etmek için ise bir adet dizüstü bilgisayar kullanılmıştır. Tüm sistemler gigabit ağ anahtarına CAT 6 kablo ile bağlanmıştır.

İstemci ve sunucular üzerinde standart olarak gigabit ağ kartı olmadığı için fazladan 10/100/1000 Mbit çalışabilen rtl8169 chipsetli ağ arabirim kartları PCI 32bit 33MHz yuvalara takılmıştır. Saldırı için kullanılan sistemde 10/100/1000 Mbit çalışabilen rtl8169 chipsetli ağ arabirimi anakarta dahildir. Firewall sisteminde ise iki adet 10/100/1000 Mbit hızlarda çalışabilen nvidia nForce 3 ve rtl8169 chipsetli anakarta dahil ağ arabirim kartı bulunmaktadır.




Şekil 1. Ağ Ortamı

### 2.2 İşletim Sistemleri

Tüm bilgisayar sistemlerinde GNU/Linux Ubuntu Server 7.10 sürümü işletim sistemi yüklenmiştir. Sistemler üzerinde çalışan GNU/Linux kernel sürümü işletim sistemi ile gelen 2.6.22'dir. Kontrol için kullanılan dizüstü bilgisayarda GNU/Linux Ubuntu Desktop 7.04 sürümü yüklüdür. Firewall makinesinde yine GNU/Linux Ubuntu Server 7.10/kernel 2.6.22 ve OpenBSD işletim sisteminin kararlı sürümü olan 4.2 kurulmuştur.

İşletim sistemlerinde EK I'de verilen ince ayarlar yapılmıştır. İşletim sistemleri kaynak kodlarında hiçbir değişiklik yapılmamıştır.

### 2.3 Ölçüm Araçları

Ölçüm için Netperf 2.4.3 sürümü kullanılmıştır [15]. Netperf istemci-sunucu mimarisinde çalışan bir yazılımdır. İstemci tarafında netperf ve sunucu tarafında çalışan netserver uygulamalarından oluşur. Sunucu olarak kullanılan bilgisayarlarda netserver uygulaması çalıştırılır. İstemci sistemde netperf programına parametre olarak netserver uygulamasının çalıştığı ip numarası ve diğer test parametreleri verilir. Netperf uygulaması ilk olarak sunucu ile TCP protokolü ile kontrol bağlantısı yaparak parametre ve sonuçların sistemler arasında geçişini sağlar.

Kontrol bağlantısı yapıldıktan sonra farklı bir veri bağlantısı açılarak ölçüm işlemi bu bağlantı üzerinden yapılır. Ölçüm bittikten sonra sonuçlar sunucu tarafından istemciye kontrol bağlantısı aracılığıyla aktarılır ve kullanıcıya gösterilir.

Saldırı sisteminde ise GNU/Linux çekirdeği içerisinde çalışan, yüksek yoğunlukta paket üretebilen pktgen uygulaması kullanılmıştır [16]. Bu uygulama milyonlarca farklı özelliklerde paketi saniyeler içerisinde üretebilmektedir. Özellikle ağ arabirim kartlarının paket gönderme performansını ölçmek için kullanılmaktadır.

### 2.4 Firewall Kurallarının oluşturulması

Testler sırasında firewall sistemleri üzerinde ilgili istemci ve sunucuları ilgilendiren kurallar değerlendirmenin en sonunda olacak şekilde değişken kural kümeleri yaratılmıştır. Bu kural kümelerinin herbiri 1, 10, 50, 100, 200, 500 ve 1000 kural içermektedir. PF kural uygulayıcısı kuralları yüklerken özel bir optimizasyon kullandığı için bu özellik kapatılarak netfilter ile ayni koşullarda kurallara sahip olabilmesi sağlanmıştır [11].

