AĞ ÜZERİNDEN YAVAŞLAMA TABANLI ANOMALİ TESPİTİ

SEÇKİN ANIL ÜNLÜ

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

HAZİRAN 2011

ANKARA

Fen Bilimleri Enstitü onayı

_____

Prof. Dr. Ünver KAYNAK

Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

_____

Doç. Dr. Erdoğan DOĞDU

Anabilim Dalı Başkanı

Seçkin Anıl ÜNLÜ tarafından hazırlanan AĞ ÜZERİNDEN YAVAŞLAMA TABANLI ANOMALİ TESPİTİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

_____

Yrd. Doç. Dr. Hüsrev Taha SENCAR

Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Doç. Dr. Kemal BIÇAKCI _____

Üye : Doç. Dr. Bülent TAVLI _____

Üye : Yrd. Doç. Dr. Tansel ÖZYER _____

Üye : Yrd. Doç. Dr. H. Taha SENCAR _____

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

……..……..

Seçkin Anıl ÜNLÜ

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi

Enstitüsü : Fen Bilimleri

Anabilim Dalı : Bilgisayar Mühendisliği

Tez Danışmanı : Yrd. Doç. Dr. Hüsrev Taha SENCAR

Tez Türü ve Tarihi : Yüksek Lisans – Haziran 2011

Seçkin Anıl ÜNLÜ

AĞ ÜZERİNDEN YAVAŞLAMA TABANLI ANOMALİ TESPİTİ

ÖZET

Bu çalışmamızda, bilgisayarlarda meydana gelen bazı anomalilerin neden olduğu yavaşlamayı, bilgisayar dışından ve edilgen olarak tespit etmeyi amaçlayan bir yaklaşımı test ediyoruz. Bu yaklaşımın farklı olduğu nokta ve önemli özelliği, bilgisayar üzerinden yapılan ölçümlerle değil, tamamen dışarıdan, ağ etkileşimlerinin esas alınmasıyla çalışıyor olmasıdır. Burada önerilen yaklaşım, bilgisayarın bazı ağ etkileşimlerinde sunucuya yanıt verme sürelerinin dışarıdaki bir gözlemci bilgisayar üzerinden ölçülmesi ve bu verinin çözümlenmesiyle yavaşlamanın tespit edilmesidir. Bu amaçla bazı istatistiksel testler ve bunlara dayalı algoritmalar denenmiş ve ne düzeyde başarılı oldukları incelenmiştir.

Bilgisayarın yavaşlamasına sebep olan anomaliler, donanımsal ve yazılımsal değişiklikler, hatalar ve sorunlar, kötü amaçlı yazılımların bulaşması olarak tanımlanmıştır. Bu etkenlerin sebep olduğu etkilerin tespit edilebilmesi için bu etmenleri içeren farklı sistem yapılandırmaları üzerinde testler yapılmış ve bunlardan alınan verilerle yöntem incelenmiştir.

Verilerin toplanabilmesi için bir test ortamı oluşturulmuş ve belirli etkinlikleri içeren uygulamalar istemci makine üzerinde çalıştırılarak, istemcinin ağ trafiği gözlemci bir makine üzerinden toplanmıştır. Bu veri içerisinden bazı protokoller için yanıt süreleri hesaplanmış ve yapılan istatistiksel testler sonucunda yavaşlama miktarının dışarıdan ölçülebilir ve temel durumdan ayırt edilebilir boyutta olduğu görülmüştür. Sonuçlar, önerilen yaklaşımın kullanılmaya uygun olduğunu göstermektedir.

Anahtar Kelimeler: Ağ, Ağ trafiği, Anomali, Değişim tespiti, Yavaşlama, Bilgisayar, Yazılım, İşletim sistemi, Donanım, Güvenlik

University : **TOBB University of Economics and Technology**

Institute : **Institute of Natural and Applied Sciences**

Science Programme : **Computer Engineering**

Supervisor : **Asst. Prof. Hüsrev Taha SENCAR**

Degree Awarded and Date : **M. Sc. – June 2011**

Seçkin Anıl ÜNLÜ

ANOMALY DETECTION OVER NETWORK BASED ON SLOWDOWN DATA

ABSTRACT

In this work, we test an approach for passively detecting the slowdown which arises from some anomalies on computers over the network. This novel approach is important and differs from some other methodologies because it works without the measurements done on the computer itself, but it is based on the observation of client's network interactions out of the client. The proposed approach relies on detecting the slowdown via measurement of the elapsed time while client responds to the server in some network protocols. For this purpose, we tested and examined some statistical tests and algorithms based on these.

Anomalies which cause the slowdown of computers are defined as hardware and software changes, errors and problems, malicious software infection. In order to detect the effects of these factors, we tested the methodology over the data we collected over different systems which include these activities.

We have designed different test environments to collect data over client running some applications which include specific activities to be tested. In this setup, the client network traffic is captured over the network by an observer machine. We have calculated the response times and have seen that the slowdown amount is sufficient for detection of anomalies in reference to the ground state. Results have shown that the approach is suitable for measurement of the slowdown.

Key Words: Network, Network packet traffic analysis, Anomaly, Change detection, Slowdown, Computer, Software, Operating system, Hardware, Security

TEŞEKKÜR

Çalışmalarım boyunca yardım ve katkılarıyla bana destek olan ve beni yönlendiren değerli hocam ve tez danışmanım Yrd. Doç. Dr. Hüsrev Taha SENCAR'a, deneyimlerinden yararlandığım TOBB Ekonomi ve Teknoloji Üniversitesi Bilgisayar Mühendisliği Bölümü öğretim üyelerine, gereken durumlarda yardımlarını esirgemeyen bölüm arkadaşlarıma, bana verdikleri her türlü destek nedeniyle ailem ve arkadaşlarıma teşekkürü borç bilirim.

İÇİNDEKİLER

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

KISALTMALAR

Kısaltmalar	Açıklama
ACK	TCP acknowledgement flag
ASL	Achievable Significance Level
AV	Anti-Virus
CHAP	Challenge-Handshake Authentication Protocol
CPU	Central Processing Unit
DNS	Domain Name System
FTP	File Transfer Protocol
FW	Firewall
G/Ç	Giriş/Çıkış
HD	High Definition
I/O	Input/Output
IMAP	Internet Message Access Protocol
IP	Internet Protocol
ISO	ISO 9660 file system
KL	Kullback-Leibler
KS	Kolmogorov-Smirnov
MİB	Merkezi İşlemci Birimi
NAT	Network Address Translation
OS	Operating System
PF	Persistence Factor
POP3	Post Office Protocol v3
RAM	Random Access Memory
SP1/3	Service Pack 1/3
SRP	Secure Remote Password Protocol
SSL	Secure Socket Layer
SYN	TCP synchronize sequence numbers flag
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
USB	Universal Serial Bus
WiFi	Wireless Fidelity

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklamalar**
$\hat{d}_i$	i. önyükleme tahmini
d_{hi}	KL-uzaklığı testinde kritik bölgenin başlangıç sınırı
d_t	İki pencere arasındaki KL-uzaklığı
$F(x)$	X örneklemi için deneysel olasılık dağılımı
k	Önyükleme örneği sayısı
$Pr(x)$	x'in gerçekleşme olasılığı
P_w	W penceresinden oluşturulan deneysel olasılık dağılımı
R^d	d boyutlu gerçek sayı uzayı üzerinde bir nokta
W_A	A'dan yapılan rastgele ölçümlerin sıra toplamlarını gösteren rastgele değişken
W_i	x_i ile biten n adet akış değerini içeren pencere
α	Güven aralığı, achievable significance level (ASL)
δ	kdq-ağacı için bir hücrenin en büyük kenar boyutu
τ	kdq-ağacı için bir hücredeki en fazla nokta sayısı
γ	KL-testi sürerlik katsayısı (persistence factor)

1. GİRİŞ

Bu çalışmamızda, bir ağda bulunan bilgisayarların veri trafiğinin incelenmesiyle, bu bilgisayarlardan bazılarında görülen yavaşlamadan yola çıkılarak, meydana gelen anomalilerin tespit edilmesi amaçlanmıştır. Buradaki temel yaklaşım, bilgisayarlardaki yavaşlamanın, ağ üzerinden, yani bilgisayar dışından ve edilgen olarak istemci bilgisayarların bazı protokollerdeki yanıt verme sürelerinin ölçülmesiyle tespit edilmesidir [1]. Bu yöntem, yavaşlama tespitinin ağ üzerinden yapılması sayesinde bilgisayar sisteminden bağımsız olarak merkezi bir noktadan kullanılabilmektedir. Tespit işleminin bilgisayar dışından yapılması sayesinde de sistemde bulunabilecek kötü amaçlı yazılımların faaliyetinden de etkilenmemektedir.

Bir bilgisayarın yavaşlamasına neden olabilecek etkenler arasında donanımsal ve yazılımsal değişiklikler, arızalar, yanlış yapılandırmalar, hatalı aygıtlar ve sürücüler, özellikle üst düzeydeki türleri olmak üzere kötü amaçlı yazılımlar sayılabilir. Bilgisayardaki donanım bileşenlerinin arızalanması veya değiştirilmesi, özellikle işletim sistemi ve anti-virüs/güvenlik duvarı gibi sistem seviyesindeki yazılımların değiştirilmesi veya yüklenmesi bilgisayar başarımını doğrudan etkilemektedir. Genellikle kendilerini ve sistemde yaptıkları değişiklikleri gizlemek amacıyla işletim sistemi çekirdeği üzerinde belirli işlevleri yamalayarak çalışan rootkit ve benzeri üst düzey kötü amaçlı yazılımlar, sistem kaynaklarını kendi amaçları çerçevesinde tüketerek sistem başarımında önemli ölçüde azalmaya neden olmaktadır. Özellikle bu tür yazılımları, bulaştığı bilgisayarın içerisinden, bilinen kod imzalarıyla eşleyerek bulmak gibi geleneksel yöntemlerle tespit etmek, bu yazılımların aldığı karşı-önlemler nedeniyle çok zor olmakta ve bazen mümkün olmamaktadır. Bu nedenle tespit işleminin, dışarıdan, sürekli olarak, pratik ve etkin bir biçimde yapılması önemli bir gereksinimdir [1].

Çalışmamızda test ettiğimiz bu yaklaşımın bilgisayar üzerinden ve dışından uygulanmakta olan diğer yöntemlerle beraber kullanılmasıyla anomalilerin meydana geldiği makinelerin tespitini kolaylaştıracağını düşünüyoruz.

1.1. Sorunun Tanımlanması ve Anomaliler

Bilgisayarların başarımında kalıcı veya uzun süreli bir azalmaya neden olabilecek değişikleri "anomali" olarak tanımlıyoruz. Bu anomalilerin saptanması için var olan ve üzerinde çalışılmakta olan çeşitli yöntemler mevcuttur. Tezimizde test ettiğimiz yaklaşım, bu sorunu tespit etmek için kullanılan diğer yöntemlerin etkinliğini arttırabilecek olan ek bilgiyi sunmaktadır. Bu yaklaşım, anomalilerin sebep olduğu yavaşlamanın, bir ağdaki bilgisayarların bazı protokollerde sunucuya yanıt verme sürelerinin gözlenmesiyle, ağ üzerinden edilgen bir biçimde tespit edilmesini amaçlanmaktadır. Bu bilgi, sistem yöneticileri tarafından özellikle bu bilgisayarların incelenmesi için kullanılabilir.

Anomali olarak tanımladığımız durumlar aşağıdaki şekillerde oluşabilir:

- Donanımsal değişiklikler ve arızalar
- Yazılımsal değişiklikler ve arızalar, aygıt yazılımları ve sürücü sorunları / değişiklikleri
- Kötü amaçlı yazılımların bulaşması

Bu anomalilere örnek olarak şunları verebiliriz: Arızalı bir ağ kartı, ağ paketlerine verilecek yanıt süresini uzatabilir, aynı şekilde bu ağ kartının daha hızlı bir kartla değiştirilmesi veya Ethernet kartı yerine kablosuz ağ kartının kullanılması da yanıt süresini etkiler. Tüm bilgisayar bileşenleri, MİB ve bellek üzerinden yönetildiği için sabit disk, ekran kartı ve diğer G/Ç kartları da sistem başarımını doğrudan etkilemektedir. İşletim sisteminin değiştirilmesi, bu sisteme anti-virüs ve güvenlik duvarı gibi sistem seviyesinde çalışan uygulamaların kurulması gibi değişiklikler de bilgisayar başarımını etkilemektedir. Bu etkilerin nedenlerine baktığımızda görece yüksek ve sürekli MİB ve G/Ç kaynağı kullanımı olduğunu görmekteyiz.