### 2.5 Test Sonuçlarının Toplanması

Performans değerlendirmelerinin doğru olarak yapılabilmesi için, teste katılan tüm sistemler üzerine bir betik yüklenmiştir. Bu betik kontrol bilgisayarı üzerinde çalışan Clusterssh [17] uygulaması yardımı ile teste katılan tüm sistemler üzerinde aynı anda çalıştırılmıştır. Betik her bir bilgisayar üzerinde sistemin türüne göre farklı

şekilde çalışmaktadır. Tüm sistemlerde testen önce ve testen sonra olmak üzere ağ arabirim, kesme ve varsa netperf uygulaması sonuçları alınarak kaydedilmiştir. Firewall sisteminde ise ayrıca filtreleme istatistikleri alınmıştır. Alınan bu sonuçlar değerlendirilmek üzere kontrol bilgisayarına aktarılmıştır.

### 2.6 Test Sonuçlarının Anlamlı Hale Dönüştürülmesi

Değerlendirilmek üzere kaydedilen veriler kontrol bilgisayarı üzerinde toplanmıştır. Bu veriler ham halde kaydedilmiş olduklarından, bu verilerin daha anlamlı bir şekle dönüştürülmesi için betikler yazılmıştır. Bu betikler yardımı ile oluşturulan veriler Gnuplot [18] uygulaması ile grafik şekline dönüştürülmüştür.

## 3. ÖLÇÜMLERİN YAPILMASI

Yazılım tabanlı Firewall sistemlerinin performansını ölçebilmek için IETF'nin önerdiği Firewall ölçüm yöntemlerinden yararlanılmıştır [19].

Taşıma katmanındaki UDP [20] ve TCP [21] protokollerinin performansa etkilerini öğrenebilmek için netperf uygulamasının TCP_STREAM ve UDP_STREAM testleri kullanılmıştır [15]. Bu iki testin ortak özelliği, istemci tarafından sunuculara tek yönlü UDP yada TCP trafiği oluşturulabilmesidir. TCP performansını ölçmek için ağ istatistiklerinden, UDP performansını ölçmek için ise raporlanan sunucuya ulaşabilen paket sayısı değerleri kullanılmıştır.

Ölçümler üç farklı şekilde yapılmıştır:

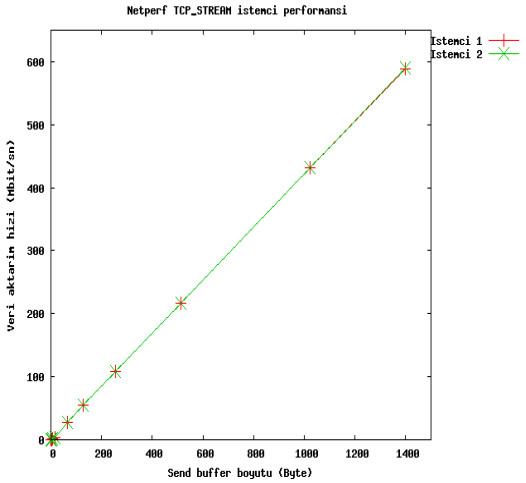
- Doğrudan bağlantı,
- Firewall ve kural sayısı değişken olarak,
- Firewall sistemi saldırı altındayken ve kural sayısı değişken olarak.

Bu ölçümler tek tek ele alınırsa;

### 3.1 Doğrudan Bağlantı

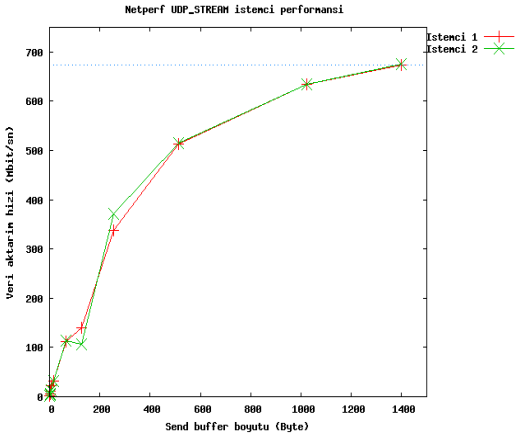
Firewall performansının tam olarak ölçülebilmesi için istemci ve sunucular arasında oluşturulabilecek en fazla trafik miktarının bilinmesi gerekmektedir. Bu işlem istemci ve sunucuların gigabit ağ anahtarı üzerinden birbirlerine bağlanması ile yerine getirilmiştir. Testler, değişken paket boyutlarında herbiri birer dakika olmak üzere hem TCP hemde UDP protokolü için yapılmıştır.

Şekil 2'de istemci ve sunucular arasında doğrudan bağlantıda TCP protokolü kullanılarak test yapılmıştır. Veri aktarım hızı, paket gönderim boyutunun artması ile doğru orantılı olarak artarak en fazla 590Mbit/sn çıkmıştır. Her iki istemcide'de yaklaşık olarak ayni sonuçlar elde edilmiştir.




Şekil 2. Doğrudan bağlantıda istemcilerin TCP_STREAM testi değerleri

Şekil 3'de istemci ve sunucular arasında doğrudan bağlantıda UDP protokolü kullanılarak test yapılmıştır. Veri aktarım hızı, paket gönderim boyutunun artması ile eğri halinde artarak en fazla 670 Mbit/sn kadar çıkmıştır. Her iki istemcide'de yaklaşık olarak benzer sonuçlar elde edilmiştir.




Şekil 3. Doğrudan bağlantıda istemcilerin UDP_STREAM testi değerleri

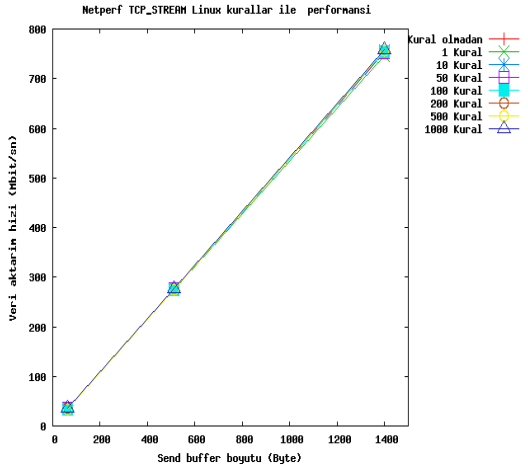
Her iki farklı protokol için yapılan testlerde gigabit seviyelere erişilememesinin nedeni istemci ve sunucular üzerindeki ağ arabirim kartlarının yerleştirildikleri veri yollarının sınırlamalarından kaynaklanmaktadır.

### 3.2 Firewall Kullanılarak ve Kural Sayısı Değiştirilerek Yapılan Test

Bu aşamada her iki AKK yazılım tabanlı Firewall farklı kural sayıları ve farklı protokoller kullanılarak test edilmişlerdir. Verilen sonuçlar teste tabi olan sistemlerin oluşturabildiği toplam trafik miktarlarıdır.

**Linux netfilter ile yapılan testler;**
Şekil 4'de Linux işletim sistemi üzerinde çalışan netfilter ile farklı kural kümeleri ve 64, 512 ve 1400 Byte'lık farklı paket gönderim boyutları ile yapılan TCP_STREAM testlerinin sonuçları verilmiştir.

Açıkça görülebileceği üzere Netfilter'in TCP protokolü için kuralsız ve farklı kural kümeleri ile elde ettiği sonuçlar bire bir örtüşmektedir. Burda netfilterin statefull özelliğininde performansa katkısı olmuştur. Ancak en fazla 760Mbit/sec veri aktarım hızına ulaşılmıştır.