Bugün bilgisayarlar ve ağ sistemleri değişik kaynaklardan gelen saldırılara ve kötü yazılımların bulaşmasına oldukça açık durumdadır [2]. Özellikle belirli hacker gruplarının birçok bilgisayarın denetimini ellerine alarak oluşturduğu botnetler, bunların oluşturulabilmesi için bu grupların yaydıkları virüs, solucan, arkakapı ve rootkit gibi üst düzey kötü amaçlı yazılımlar, ciddi ekonomik zararlara ve kayıplara

neden olmaktadır. Bu kötü amaçlı yazılımlar, amacını gerçekleştirebilmek için yaptıkları yamalarla sistem kaynaklarını yüksek düzeyde kullanmakta ve bilgisayarların yavaşlamasına sebep olmaktadır. Gittikçe gelişen ve yeni ortaya çıkan saldırı tekniklerini kullanarak çalışan rootkitler, anti-virüs, güvenlik duvarları, sızıntı tespit ve önleme sistemleri gibi araçlara ve alınan diğer önlemlere karşı kendilerini korumaya ve gizlemeye çalışmakta, bunun için uyguladıkları teknikler de genellikle sistemlerin önemli ölçüde yavaşlamasına neden olmaktadır [3,4]. Bu teknikler nedeniyle, kötü amaçlı yazılım dosyalarının imzalarını ve belirtilerini tarayarak çalışan anti-virüs yazılımları, bu düzeydeki zararlı yazılım bulaşmalarının tamamını, özellikle sistem çalışmaktayken, yüksek doğrulukta tespit edememektedir [5]. Bunlara karşı geliştirilen farklı tespit yöntemleri mevcut olsa da bu yöntemler, uygulaması zor olan ve pratik olmayan yöntemlerdir [6]. Etkili olan bazı yöntemler, sistemin temiz bir kopyasının karşılaştırmada referans olarak elimizde bulunmasını ve teker teker bunun üzerinden dosya denetimi yapılmasını gerektirir. Sürekli olarak temiz bir kopyanın tutulması, uzun süren karşılaştırma işlemlerinin periyodik olarak tekrarlanması gibi gereklilikler bu sistemleri pratiklikten uzaklaştırmaktadır. Diğer bir yöntem ise sistemin güncel kopyasının sürekli tutulmasını gerektirmemekte, açık ve kapalı durumları arasında kıyaslama yaparak zararlı yazılımları tespit etmeye çalışmaktadır [7]. Bugün genellikle çoğu rootkit tespit aracı gizli dosya, dizin ve işlemleri aramakta, kullanıcı modu ile çekirdek modu üzerinden alınan bu bilgiler karşılaştırmakta ve çekirdek üzerindeki bazı önemli noktalardaki değişiklikleri tespit etmeye çalışmaktadır. Bunun için belirli rootkit türlerini hedef olarak seçmekte ve buna göre oluşturulan listeler üzerinden çalışmaktadır. Fakat zararlı yazılım geliştiricileri de bu yöntemlerden haberdardır ve sürekli yeni teknikler geliştirerek ve ürünlerdeki gelişmeleri takip ederek bunlara karşı özel önlemler almakta ve bu teknikleri etkisiz hale getirebilmektedir. Özellikle bu tür gelişmiş zararlı yazılımların bulaşmış olduğu bir sistem üzerinden, bunları güvenilir biçimde tespit edebilecek, oturmuş bir yöntem mevcut değildir. Bu nedenle bir bilgisayara bulaşmış olan bu üst düzey zararlı yazılımların tespit edilmesi açık ve önemli bir sorun olarak ortada durmaktadır. Çalışmamızda kullandığımız ve test ettiğimiz yaklaşım, geleneksel yöntemlerle bilgisayar içerisinden güvenilir bir şekilde tespit edilemeyen bu üst

düzey zararlı yazılımların ağ üzerinden tespit edilmesine yardımcı olacak bir yöntemi sunmaktadır.

Bu yaklaşımı test edilebilmek amacıyla, bilgisayarların yavaşlamasına sebep olabilecek etkenlerin, kurduğumuz ağ yapılandırması altında gözlemlenmesi ve buradan aldığımız verilerin istatistiksel yöntemler kullanılarak incelenmesi amaçlanmıştır. Kurulan test ortamlarından elde edilen yanıt süreleri üzerinden değişim noktalarının tespit edilmesi için farklı istatistiksel testler ve bunlara dayalı algoritmalar denenmiş ve bunların ne kadar başarılı olabildiği gözlemlenmiştir.

2. YÖNTEM

Günümüzde İnternetin yakın zamandaki hızlı gelişimiyle birlikte ağa sürekli bağlı bilgisayar sayısı ciddi şekilde artmış durumdadır ve artık bilgisayar kullanıcılarının büyük çoğunluğu gün içerisinde önemli ölçüde İnternetten bilgi alışverişi yapmaktadır. İnternetin altyapısını oluşturan TCP/IP protokol kümesi yaygın olarak kullanılmaktadır. Bu protokoller içerisinde yer alan ve karşılıklı etkileşim içeren aşamalar önerilen yöntemin test edilmesi için uygun bulunmuştur. Aşağıdaki protokol adımları sistemin test edilmesi için uygundur:

- soru-yanıt yöntemleri (question-response)
- sorgu-yanıt yöntemleri (challenge-response)
- el sıkışma/görüşme yöntemleri (handshaking/negotiation)

Bu etkileşimler aynı protokol içerisinde birden fazla aşama şeklinde de gerçekleşebilir. Bu yöntemleri içeren ve sistemin test edilmesine uygun olabilecek protokollere örnek olarak TCP 3-yönlü el sıkışma, DNS sorgulama, çeşitli sorgu-yanıt kimlik doğrulama protokolleri (POP3 ve IMAP içerisinde yer alan doğrulama yöntemleri, CHAP, SRP, Kerberos gibi), SSL/TSL gösterilebilir. Bu çalışmamızda TCP 3-yönlü el sıkışma ve DNS sorgulamaları gibi sık kullanılan ve toplanan veri içerisinde en çok rastlayabileceğimiz etkileşimleri seçtik.

2.1. TCP 3-Yönlü El Sıkışma Yanıt Süresi Ölçümü

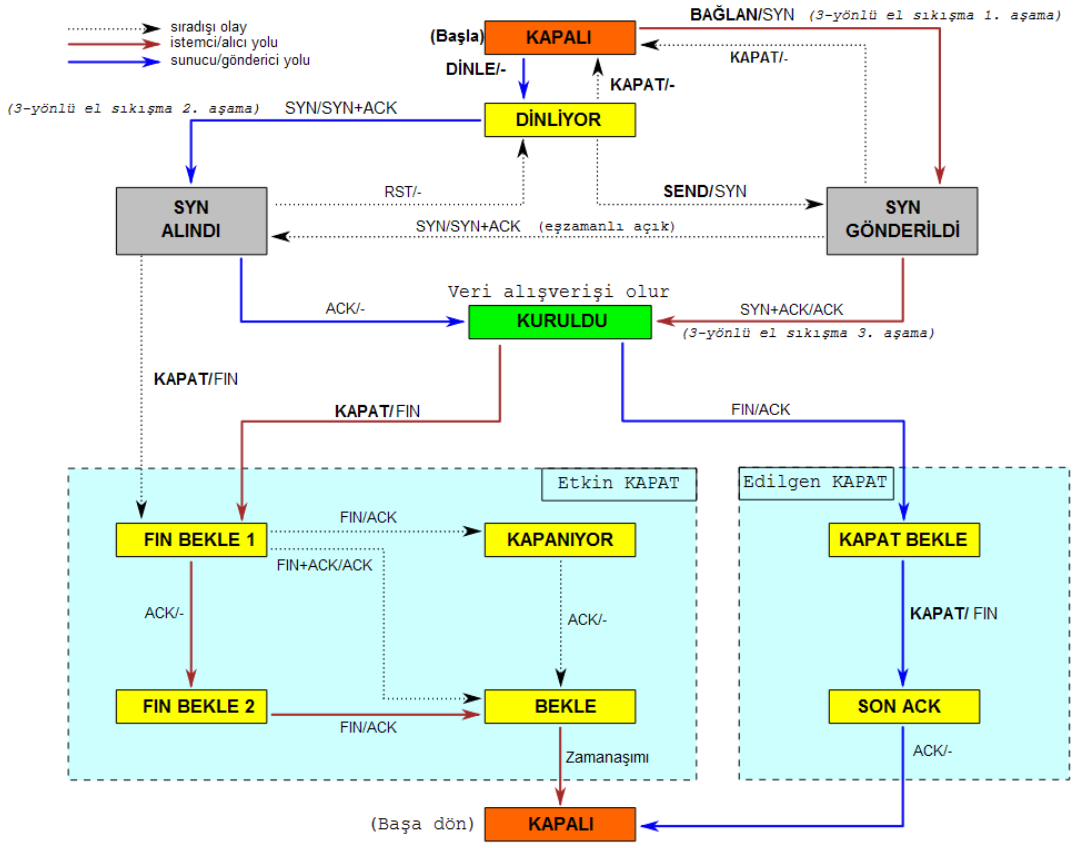
Ağ trafiğinin büyük çoğunluğunu oluşturan TCP protokolü içerisinde karşılıklı olarak bağlantıların kurulabilmesi için 3-yönlü el sıkışma (3-way handshake) adını verdiğimiz bir aşama gerçekleşmektedir. Bu aşamada A ve B bilgisayarları arasında iletişim sırasıyla aşağıdaki şekilde kurulur:

1. İlk önce A bilgisayarı B bilgisayarına SYN bayrağı açılmış bir TCP paketi gönderir.
2. B bilgisayarı bu paketi aldığını onaylamak için A'ya SYN ve ACK bayrakları açık bir paketle yanıt verir.
3. A bilgisayarı B'ye ACK bayrağı açık bir TCP paketi yollar.
4. B bilgisayarının bu mesajı almasıyla birlikte bağlantı kurulmuş olur.

Bit başı	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Kaynak portu																Hedef portu															
32	Sıra numarası (Sequence)																															
64	Onay numarası (Acknowledgment) (eğer ACK açıksa)																															
96	Veri başı				Rezerve			CWR	ECE	URG	ACK	PSH	RST	SYN	FIN		Pencere boyutu															
128	Denetleme (Checksum)																Acil işaretçisi (Urgent) (eğer URG açıksa)															
160 ...	Seçenekler (eğer Veri başı > 5 ise) ...																								dolgu							

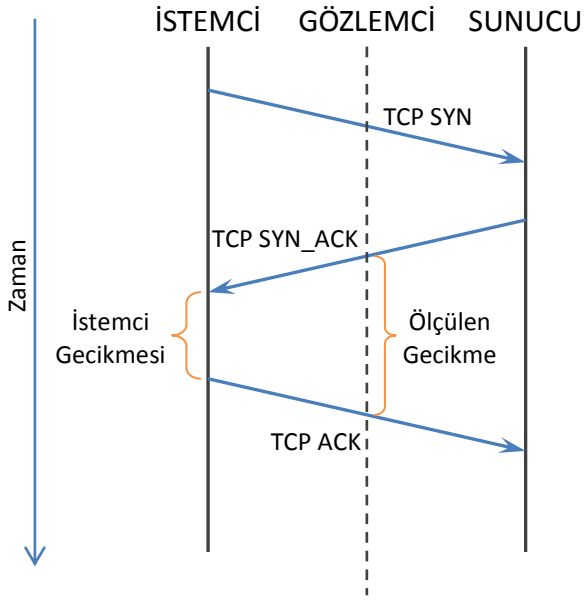
Şekil 2.1. TCP Paket Başlığı

Her iki makine de gönderdikleri ilk pakette sıra numaralarını rastgele seçer, daha sonraki paketlerde bu sıra numaralarından devam ederek haberleşir. TCP paket başlığı Şekil 2.1'de verilmiştir. SYN bayrağı açıksa o makinenin sıra numarası alanındaki değerin bir fazlası olacak şekilde karşı tarafın SYN+ACK paketindeki onay numarası ayarlanır ve iletişim bu şekilde devam eder. Bu iletişime ait durum şeması Şekil 2.2'de gösterilmektedir.



Şekil 2.2. TCP Durum Şeması

Ölçmek istediğimiz değer, makinenin kendisine SYN+ACK paketi gelmesi ile buna karşılık onay olarak verdiği ACK paketini göndermesi arasında geçen süredir. Bu süre genel makine başarımının değişimi için iyi bir göstergedir. Farklı ölçüm sistemi yapılanmalarına göre bu sürenin üzerine ölçümü yapan bilgisayara olan ağdaki erişim süresi hem karşı makineden gelişinde hem de makinenin verdiği yanıtta eklenmektedir. Bu sürenin de ağ yoğunluğu, ağ türü (Ethernet, WiFi, vb.) gibi çeşitli durumlardan etkilendiğini göz önüne almamız gerekir (Şekil 2.3).



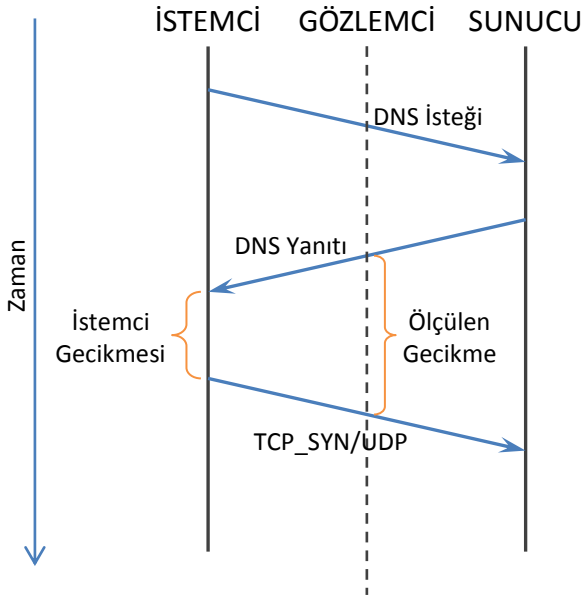
Şekil 2.3. TCP 3-Yönlü El Sıkışma için Ölçülen Gecikme

2.2. DNS Sorgulama ve Sunucuyu Ziyaret Gecikmesi Ölçümü

İnternetteki IP adreslerinin insanlar tarafından akılda tutulması zor olduğundan Alan Adı Sistemi (Domain Name System, DNS) adını verdiğimiz bir IP-alan adı eşleşmesi çözümleyici bir sistem oluşturulmuştur. İnternet üzerinde veya yerel ağda bulunan DNS sunucuları, İnternet adreslerinin IP adresi karşılıkları saklamakta ve istemcilerin isim sorgularına yanıt vermektedir. İnternet üzerinde veri alışverişi için kullanılan tüm alan adlarının IP adresine dönüştürülmesi gerektiğinden DNS protokolü de sık kullanılan protokollerden biridir. Bir A istemcisi bir B İnternet sunucusu adresine bağlantı kurmadan önce Şekil 2.4'te gösterilen aşağıdaki işlemler gerçekleşir:

1. Önce A istemci bilgisayarı B sunucusuna bağlanmak için B'nin alan adını ayarlanmış olan C, DNS sunucusundan sorgular.
2. C, DNS sunucusundan bu sorguya karşılık yanıt gelir ve bu yanıtta B'nin IP adresi bulunur.
3. A istemcisi B sunucusuna kendisine verilen IP adresi ile bağlantı açmaya başlar. Bu bağlantı TCP veya UDP protokolünde olabilir.

DNS sunucuları genellikle önceden sabit olarak ayarlanmakta veya DHCP protokolü üzerinden istemcilere dağıtılmaktadır. Bir sunucunun yanıt verememesi olasılığına karşılık genellikle ikinci veya üçüncü sunucular da burada yer alır.



Şekil 2.4. DNS Yanıtı ve Ardından Yeni Bağlantı için Ölçülen Gecikme

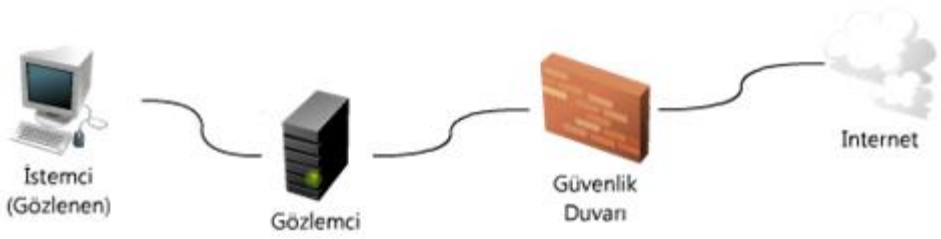
Genellikle bir alan adı için çözümlenen IP adresleri sık değişmediğinden bu çözümlemeler işletim sistemi DNS hizmeti veya İnternet uygulamalarının kendi DNS önbelleklerinde geçici bir süreyle saklanır. Böylece bu çözümlemelerin sık tekrar edilmesi engellenir ve başarım artışı sağlanır. Bu nedenle mümkün olduğunca fazla DNS sorgusu yakalamak için uzun bir farklı adres listesi kullanmamız veya DNS önbelleklerini temizlememiz gerekir.

Testlerimiz içerisinde Alexa isimli İnternet sitelerinin kullanım ve ziyaret istatistiklerini takip eden bir kuruluşun yayınladığı "En Çok Ziyaret Edilen 1.000.000 Site" listesinden [8] ilk 4472 adedini içeren bir liste oluşturduk ve istemcinin bu listeyi sırayla gezmesini sağladık.

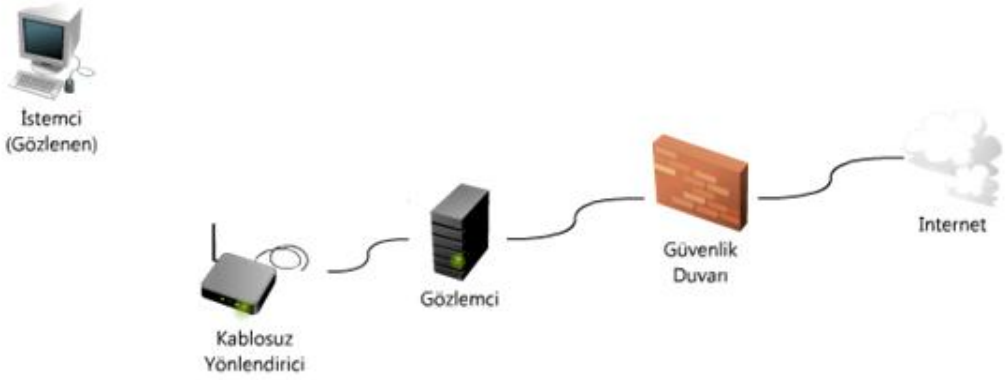
DNS ölçümlerinden elde ettiğimiz gecikme süresi uygulama seviyesine kadar çıkan bir işlemi kapsadığından bilgisayarın kullanıcı modundaki genel başarımının TCP ölçümlerine göre daha iyi bir göstergesidir. Fakat DNS yanıtı ile ardından sunucuya açılan bağlantı işlemi, TCP 3-yönlü el sıkışma gibi tek bir protokolün içerisinde yer alan bir işlem olmadığından ve dolayısıyla farklı uygulamalar, çözümledikleri DNS adreslerine farklı zamanlarda bağlantı kurmayı seçebileceklerinden, yeterince hızlı davranan ve aldığı DNS yanıtı ile sunucuya bağlantı açmaya başladığı nokta arasında keyfi işlemler yapmayan bir uygulama ölçümlerin sağlıklılığı açısından önemlidir.

2.3. Test Ortamı

Belirlediğimiz karşılıklı etkileşim içeren bu protokollere ait ağ trafiği verisini toplamak amacıyla bir test ortamı oluşturduk ve bunun için Şekil 2.5'teki gibi birbirine Ethernet ile bağlanmış olan ve 10'ar Mbit'lik Ethernet kartlarına sahip 3 makine kullanılmıştır. Bu makinelerden İnternete bağlı olan ve diğer makineleri NAT üzerinden İnternete ulaştıran makineye Ubuntu Linux yüklenmiş, iptables güvenlik duvarı etkinleştirilmiş ve NAT hizmeti de bunun üzerinden ayarlanmıştır. Gözlemci makineye de aynı şekilde Ubuntu Linux yüklenmiş ve tcpdump aracı kullanılarak paket trafiği bu makine üzerinden yakalanmıştır. Çekirdek seviyesindeki paket filtreleme özelliklerini kullandığından bu işlemin ölçüme olan etkisi en az boyutta olmaktadır. Ayrıca bu makine sadece gözleme işi için ayrılmış ve üzerinde başka bir ağır işlem çalıştırılmamıştır. İstemci makinesine ise Microsoft® Windows® XP SP1, Microsoft® Windows® XP SP3 ve Microsoft® Windows® 7 Professional (x86) kurularak testler yapılmıştır. Bu bilgisayarlar Intel® Pentium® 4 D 3.0 GHz çift çekirdek 64-bit işlemcilere ve 2 GB belleğe sahiptir. Bilgisayarlar birbirlerine yüksek başarımlı bir switch aracılığıyla bağlanmış ve İnternete çıkan karttaki hariç tüm IP adresleri elle atanmıştır (Şekil 2.7).

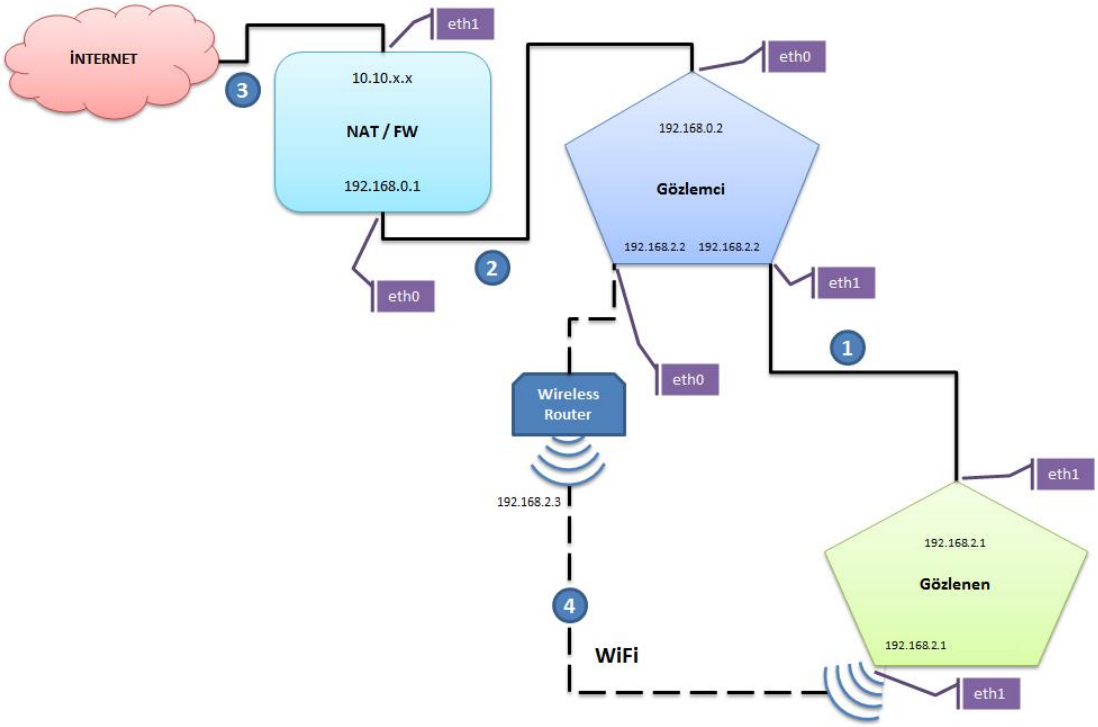


Şekil 2.5. Ethernet ile Birbirine Bağlı Olan Test Ortamı



Şekil 2.6. Kablosuz Ağ Test Ortamı

Kablosuz ortamda yapılacak olan testler içinse Şekil 2.6'da gösterildiği gibi gözlemci makineye Linksys 802.11g 54 Mbps kablosuz yönlendirici bağlanmış, İstemci makineye bir USB kablosuz ağ kartı takılarak gözlemci üzerinden İnternete çıkması sağlanmıştır.



Şekil 2.7. Testler için Kullanılan Ağ Yapısı ve Yapılandırması

İstemci tarafında testte kullanılan 4472 adet adres, sağlıklı sonuç alana kadar bu adresleri sırayla gezecek çeşitli araçlarla denenmiştir. İlk önce denenen "Page Update Watcher" [9] isimli uygulama giderek yükselen gecikme değerleri vermiştir, bunun nedeninin uygulamanın adresleri gezme hızı ayarlanamadığından listede ilerledikçe aldığı DNS yanıtlarına karşılık adresi ziyaret süresinin artması olduğunu düşünüyoruz. Bu araçtan sonra ikinci olarak sistemdeki varsayılan tarayıcı olan Microsoft® Internet Explorer ile listenin gezilebilmesi için Visual Basic Scripting (VBScript) dilinde ufak bir betik [10] yazılmış, bu yöntem de Internet Explorer betik üzerinden istenildiği şekilde kontrol edilemediği ve çeşitli sorunlara yol açtığı için uygun bulunmamıştır. Son olarak Wget Linux indirme yöneticisi [11] aracının Windows sürümü denenmiş ve bu aracın birçok ayarı desteklediği böylece sağlıklı sonuçlar verecek en uygun ayarlar bulunduğunda testler için uygun olacağına karar verilmiş, bir süre bu araçla alınan sonuçlar, değişik ayarlar test edilerek testlerin hızı ve doğru sonuçları verebilmesi açısından eniyileştirilmiştir. En uygun ayarların DNS önbelleği kapalı olacak şekilde 10 sn. bağlantı zaman aşımı süresi, en fazla 1 tekrar deneme ve 3 yönlendirme, listedeki adresler arasında da en fazla 3 sn. bekleme

olduğu görülmüştür. Bu ayarlarla İstemci makine testlerde her test arasında yaklaşık 5'er dk. bekleme süresiyle listeyi tekrar tekrar gezecek şekilde çalıştırılmış, her test öncesinde alınan yedeklerden sistem sıfırlanmıştır. Bir test grubu için liste 4'er kez tekrar gezilmiş, böylece yaklaşık 17-18.000 yanıt süresi örneği elde edilmiştir.

Testlerdeki veriler gözlemci makine üzerindeki eth0 kartından tcpdump aracı "-i eth0 -n -s 256 -w file.pcap 'not (dst 144.122.144.177 or src 144.122.144.177)'" seçenekleriyle [12] çalıştırılarak toplanmış ve PCAP biçiminde kaydedilmiştir. Bu seçenekler ile yakalanan paket boyutu 256 bayt ile sınırlanarak sadece gereksinim duyduğumuz kısmın kaydedilmesi sağlanmış, FTP testinin yapıldığı adresler çıkarılarak buradaki gereksiz yüksek miktarda verinin kaydedilmesi engellenmiştir.

Testlerden alınan PCAP dosyalarının işlenerek TCP ve DNS yöntemleri için gecikme verilerinin çıkarılması amacıyla Java programlama dilinde jNetPcap kütüphanesi [13] kullanılarak bir uygulama yazılmıştır. Bu uygulamayla TCP 3-yönlü el sıkışmaların ve sorgulanan DNS adresiyle bağlantı kurulan adresi birbirine eşleşen paket iletişiminin gecikme değerleri hesaplanmıştır.

3. TESTLER

Yöntemimizin test edilmesi için gözlenen bilgisayar sisteminde aşağıdaki durumlar göz önüne alınmıştır:

- Temel durum
- Tek video oynatımı
- Çift video oynatımı
- FTP'den dosya indirme
- Disk etkinliği (dosya kopyalama)
- Gözlemciden ağ trafiğini almak yerine gözlemciyi atlayarak güvenlik duvarı makinesinden alınan kayıt (1 hop)
- İstemci makinesine güvenlik duvarı kurularak

İstemci makinedeki bu değişikliklerin üreteceği yoğun MİB ve G/Ç kullanımının etkilerinin ölçülmesi amacıyla her durum için veriler toplanmıştır. Her bir durum için testler en az 4'er kez tekrarlanarak daha sağlıklı sonuç elde edilmesi sağlanmıştır. Bu işlemlerin yanında istemci makine üzerinde başka bir işlem çalıştırılmamış, güncellemeler kapatılmış ve her test öncesi sistem yedekten kopyalanarak sıfırlanmıştır.

Bu testlerin detayları ve görmeyi beklediğimiz etkiler şu şekildedir:

- Temel durum diğer testlere referans oluşturmaktadır.
- Video testlerinde KMPlayer [14] uygulaması ile yüksek çözünürlüklü (Full HD) bir video dosyası tekrar tekrar oynatılmış, böylece yaklaşık % 40 civarında MİB kullanımı sağlanmıştır. Bilgisayarlar çift çekirdekli olduğu için çift video testi de uygulanmıştır. Yoğun MİB kullanımının gecikme süresini arttırması beklenmelidir.
- FTP testinde Pardus yerel Linux dağıtımının ISO dosyası sürekli indirilip silinerek yoğun ağ trafiği uygulanmaya çalışılmış ve gecikme değerlerinin bundan nasıl etkileneceğinin gözlenmesi amaçlanmıştır.
- Disk işlemi testinde büyük boyutlu bir dosya sürekli kopyalanarak silinmiş ve yoğun G/Ç etkinliği oluşturulması amaçlanmıştır.