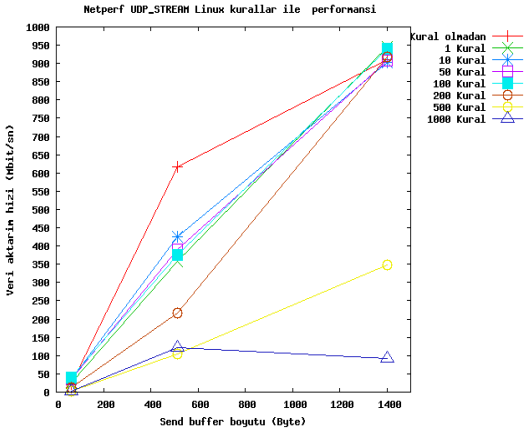

Şekil 4. Linux netfilter ve farklı sayıda kural ile TCP_STREAM testi değerleri

Şekil 5'de Linux netfilter ile farklı kural kümeleri ve 64, 512 ve 1400 Byte'lık farklı paket gönderim boyutları ile yapılan UDP_STREAM testlerinin sonuçları verilmiştir.

UDP protokol testlerinde TCP göre daha fazla veri gönderilebilmektedir. Bunun nedeni UDP protokolünün doğasından kaynaklanmaktadır. UDP protokolünde doğrulama, akış yada sıralama denetimi yapılmamaktadır. Bu yüzden TCP protokolüne göre çok daha fazla veri gönderilebilmektedir.

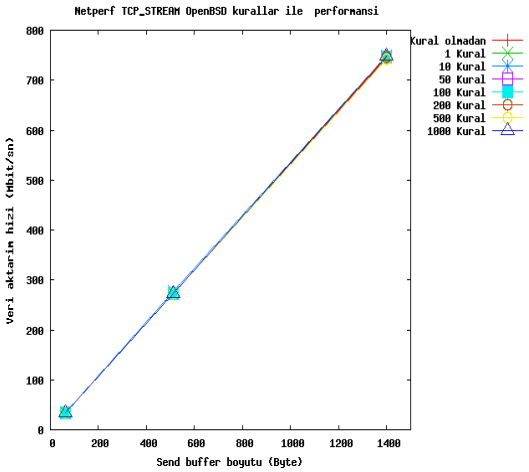
UDP_STREAM testlerinde kuralların performans üzerinde negatif etkisi görülebilmektedir. 200 kurala kadar 1400Byte'lık paketlerde 980Mbit/sn veri aktarım hızına ulaşılmıştır. Ancak 500 ve 1000 kuralda performansın 100Mbit/sn altına düştüğü gözlenmiştir.

Bunun kaynağı UDP protokolünün statefull olmamasıdır. Bu yüzden netfilterin paketleri tüm kural tabanından geçirmesi gerekmektedir. Kural sayısı arttıkça bu da performansın düşmesine neden olmaktadır.




Şekil 5. Linux netfilter ve farklı sayıda kural ile UDP_STREAM testi değerleri

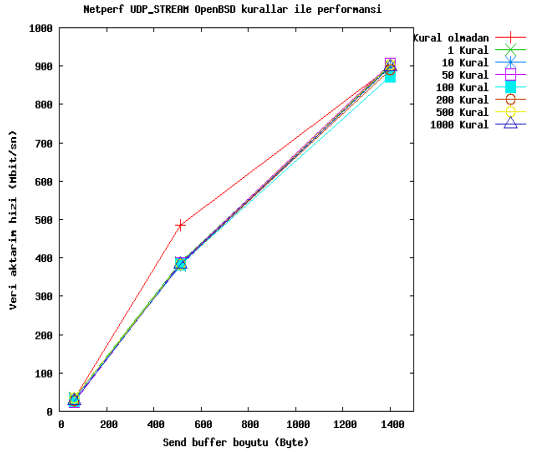
**OpenBSD PF ile yapılan testler;**
Şekil 6'da OpenBSD işletim sistemi üzerinde çalışan PF ile farklı kural kümeleri ve 64, 512 ve 1400 Byte'lık farklı paket gönderim boyutları ile yapılan TCP_STREAM testlerinin sonuçları verilmiştir.




Şekil 6. OpenBSD PF ve farklı sayıda kural ile TCP_STREAM testi değerleri

TCP_STREAM testinde Linux netfilterdeki ile yaklaşık olarak ayni sonuçlar alınmıştır. 1400 Byte paket boyutunda 760Mbit/sn'lik en yüksek hız değerine ulaşılmıştır. Bu testlerde daha fazla performans elde edilememesinin nedeni, TCP protokolünün UDP'ye göre daha fazla işlem gerektirmesidir.

Şekil 7'de OpenBSD işletim sistemi üzerinde çalışan PF ile farklı kural kümeleri ve 64, 512 ve 1400 Byte'lık farklı paket gönderim boyutları ile yapılan UDP_STREAM testlerinin sonuçları verilmiştir.




Şekil 7. OpenBSD PF ve farklı sayıda kural ile UDP_STREAM testi değerleri

OpenBSD üzerinde çalışan PF'in UDP protokolü için de TCP kadar ölçeklenebilir olduğu görülebilmektedir. Kural sayısı artsa bile performanstan ödünç verilmemiştir. Ancak netfilter'in 200 kurala kadar gigabit performansa yaklaştığı gözlemlenmesine rağmen OpenBSD ve PF'de 900Mbit/sn dolaylarında veri aktarım hızı elde edilmiştir. Buna rağmen kural değerlendirme sistemi özellikle 200 kuraldan sonra Linux netfilter'e göre daha etkin çalışmaktadır.

### 3.3 Firewall Sistemi Saldırı Altındayken ve Kural Sayısı Değiştirilerek Yapılan Test

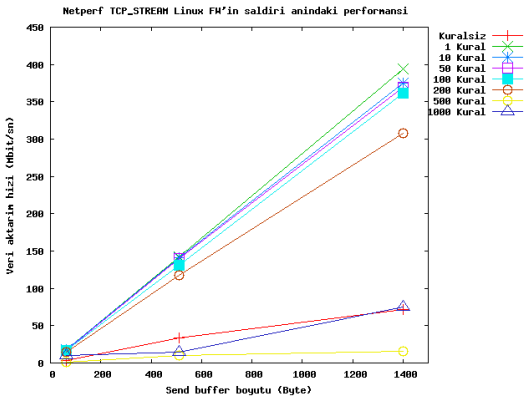
Bu aşamada her iki AKK yazılım tabanlı Firewall Yazılımı farklı kural sayıları ve farklı protokoller kullanılarak saldırı altındayken test edilmişlerdir. Verilen sonuçlar teste tabi olan sistemlerin oluşturabildiği toplam trafik miktarlarıdır.

Saldırı Pktgen [16] yardımı ile yapılmıştır. Saldırı, sabit olarak 95000pps, 60 Byte'lık paketler gönderilerek yapılmıştır. Saldırı makinesinden 46Mbit/sn veri akış trafiği oluşturan 65535 farklı kaynak adresten sunucu olarak çalışan bilgisayarlara doğru yapılmıştır. Üretilen paketler UDP protokolünü kullandığı için, bu testler sırasında sadece TCP_STREAM testleri ile ölçümler yapılmıştır.

**Linux netfilter ile yapılan testler;**

Şekil 8'de Linux işletim sistemi üzerinde çalışan netfilter ile farklı kural kümeleri ve 64, 512 ve 1400 Byte'lık farklı paket gönderim boyutları ile saldırı altındayken yapılan TCP_STREAM testlerinin sonuçları verilmiştir.

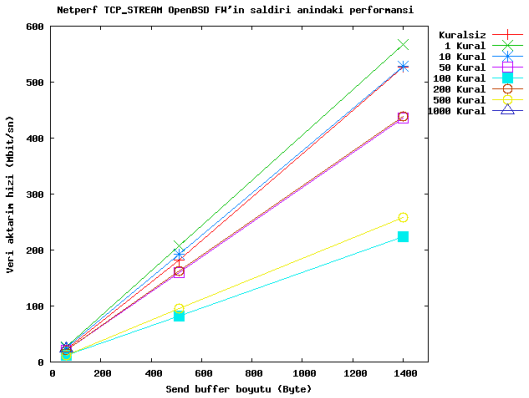
Saldırı altındayken, firewall etkin olmadığı takdirde veri aktarım hızının çok düşük olduğu test sonuçlarında görülebilmektedir. Firewall etkinleştirildikten sonra ise performansın saldırı olmayan testlerde elde edilen sonuçlara göre yarıyarıya azaldığı görülmüştür. Netfilter'in saldırı altındayken sisteme 200 kurala kadar yardımcı olduğu belirlenmiştir. Ancak kural sayısı 500 yada 1000 seviyelerine geldiğinde bunun sistem performansına negatif etki yaptığı gözlemlenmiştir.