- Gözlemciyi atladıktan sonra güvenlik duvarının bulunduğu makine üzerinden okuduğumuz değerlerde ölçüm makinesinin istemciye olan ağ uzaklığının etkisini ölçmeyi amaçladık. Hem sunucudan istemciye doğrulamanın/yanıtın ulaşması hem de istemciden buna karşı yanıtının görülmesine kadar aradaki ek atlamadan oluşan gecikme test değerine yansıyacaktır.
- İstemciye kurulan Agnitum Outpost Security Suite Pro 7.0 [15] sürümü ile anti-virüs ve güvenlik duvarı uygulamalarının gecikme değerlerini nasıl etkilediği ölçülmüştür. Her ne kadar sistem seviyesinde çalışmasına rağmen güvenlik duvarı bütün ağ trafiğini kontrol ettiği için gecikme değerlerinin ciddi olarak etkilenmesi beklenmelidir.

3.1. Test için Farklı Ağ Yapılandırmaları

Bu test grupları değişik donanımsal ve yazılımsal yapılandırmalarla tekrarlanarak bu değişimlerin yanıt verme sürelerini nasıl etkilediğinin ölçülmesi amaçlanmıştır. Bu yapılandırmalar aşağıdaki gibidir:

- *Ethernet ağı:*
    - *Microsoft Windows XP SP1:* Windows XP işletim sistemi SP1 servis paketi sürümünde ilk sürümdeki bazı ciddi açıklar giderilmiştir.
    - *Microsoft Windows XP SP3:* Bu servis paketi sürümünde sadece içeriye bağlantıları denetleyen basit bir bütünleşik güvenlik duvarı mevcuttur.
    - *Microsoft Windows 7 Pro x86:* Microsoft, yeni işletim sistemi sürümünde çekirdek de dâhil olmak üzere sisteme ek yük getirecek arayüz değişiklikleri içeren fakat yeni nesil işlemci ve grafik kartlarının özelliklerini daha iyi kullanan radikal değişikliklere gitmiştir. Bu nedenle bu sürümde yaptığımız testlerde gecikme değerlerinin yükselmesini bekliyoruz.

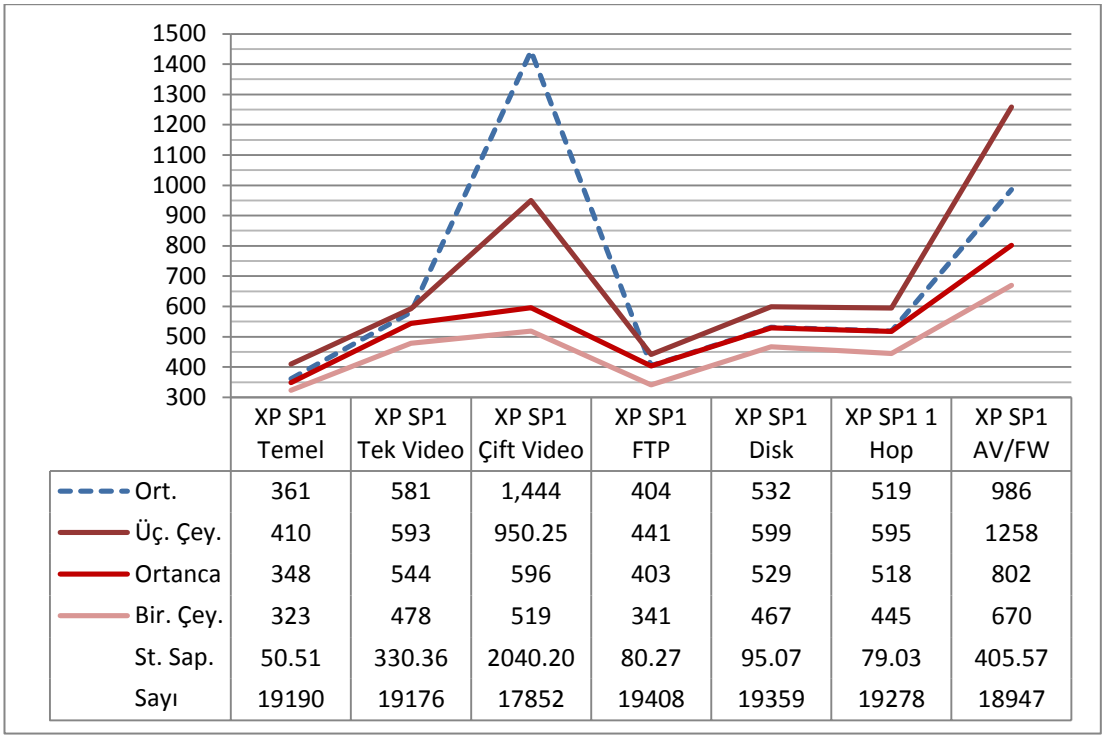
- *Kablosuz ağ:*
    - *Microsoft Windows 7 Pro x86:* Kablosuz ağ bağlantısı üzerindeki gecikmelerin Ethernet ile kıyaslanmasını amaçlıyoruz.

3.2. Test Sonuçları

Yapılan testlerden yazılan Java uygulaması aracılığıyla gecikme değerleri elde edilmiştir. Bu değerler incelenirken TCP 3-yönlü el sıkışma değerleri için 1 ms, DNS değerleri için de 10 ms sınır konularak bu değerlerin üzerindeki ölçümler çıkarılmıştır. Bununla muhtemel yanlış eşleşmelerin ve anormal büyüklükte gecikmelerin ölçümlere etki etmesi engellemiştir. Sonuçlarda gösterilen tüm değerler µs (mikro saniye) birimindedir. Tablolarda en alt satırda ölçüm sayıları gösterilmiştir.

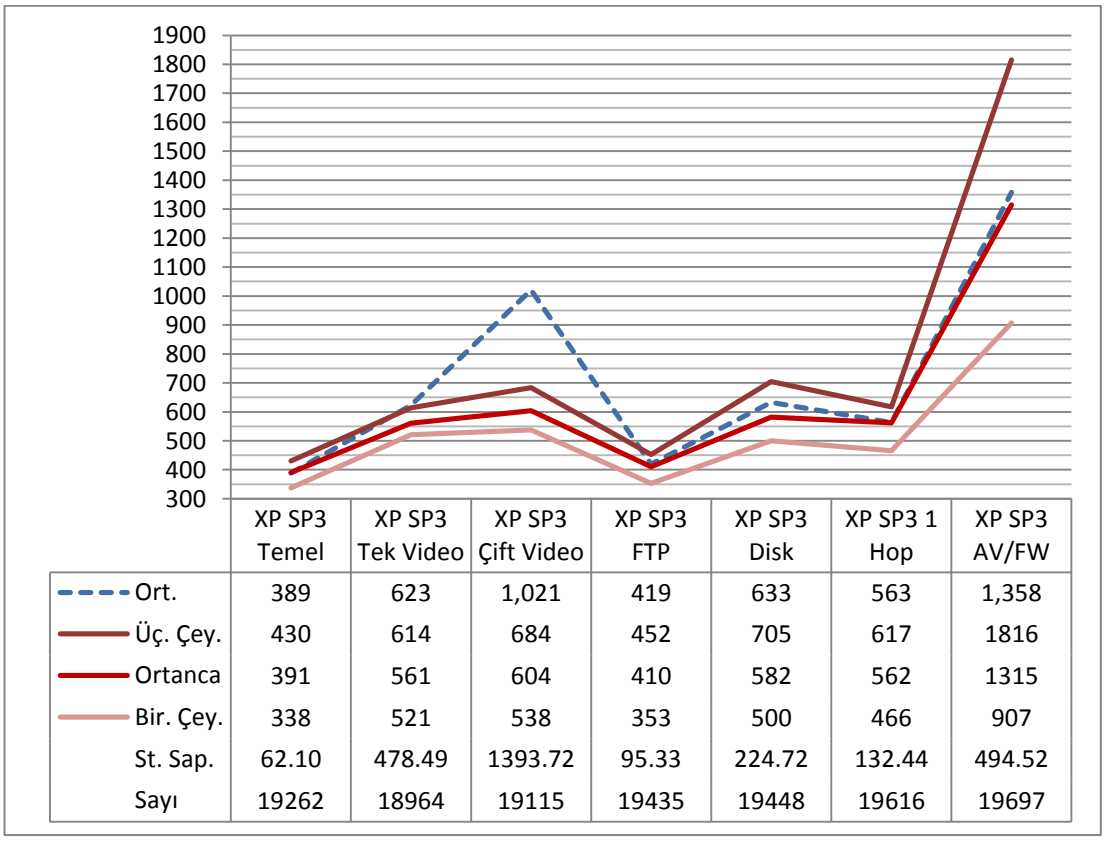
Ethernet üzerinden Windows XP SP1 yüklü İstemci için yapılan DNS ölçümleri Tablo 3.1'de ve histogram Tablo 3.3'te görülmektedir. Sonuçlarda beklendiği gibi video testlerinde ve anti-virüs/güvenlik duvarı testinde standart sapma değerleri yükselmiş, özellikle çift video oynatılması makinenin her iki çekirdeğinde yüksek MİB kullanımına yol açarak ağ başarımını etkilemiştir. Ölçüm makinesinin araya eklenerek ölçümlerin NAT makinesinden yapılması yaklaşık 155-160 µs gecikme eklemiştir. Bu da aradaki makinenin ortalama 80 µs civarı gecikmeye sebep olduğunu göstermektedir. FTP etkinliğinin neden olduğu ek gecikme çok düşüktür, disk (G/Ç) işlemleri ise yaklaşık 1 hop testine yakın gecikmeye sebep olmuştur.

Tablo 3.1. Windows XP SP1 DNS Test Sonuçları (µs)



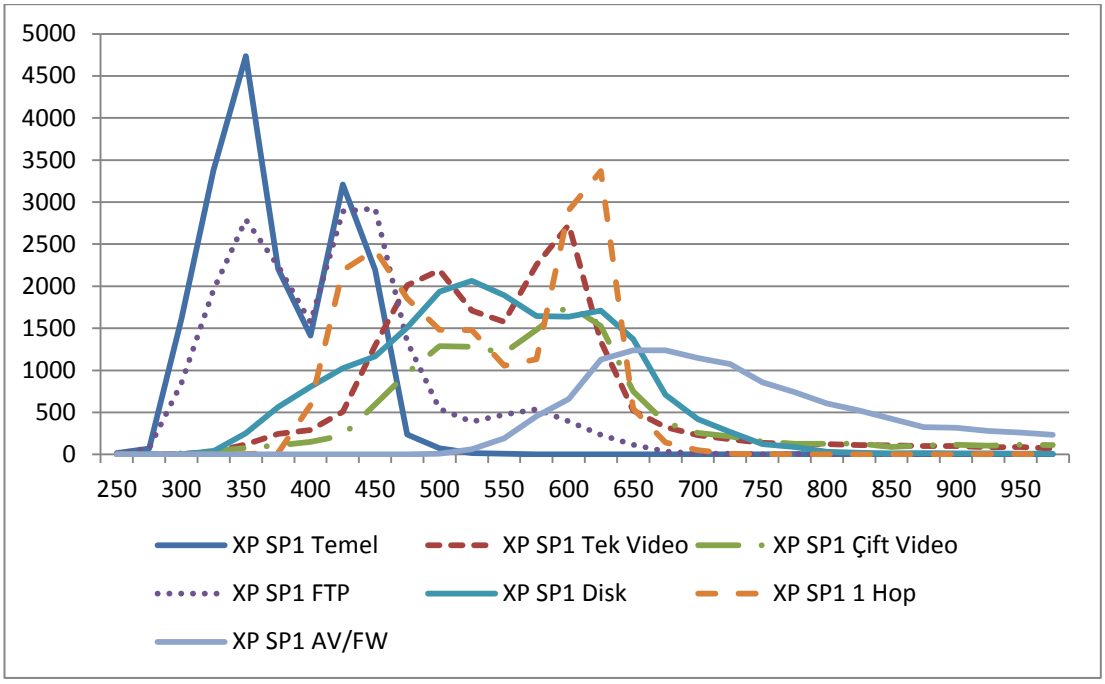
	XP SP1 Temel	XP SP1 Tek Video	XP SP1 Çift Video	XP SP1 FTP	XP SP1 Disk	XP SP1 1 Hop	XP SP1 AV/FW
Ort.	361	581	1,444	404	532	519	986
Üç. Çey.	410	593	950.25	441	599	595	1258
Ortanca	348	544	596	403	529	518	802
Bir. Çey.	323	478	519	341	467	445	670
St. Sap.	50.51	330.36	2040.20	80.27	95.07	79.03	405.57
Sayı	19190	19176	17852	19408	19359	19278	18947

Tablo 3.2. Windows XP SP3 DNS Test Sonuçları (µs)

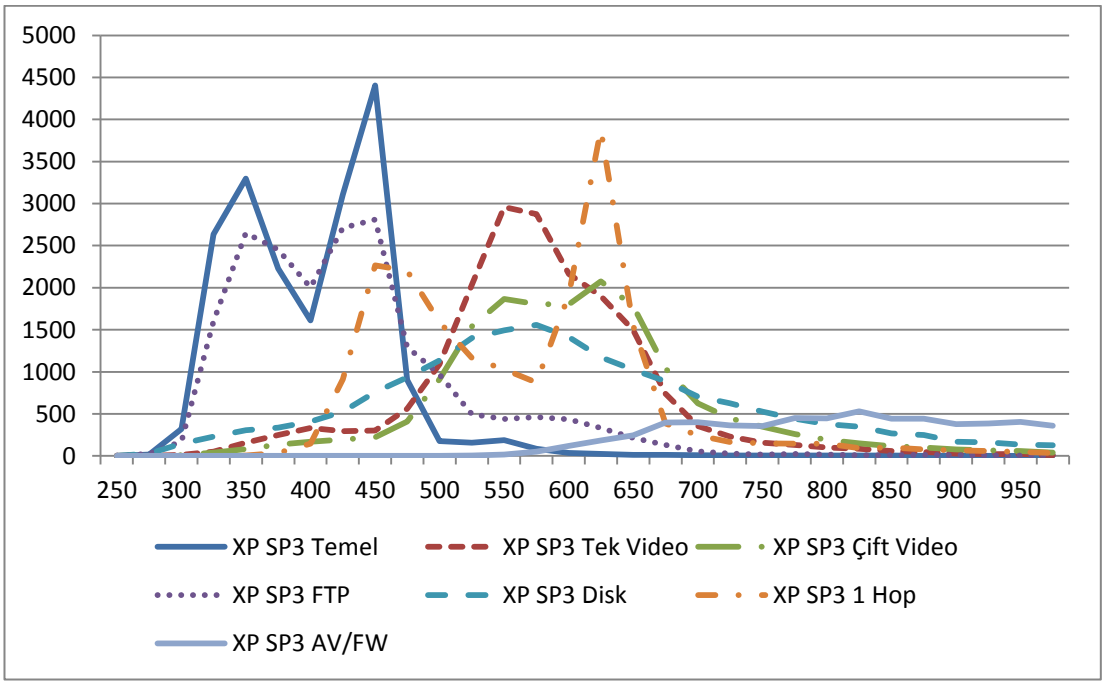


	XP SP3 Temel	XP SP3 Tek Video	XP SP3 Çift Video	XP SP3 FTP	XP SP3 Disk	XP SP3 1 Hop	XP SP3 AV/FW
Ort.	389	623	1,021	419	633	563	1,358
Üç. Çey.	430	614	684	452	705	617	1816
Ortanca	391	561	604	410	582	562	1315
Bir. Çey.	338	521	538	353	500	466	907
St. Sap.	62.10	478.49	1393.72	95.33	224.72	132.44	494.52
Sayı	19262	18964	19115	19435	19448	19616	19697