Şekil 8. Linux netfilter ve farklı sayıda kural ile saldırı altındayken TCP_STREAM testi değerleri

**OpenBSD PF ile yapılan testler;**

Şekil 9'da OpenBSD işletim sistemi üzerinde çalışan PF ile farklı kural kümeleri ve 64, 512 ve 1400 Byte'lık farklı paket gönderim boyutları ile saldırı altındayken yapılan TCP_STREAM testlerinin sonuçları verilmiştir.




Şekil 9. OpenBSD PF ve farklı sayıda kural ile saldırı altındayken TCP_STREAM testi değerleri

OpenBSD işletim sistemi saldırı altındayken bile PF etkin olmamasına rağmen GNU/Linux'a göre çok daha iyi sonuçlar elde etmiştir. PF etkinleştirildikten sonra veri aktarım performansı bir miktar artmış ancak kurallar eklendikçe gittikçe düşmeye başlamıştır. En dramatik düşüş 500 ve 1000 kuralda yaşanmıştır. Ancak buna rağmen 1'den 10 kurala kadar Linux ve netfilter ikilisinden daha iyi performans elde etmiştir. Ayrıca dramatik düşüş görülen 500 ve 1000 kuralda da yine 200Mbit/sn seviyesinde veri aktarım performansı elde edilmiştir.

## 4. SONUÇLAR

Bu makalede, donanım tabanlı ve pahalı sistemler yerine alternatif oluşturabilecek AKK yazılım tabanlı firewall sistemlerinden en fazla kullanılan iki tanesi ele alınmıştır. Bu iki firewall sisteminin, standart bir PC sınıfı makine üzerinde çalışması halinde bile, gigabit ağlarda kullanılabilecek kadar iyi sonuçlar alabildiği gösterilmiştir.

Her iki firewall sisteminin performansını düşüren en önemli nedenlerden biri, kural sayısının artmasıdır. Benzer özelliklere sahip kurallar guruplar halinde kural tabanına yerleştirilirse, kural değerlendirme işlemi hızlanmaktadır. Bu yöntem sayesinde, firewall sistemlerinin performansı artırılabilmektedir.

Diğer yandan bu çalışmada, PF'in kural sayısı artsa bile kural değerlendirme işleminin netfilter'e göre daha iyi çalıştığı tespit edilmiştir.

Firewall sistemlerininin performansını etkileyen nedenlerden biri de saldırı altında yönlendirebildiği trafik miktarıdır. Çalışmada her iki firewall sistemi de saldırı altındayken test edilmiştir. Her iki firewall sisteminde performansın oldukça düştüğü ancak görevlerine devam ettikleri gözlenmiştir.

Bu çalışmada elde edilen performans sonuçları, daha iyi veri yollarına ve daha iyi ağ arabirim kartlarına sahip bilgisayar sistemleri kullanılarak geliştirilebilir. Burada elde edilen bulgular, düşük maliyetli bir bilgisayarın AKK yazılım tabanlı firewall'lar kullanılması ile pahalı cihazlara bir alternatif oluşturmasını göstermesi açısından oldukça önemlidir.