Tablo 3.3. Windows XP SP1 DNS Yanıt Süreleri Histogram (µs)



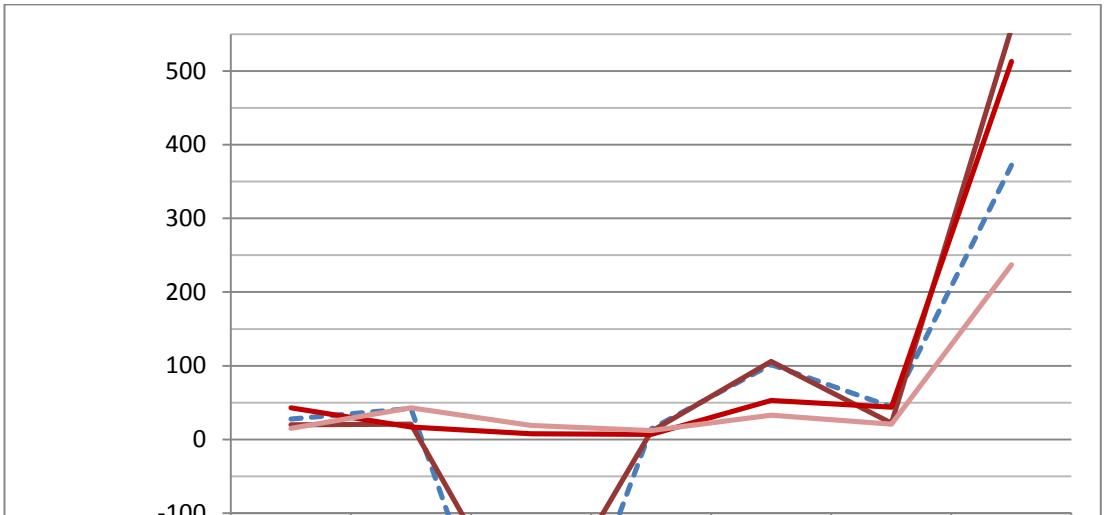
Tablo 3.4. Windows XP SP3 DNS Yanıt Süreleri Histogram (µs)



Ethernet üzerinden Windows XP SP3 için yapılan ölçümler Tablo 3.2'de ve histogramı Tablo 3.4'te görülmektedir. Sonuçlar XP SP1 testine benzer şekil

göstermektedir. Burada aradaki fazladan makinenin etkisi ortalama 175 µs ile ihmal edilebilir bir farkla öncekiyle benzer çıkmış, fakat özellikle güvenlik duvarının etkisi büyümüştür. Çünkü XP SP3'te varsayılan olarak basit bir güvenlik duvarı daha mevcuttur. Genel olarak kıyasladığımızda iki servis paketi sürümünün arasındaki farklar Tablo 3.5'te gösterilmiştir. XP SP3'teki güvenlik duvarı etkisi özellikle AV/FW testinde görülebilmektedir ve genel olarak SP3'ün ağ başarımı bu nedenle daha düşüktür. Çift video testinde test koşullarından da kaynaklanma olasılığı olmakla birlikte yeni nesil işlemcilerin daha iyi desteklenmesinin çift çekirdek video başarımını olumlu etkilediği yorumu yapılabilir.

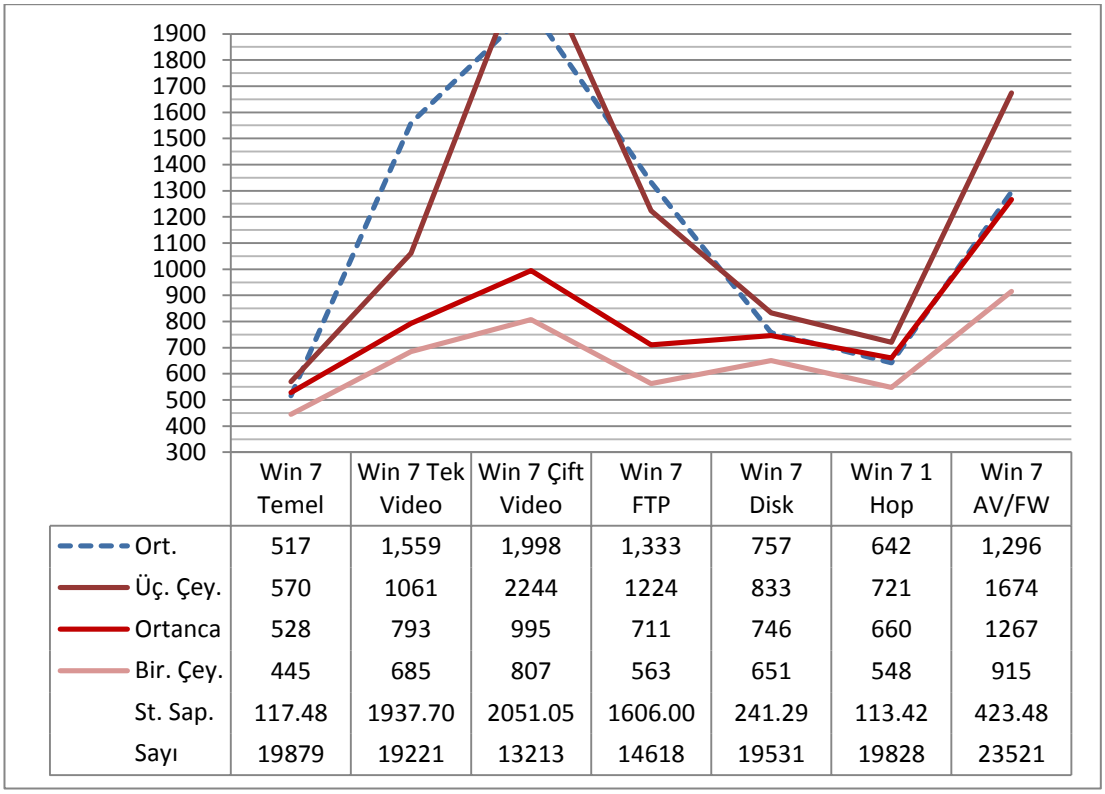
Tablo 3.5. Windows XP SP3-SP1 Testleri Arasındaki Farklar (µs)



	Temel	Tek Video	Çift Video	FTP	Disk	1 Hop	AV/FW
Ort.	28	42	-423	14	101	44	372
Üç. Çey.	20	21	-266	11	106	22	558
Ortanca	43	17	08	07	53	44	513
Bir. Çey.	15	43	19	12	33	21	237

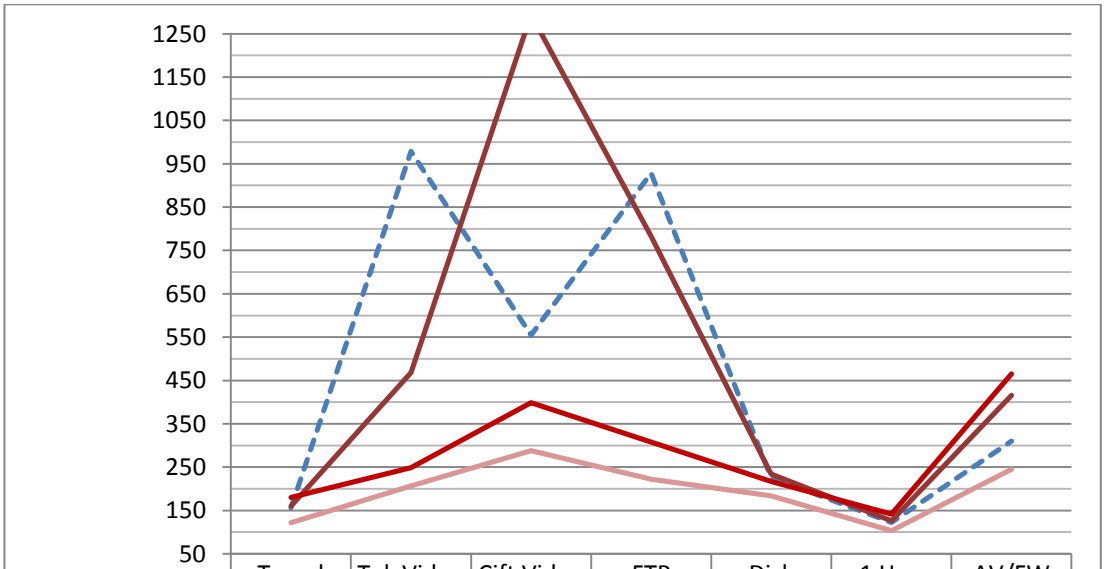
Ethernet üzerinden yapılan Windows 7 Professional x86 ölçümlerinin sonuçları ise Tablo 3.6'da ve histogramı Tablo 3.8'de gösterilmiştir. Buna göre radikal bazı değişiklikler içeren Windows 7 altında test sonuçları da öncekilere göre önemli değişiklikler göstermiştir. Örneğin daha önce temel teste çok yakın davranan FTP testinde bu sefer standart sapma yükselmiş ve bunun etkisiyle ortalama da yukarı çekilmiştir.

Tablo 3.6. Windows 7 Professional x86 DNS Test Sonuçları (µs)



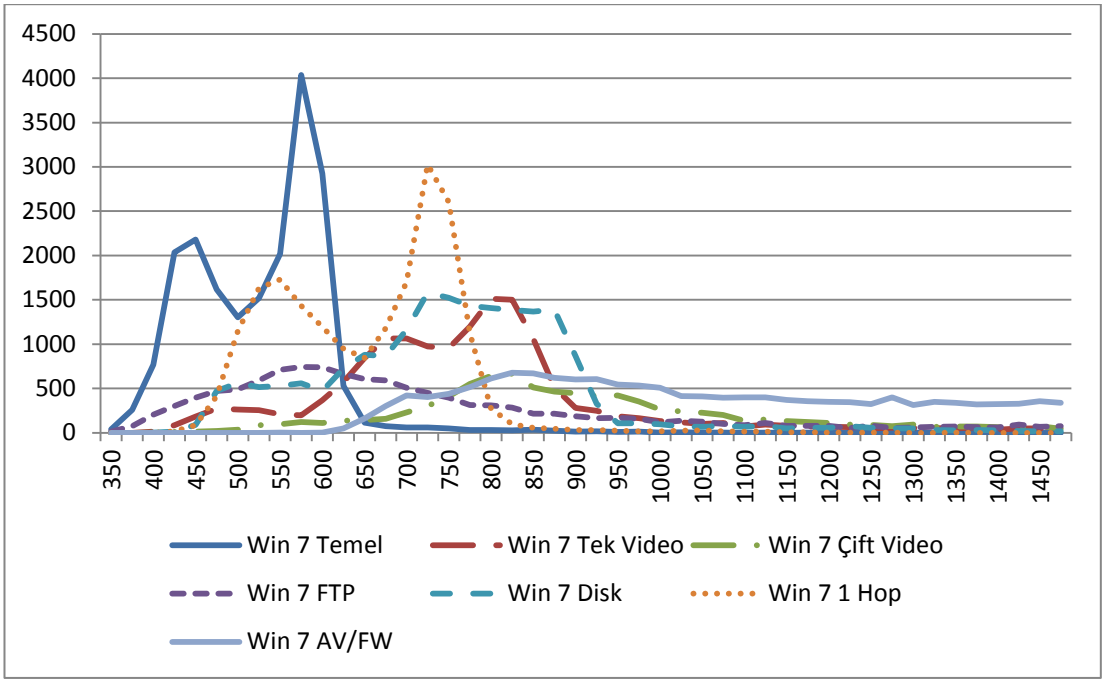
	Win 7 Temel	Win 7 Tek Video	Win 7 Çift Video	Win 7 FTP	Win 7 Disk	Win 7 1 Hop	Win 7 AV/FW
Ort.	517	1,559	1,998	1,333	757	642	1,296
Üç. Çey.	570	1061	2244	1224	833	721	1674
Ortanca	528	793	995	711	746	660	1267
Bir. Çey.	445	685	807	563	651	548	915
St. Sap.	117.48	1937.70	2051.05	1606.00	241.29	113.42	423.48
Sayı	19879	19221	13213	14618	19531	19828	23521

Tablo 3.7. Windows 7-Windows XP SP1 DNS Testleri Arasındaki Farklar (µs)

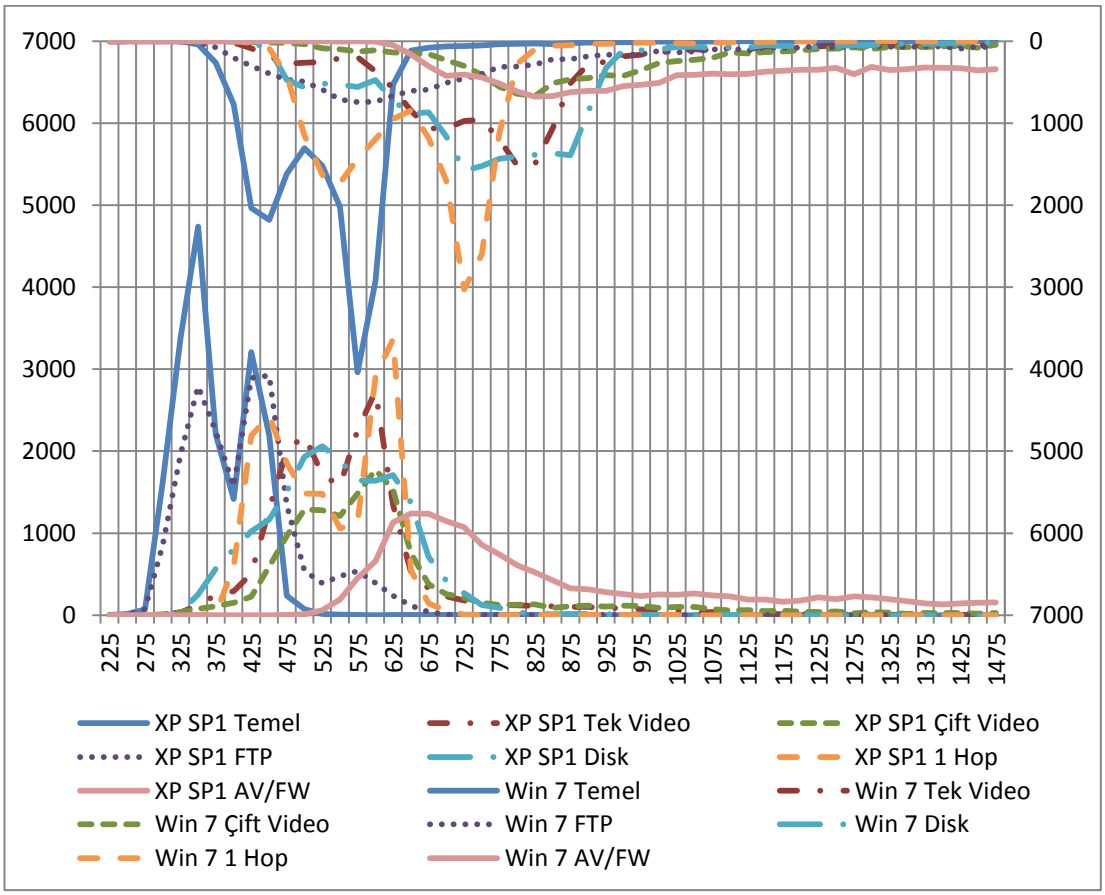


	Temel	Tek Video	Çift Video	FTP	Disk	1 Hop	AV/FW
Ort.	155	978	554	928	226	123	310
Üç. Çey.	160	468	1,294	783	234	126	416
Ortanca	180	249	399	308	217	142	465
Bir. Çey.	122	207	288	222	184	103	245

Tablo 3.8. Windows 7 Professional x86 DNS Yanıt Süreleri Histogram (µs)

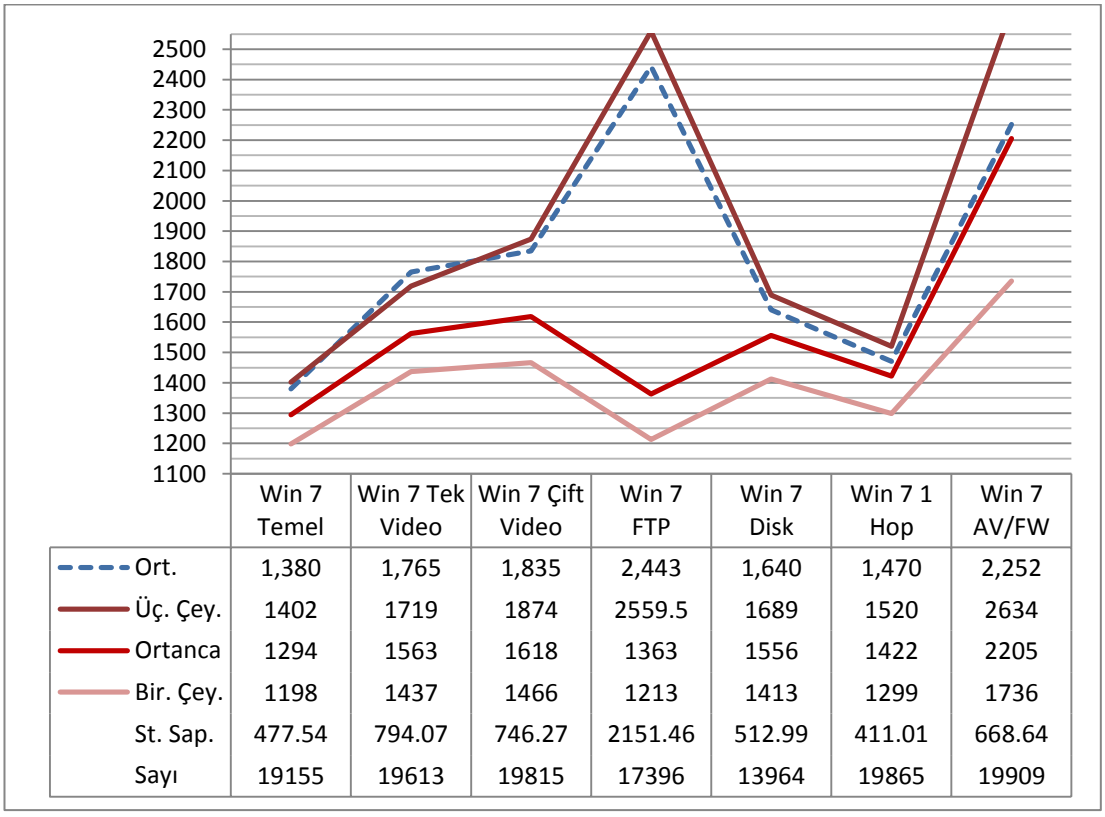


Tablo 3.9. Windows 7 Pro ve XP SP1 Testleri DNS Histogram Karşılaştırma (µs)



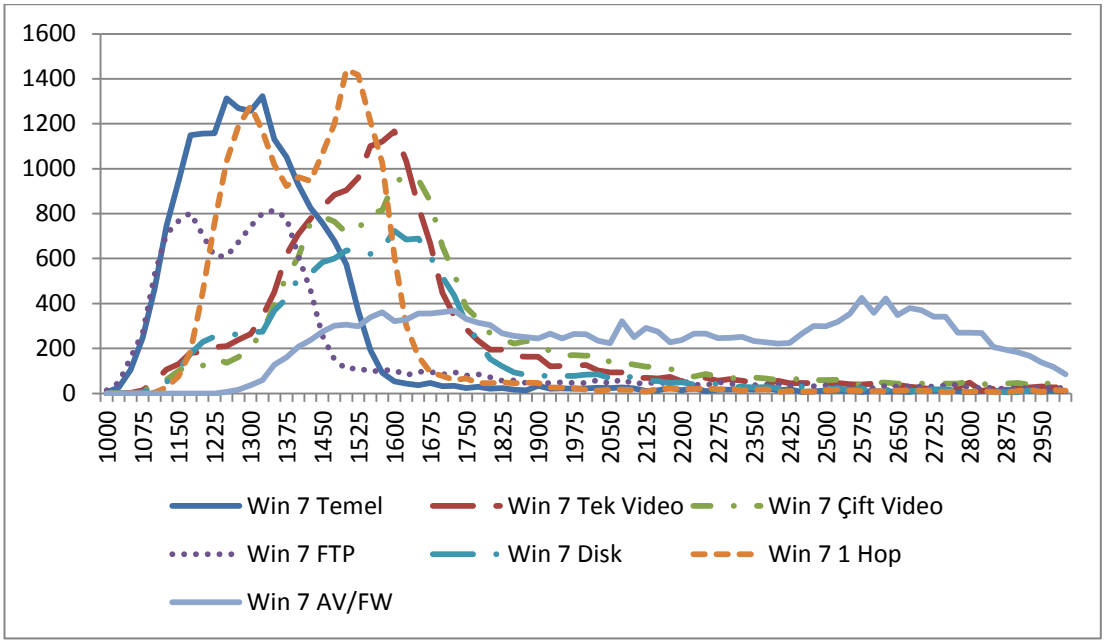
Windows 7 ve Windows SP1 arasındaki fark Tablo 3.7'de ve histogramlar arasındaki fark Tablo 3.9'da gösterilmiştir. Burada her bir test iki platform için de aynı renkte çizilmiş ve kolay kıyaslanabilmesi için Windows 7 değerleri yukarıdan aşağıya çizilmiştir. Windows 7 içerisinde de Windows XP SP3'teki gibi bütünleşik güvenlik duvarı gelmektedir, fakat bu güvenlik duvarı SP3'teki kadar basit değildir, içeri gelen ve dışarı giden bağlantıları denetleyebilmekte, kurallarında ileri düzeyde ayarlamalar yapılabilmektedir. Buna göre video testlerinde de önemli gecikme artışları olmuş, fakat gecikmeler çift video testi için tek video testi kadar artmamıştır. Bunun açıklaması Windows 7'deki yeni nesil çok çekirdekli işlemci desteğinin etkisi olabilir. Tüm testlerde genel olarak artış görülmekte, fakat 1 hop testinde bekleneceği gibi temel teste göre bir fark görülmemektedir. Çünkü aradaki makinenin etkisi İstemci sistemden bağımsızdır. FTP testindeki değerlerin Windows 7'de histograma yayıldığı, buna karşılık SP1'de daha düşük bir standart sapmaya sahip olduğu görülmektedir. Bunda Windows 7'deki bütünleşik güvenlik duvarının etkisi olabilir.

Tablo 3.10. Kablosuz Ağda Windows 7 Pro x86 DNS Test Sonuçları (µs)



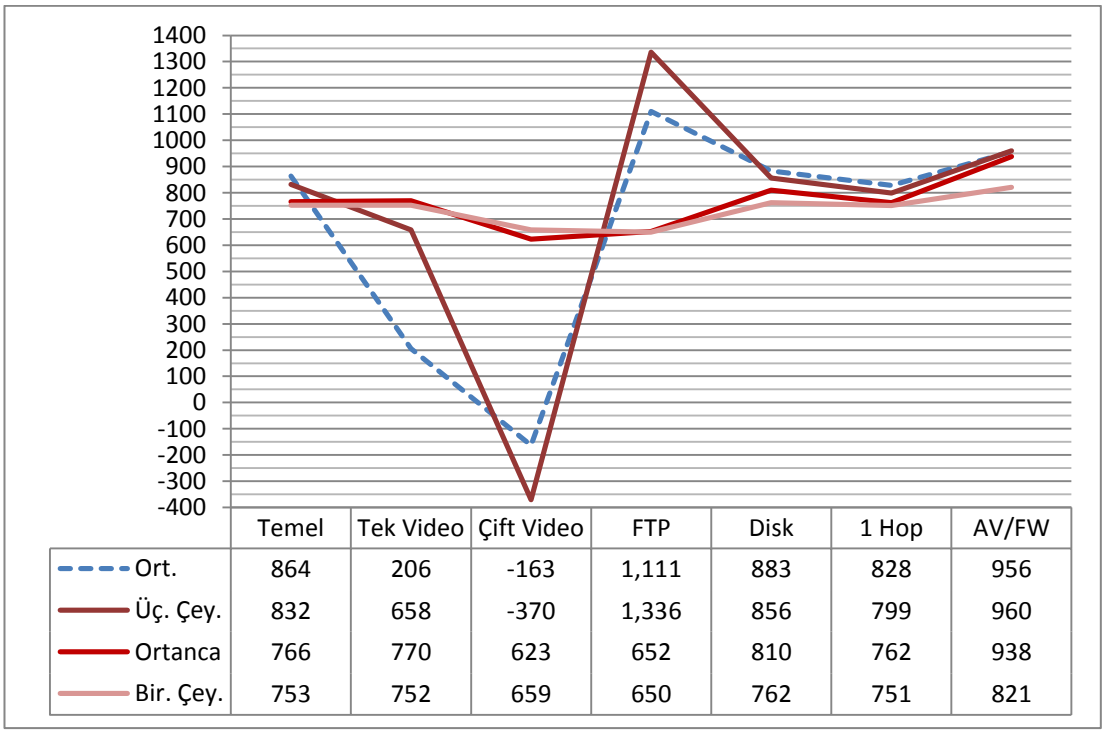
	Win 7 Temel	Win 7 Tek Video	Win 7 Çift Video	Win 7 FTP	Win 7 Disk	Win 7 1 Hop	Win 7 AV/FW
Ort.	1,380	1,765	1,835	2,443	1,640	1,470	2,252
Üç. Çey.	1402	1719	1874	2559.5	1689	1520	2634
Ortanca	1294	1563	1618	1363	1556	1422	2205
Bir. Çey.	1198	1437	1466	1213	1413	1299	1736
St. Sap.	477.54	794.07	746.27	2151.46	512.99	411.01	668.64
Sayı	19155	19613	19815	17396	13964	19865	19909

Tablo 3.11. Kablosuz Ağda Windows 7 DNS Yanıt Süreleri Histogram (µs)



Kablosuz ağ ortamında Windows 7 üzerinde yapılmış test sonuçları Tablo 3.10'da ve histogram Tablo 3.11'de görülmektedir. Beklendiği gibi kablosuz ağ ortamı Ethernet ağına göre çok daha yüksek yanıt süresi değerleri vermektedir.