## KAYNAKLAR


[1] Histories of Internet, http://www.isoc.org/internet/history/brief.shtml#Origins

[2] Internet Usage Statistics, http://www.internetworldstats.com/stats.htm

[3] Bellovin S.M., Cheswick W.R., Nework Firewalls, IEEE Communications Magazine, p 50-57, September 1994

[4] D.L Mills, The Fuzzball, In Proceedings of the SIGCOMM'88 Symposium, pg. 115-122, August 1988

[5] T. Spalink, S. Karlin, L. Peterson, Y. Gottlieb, Bulding a Robust Software-Based Router Using Network Processors, ACM Symposium on Operating Systems Principles, Pg: 216 – 229, 2001

[6] O. Paul, M. Laurent, S.Gombault, C. Duret, H. Guesdon, V. Laspreses, J. Lattman, J. Le Moal, P. Rolin, J-L. Simon, Design and Implementation of a Full Bandwidth ATM Firewall, Computer Security - ESORICS 2006, September 18-20, 2006

[7] Errin W. Fulp and Stephen J. Tarsa, Network firewall policy tries, Technical report, CS Department, Wake Forest Uni., 2004, http://www.cs.wfu.edu/~fulp/Papers/ewftrie.pdf

[8] Lili Qiu, George Varghese, Subhash Suri, Fast firewall implementations for software-based and hardware-based routers, ACM SIGMETRICS Performance Evaluation Review, v.29 n.1, p. 344-345, June 2001

[9] E. Kohler, R. Morris, B. Chen, J. Jannotti, M. F. Kaashoek, The click modular router, ACM Transactions on Computer Systems, August 2000, Volume 18 Issue 3

[10] M. R. Lyu and L. K. Y. Lau, Firewall security: policies, testing and performance evaluation, in Proceedings of the COMSAC. IEEE Computer Society, 2000, pp. 116--21.

[11] M. Adamo, M. Tablo, Linux vs OpenBSD – A firewall performance test, The USENIX Magazine, December 2005, Volume 30, Num 6

[12] Q. Ye, M. H. MacGregor, Click on a Cluster: A Viable Approach to Scale Software-Based Routers, Communications, 2007. ICC '07,24-28 June 2007

[13] H. Welte, The Netfilter framework in Linux 2.4, - Proceedings of Linux Kongress, 2000

[14] D. Hartmeier**,** Design and Performance of the OpenBSD Stateful Packet Filter (pf), Usenix 2002 Procedings

[15] Netperf, http://www.netperf.org/netperf/

[16] R. Olsson, pktgen the linux packet generator, Linux Symposium 2005, http://www.linuxsymposium.org/2005/linuxsymposium_procv2.pdf

[17] Cluster SSH, http://clusterssh.wiki.sourceforge.net/Main+Page

[18] Gnuplot Application, http://www.gnuplot.info/

[19] IETF, Benchmarking Methodology for Firewall Performance, http://www.ietf.org/rfc/rfc3511.txt

[20] IETF, User Datagram Protocol, http://tools.ietf.org/rfc/rfc768.txt

[21] IETF, Transmission Control Protocol, http://tools.ietf.org/rfc/rfc793.txt


## EK I.

GNU/Linux sistemlerde yapılan ayarlar:

```
# Send buffer
net.core.wmem_max=16777216
net.core.wmem_default=16777216

# Receive buffer
net.core.rmem_max=16777216
net.core.rmem_default=16777216

net.ipv4.tcp_tw_recycle=1
net.ipv4.tcp_tw_reuse=1

# TIME_WAIT buckets yükseltildi
net.ipv4.tcp_max_tw_buckets=65535
```

```
# FIN timeout düşürüldü
net.ipv4.tcp_fin_timeout=15

# SYN backlog yükseltildi
net.ipv4.tcp_max_syn_backlog=65536

# Netdev backlog yükseltildi
net.core.netdev_max_backlog=200000

# TCP mem ayarları
net.ipv4.tcp_mem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65535 16777216
net.ipv4.tcp_rmem = 4096 87380 16777216

net.ipv4.tcp_timestamps = 0
```

OpenBSD sistemlerde yapılan ayarlar:

```
# Permit forwarding  of IPv4 packets
net.inet.ip.forwarding=1

# Tunning
net.inet.tcp.recvspace=65536
net.inet.tcp.sendspace=65536

net.inet.udp.recvspace=65536
net.inet.udp.sendspace=65536
```