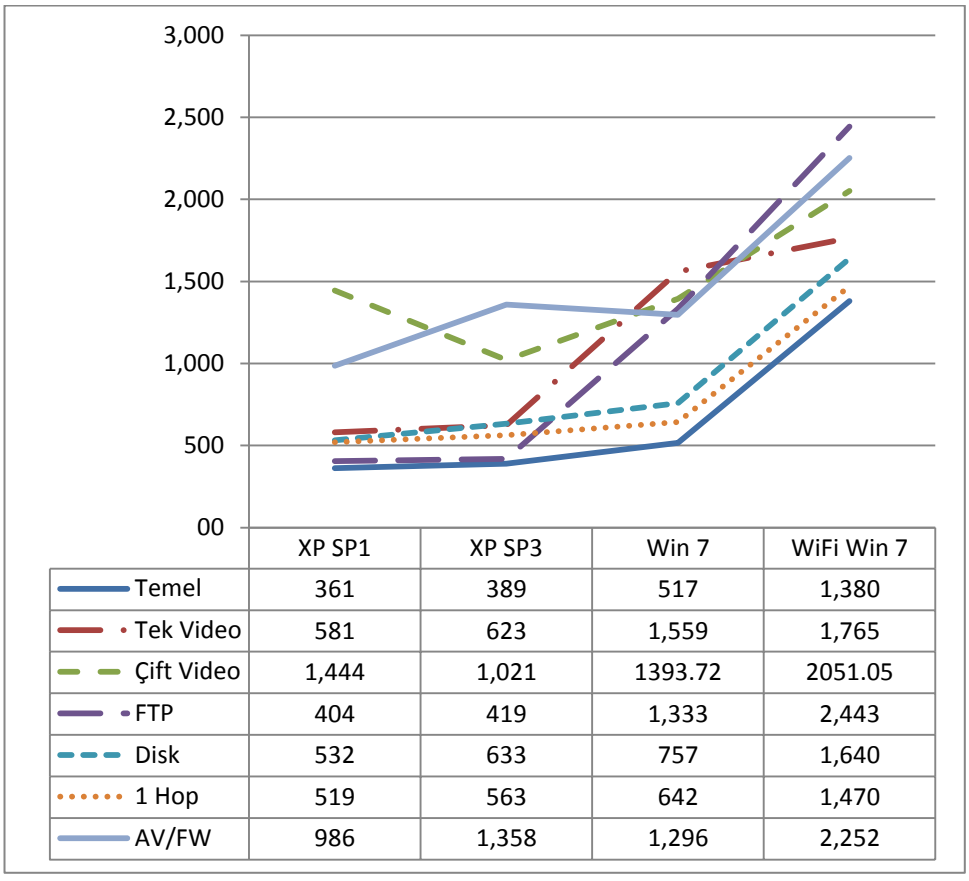
Tablo 3.12. Kablosuz-Ethernet Ortamları Arasında DNS Süre Farkları (µs)



	Temel	Tek Video	Çift Video	FTP	Disk	1 Hop	AV/FW
Ort.	864	206	-163	1,111	883	828	956
Üç. Çey.	832	658	-370	1,336	856	799	960
Ortanca	766	770	623	652	810	762	938
Bir. Çey.	753	752	659	650	762	751	821

Ethernet ortamındaki aynı sistemle aradaki süre farkları Tablo 3.12'de gösterilmiştir. Kablosuz ortamda en çok etkilenen aynı zamanda bu ortamı yoğun olarak da kullanan FTP testidir. Video testlerindeki standart sapmalar düşmüş, bu da ortalamaları etkilemiştir. Genel olarak ortalama 800 µs civarında kablosuz ortamın getirdiği gecikme olduğu görülmektedir.

Tablo 3.13. Topluca DNS Test Süresi Ortalamaları (µs)



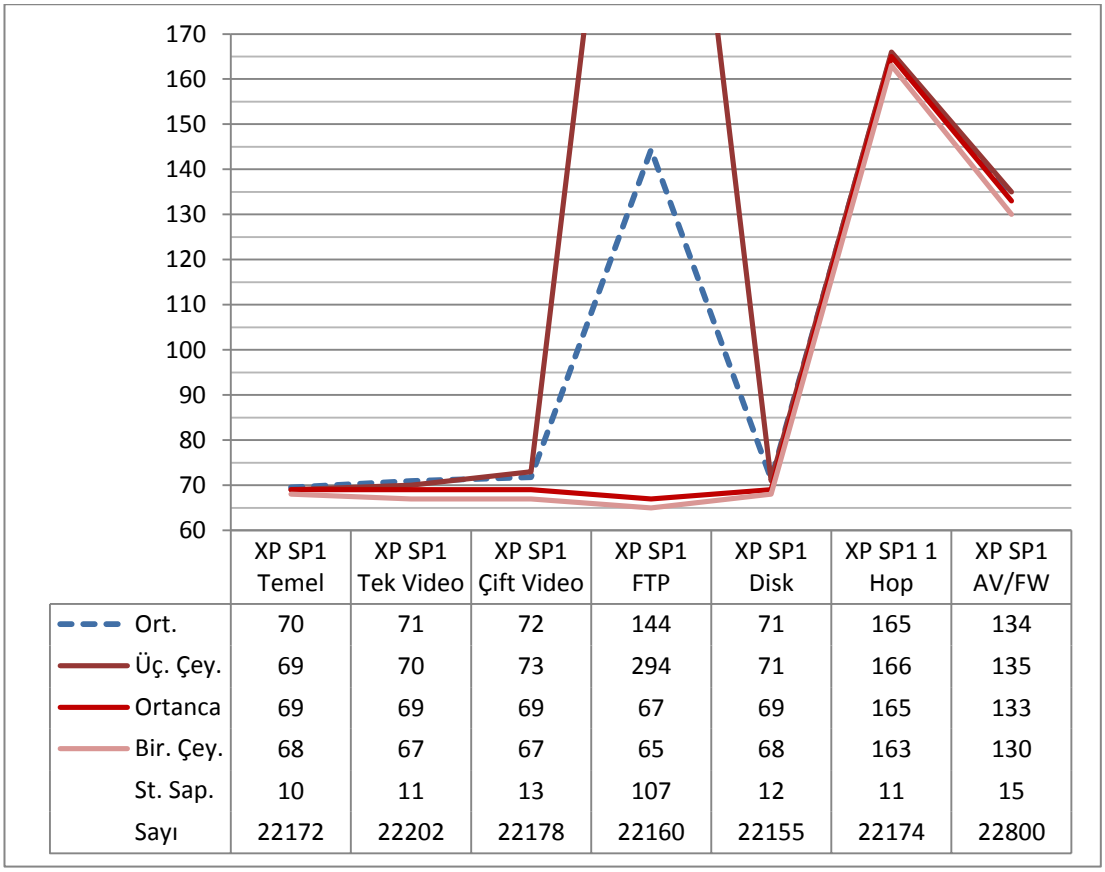
	XP SP1	XP SP3	Win 7	WiFi Win 7
Temel	361	389	517	1,380
Tek Video	581	623	1,559	1,765
Çift Video	1,444	1,021	1393.72	2051.05
FTP	404	419	1,333	2,443
Disk	532	633	757	1,640
1 Hop	519	563	642	1,470
AV/FW	986	1,358	1,296	2,252

DNS testlerinin tüm ortam ve platformlar için topluca süre ortalamaları özeti Tablo 3.13'te verilmiştir. Genel olarak işletim sistemi modernleştikçe süreler yükselmekte, kablosuz ağ ortamının gecikmeyi arttırıcı etkisi görülmektedir.

Ethernet üzerinden yapılan Windows XP SP1 testi için TCP ölçümleri Tablo 3.14'te ve histogramı Tablo 3.15'te görülmektedir. Beklendiği gibi TCP sonuçları çok daha alt düzeyde olduğu için DNS sonuçlarına göre oldukça etkinliklerden bağımsız

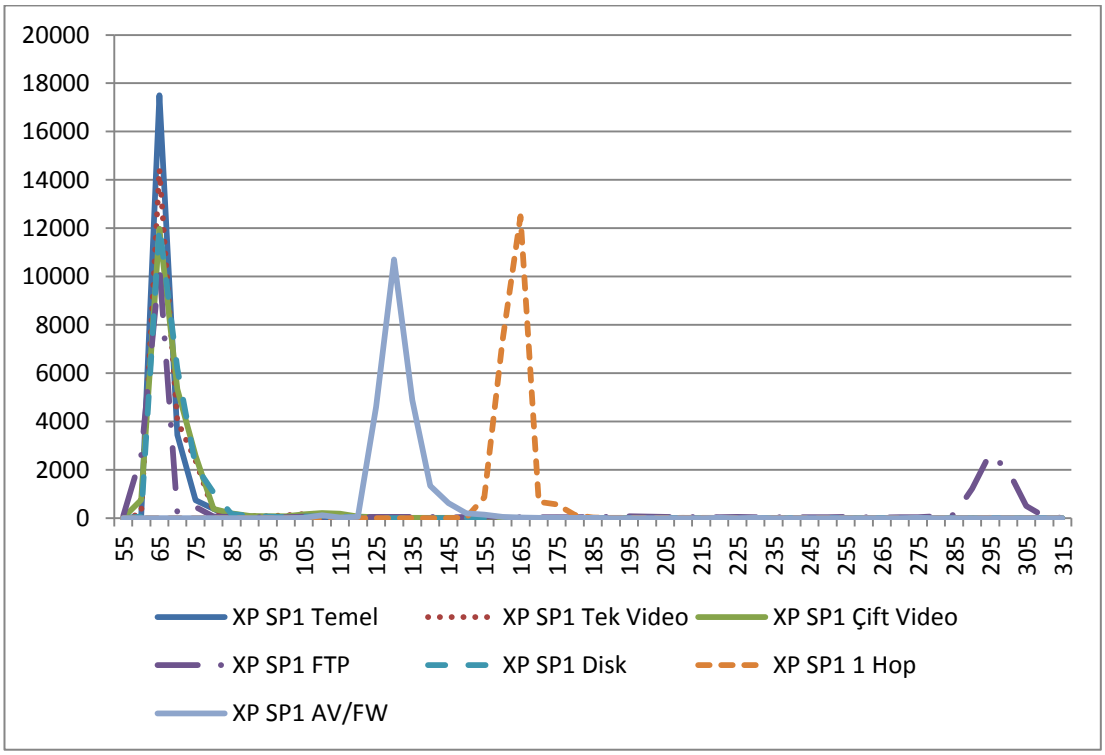
durumdadır. Sadece FTP, 1 hop ve AV/FW testlerinde beklenebileceği gibi etki görülmektedir. FTP testindeki yoğun ağ kullanımı, alt seviyedeki gecikmeyi etkileyebilmiştir, hatta standart sapma açısından en etkili test budur. Histogramdan görüldüğü üzere bu testte muhtemelen ağ trafiğinin sıkıştığı noktalarda değerler yükselmekte ve 295 µs civarını bulmakta, diğer durumlarda diğer testlerle yakın çıkmaktadır. 1 hop testinde ortalama 95-100 µs temel teste göre gecikme görülmektedir ki DNS testinde gördüğümüz 155-160 µs gecikmeyle paraleldir. Aradaki fark DNS testinde ölçülen işlemlerin gerçekleştiği ağ katmanıyla TCP testinde ölçülen işlemlerin gerçekleştiği katman arasındaki işlem süresi olabilir. Ayrıca daha önce bahsedildiği gibi DNS testinde ölçülen uygulamanın ağ başarımı açısından keyfiliği daha fazladır. Güvenlik duvarının etkisi de beklendiği gibi yüksek olmuştur ve yaklaşık 65 µs gecikme getirmektedir. Başarımına göre çeşitli güvenlik duvarı uygulamaları arasında bu süreler değişiklik gösterebilir.

Tablo 3.14. Windows XP SP1 TCP Test Sonuçları (µs)

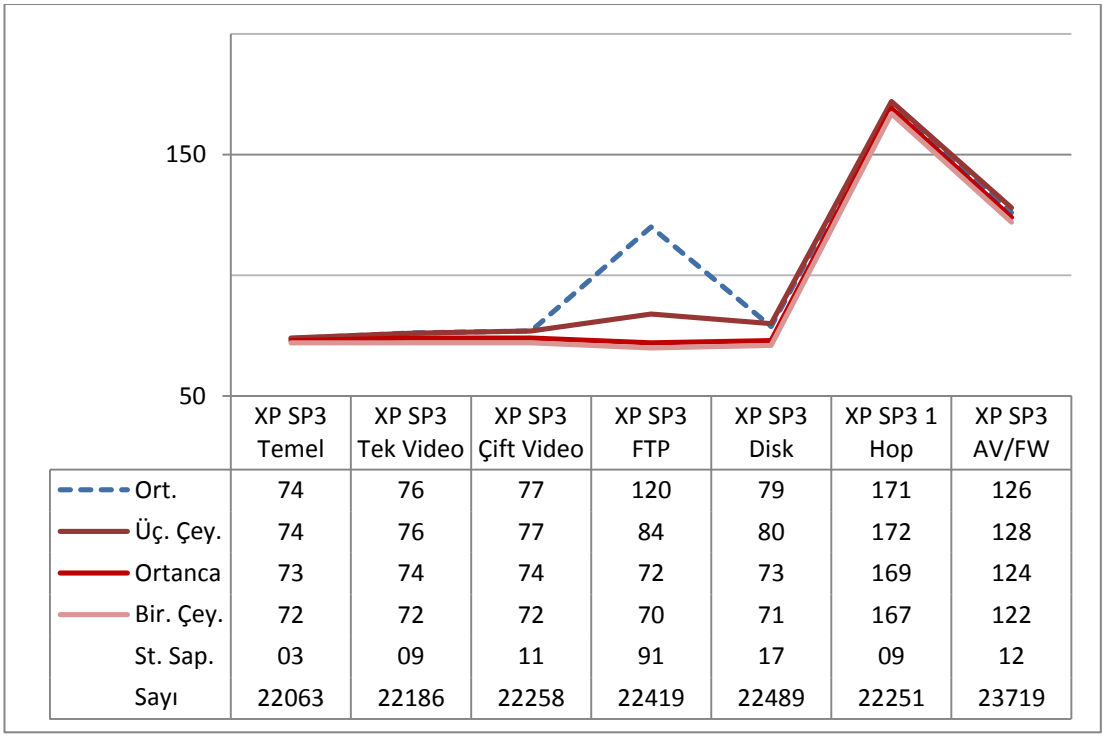


	XP SP1 Temel	XP SP1 Tek Video	XP SP1 Çift Video	XP SP1 FTP	XP SP1 Disk	XP SP1 1 Hop	XP SP1 AV/FW
Ort.	70	71	72	144	71	165	134
Üç. Çey.	69	70	73	294	71	166	135
Ortanca	69	69	69	67	69	165	133
Bir. Çey.	68	67	67	65	68	163	130
St. Sap.	10	11	13	107	12	11	15
Sayı	22172	22202	22178	22160	22155	22174	22800

Tablo 3.15. Windows XP SP1 TCP Yanıt Süreleri Histogram (µs)



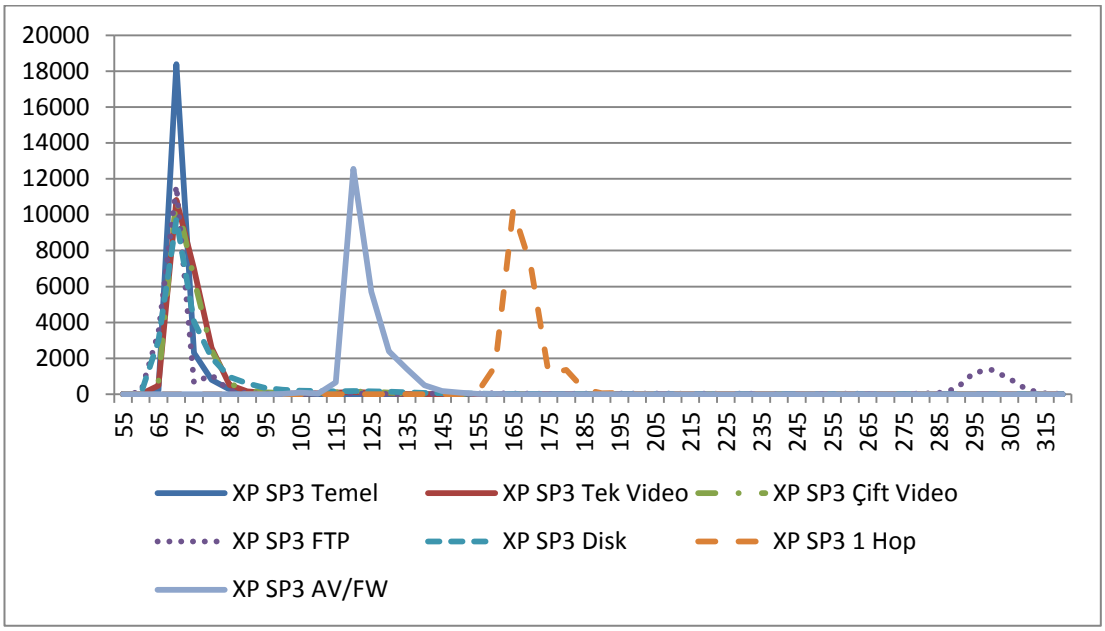
Tablo 3.16. Windows XP SP3 TCP Test Sonuçları (µs)



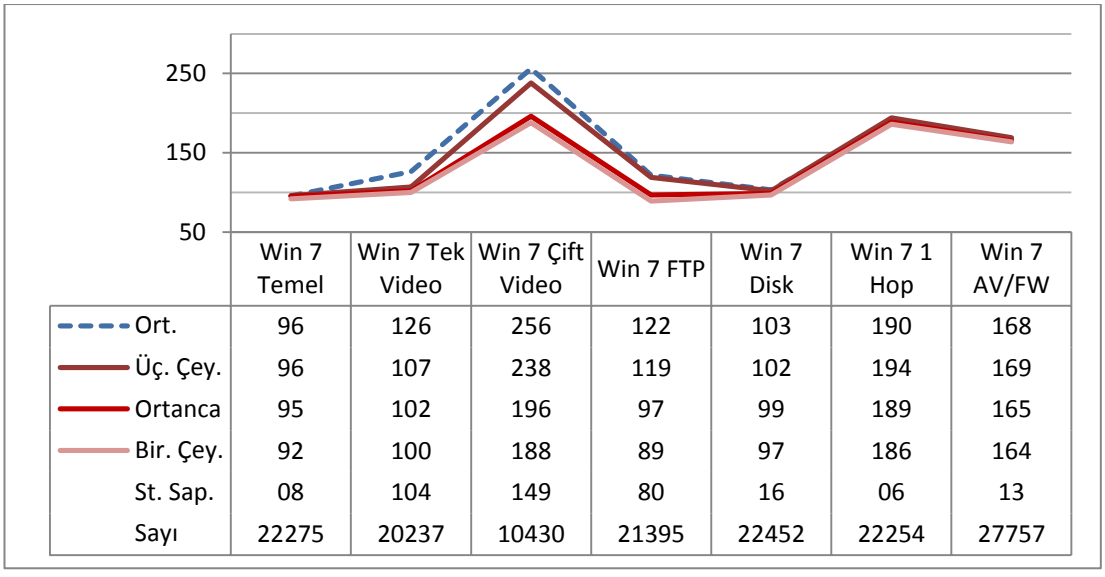
	XP SP3 Temel	XP SP3 Tek Video	XP SP3 Çift Video	XP SP3 FTP	XP SP3 Disk	XP SP3 1 Hop	XP SP3 AV/FW
Ort.	74	76	77	120	79	171	126
Üç. Çey.	74	76	77	84	80	172	128
Ortanca	73	74	74	72	73	169	124
Bir. Çey.	72	72	72	70	71	167	122
St. Sap.	03	09	11	91	17	09	12
Sayı	22063	22186	22258	22419	22489	22251	23719

Ethernet ağındaki Windows XP SP3 üzerinde yapılan TCP ölçümlerinin sonuçları Tablo 3.16'da ve histogramı Tablo 3.17'de gösterilmiştir. Burada da sonuçlar XP SP1 testiyle paralel görünmektedir. Güvenlik duvarı için sonuçlar SP1'e göre biraz daha iyi hale gelmiştir. 1 hop testi benzer gecikmeleri göstermektedir. FTP'de de yine ikiye ayrılmış bir histogram dikkat çekmektedir. Ortalama olarak değerler SP1'e göre neredeyse hiç değişmemiştir.

Tablo 3.17. Windows XP SP3 TCP Yanıt Süreleri Histogram (µs)



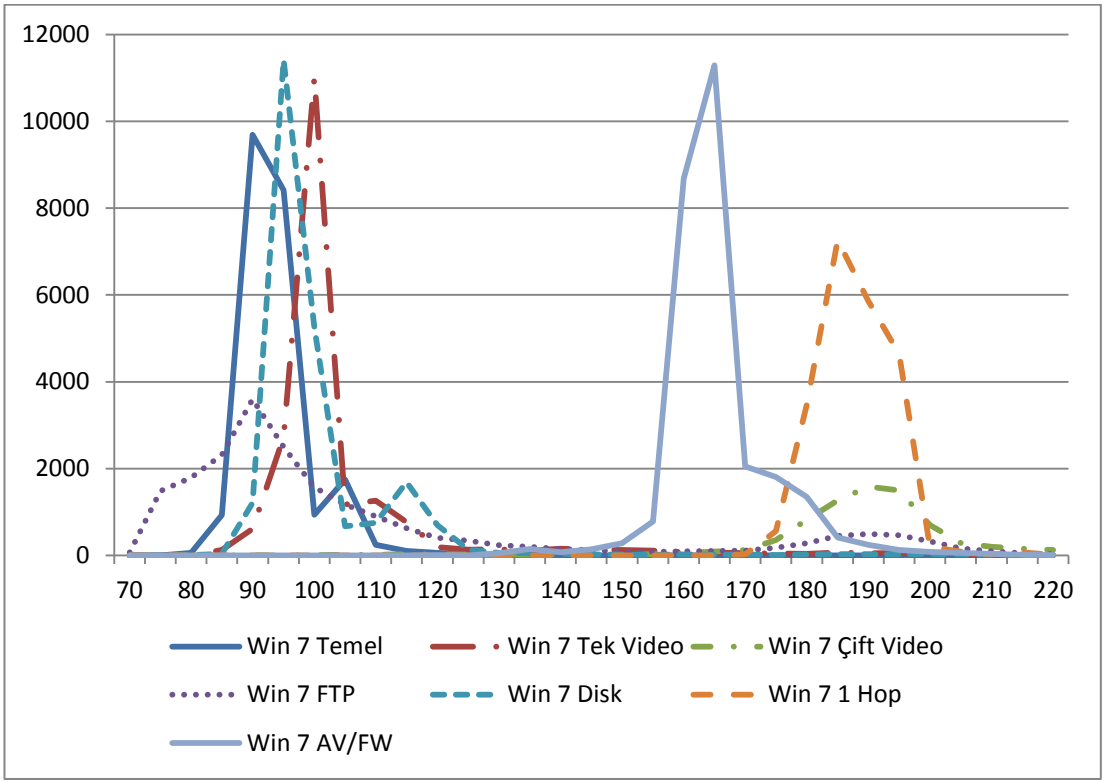
Tablo 3.18. Windows 7 Professional x86 TCP Test Sonuçları (µs)



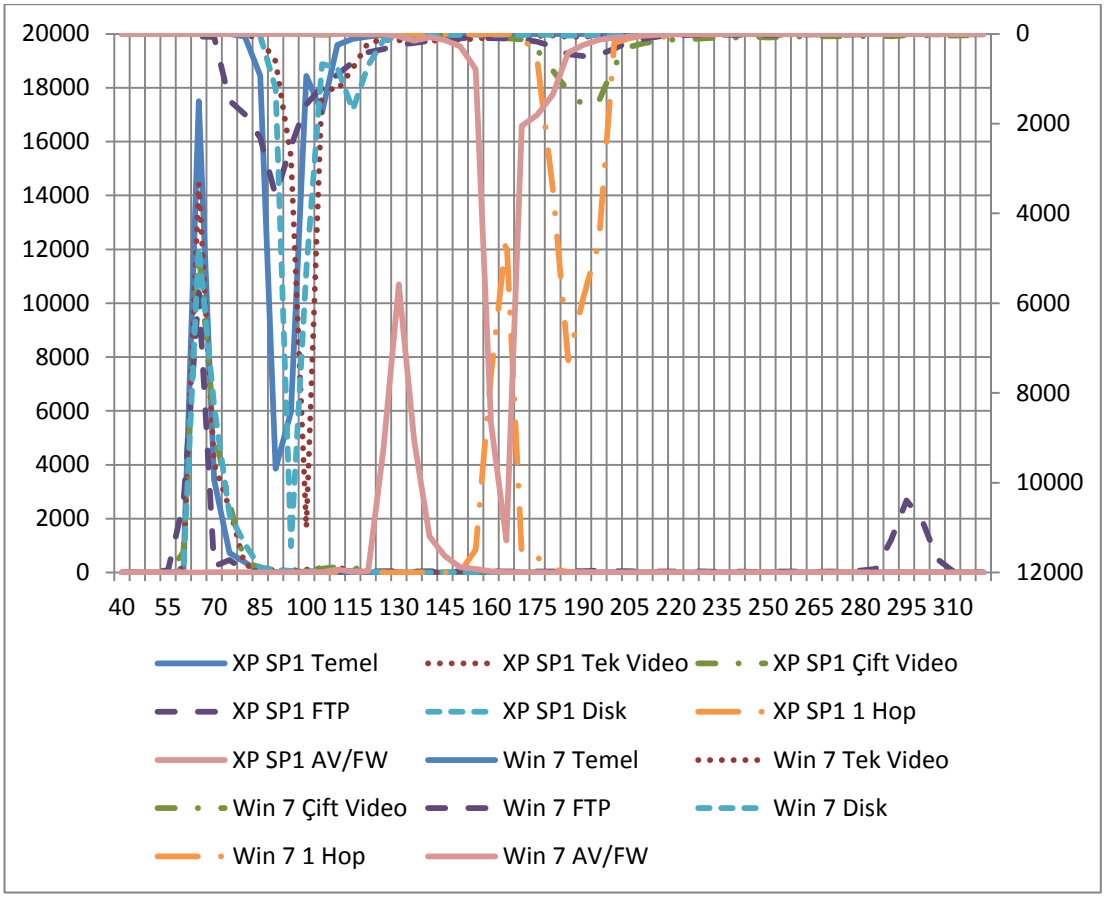
	Win 7 Temel	Win 7 Tek Video	Win 7 Çift Video	Win 7 FTP	Win 7 Disk	Win 7 1 Hop	Win 7 AV/FW
Ort.	96	126	256	122	103	190	168
Üç. Çey.	96	107	238	119	102	194	169
Ortanca	95	102	196	97	99	189	165
Bir. Çey.	92	100	188	89	97	186	164
St. Sap.	08	104	149	80	16	06	13
Sayı	22275	20237	10430	21395	22452	22254	27757

Ethernet ağı üzerinden Windows 7 Professional x86 için yapılan TCP ölçümlerinin sonuçları Tablo 3.18'de ve histogramı Tablo 3.19'da görülmektedir. Bu ölçümlerdeki farklılıklar hemen göze çarpmaktadır, özellikle çift video testi diğer sistemlerdeki gibi birlikte olduğu diğer test sonuçlarından ayrılmıştır. Bunun sebebi yeni mimaride daha iyi çok çekirdekli işlemci desteği ve grafik kartının daha etkin kullanımı olabilir. Sistemin kullanıcı arayüzü de grafik kartı ile çalıştığından video oynatılması genel başarımı daha çok etkilemektedir. FTP testindeki başarım temele göre ciddi ölçüde yükselmiştir. Ortalama, temel değerlere yakın seyretmekle birlikte, ağ altyapısının başarımının yoğun kullanımla yükseldiği görülmektedir. Bunun sebebi yeniden tasarlanan ağ çatısı olabilir. Bu da beraberinde yüksek başarım ve kullanıma göre kendisini iyileştiren bir altyapı getirmiştir [16,17]. Güvenlik duvarı testinin gösterdiği sonuçlara göre Windows 7'deki genel sistem başarımı azalması buraya yansımıştır. Test ortalamalarına bakarsak Windows XP SP1-SP3'e göre yaklaşık 20-30 µs gecikme olmaktadır. 1 hop testinin temel teste farkı ortalama 95-100 µs olduğu görülmektedir. Bu da beklendiği gibi sistemden sisteme değişmeyen bir değerdir.

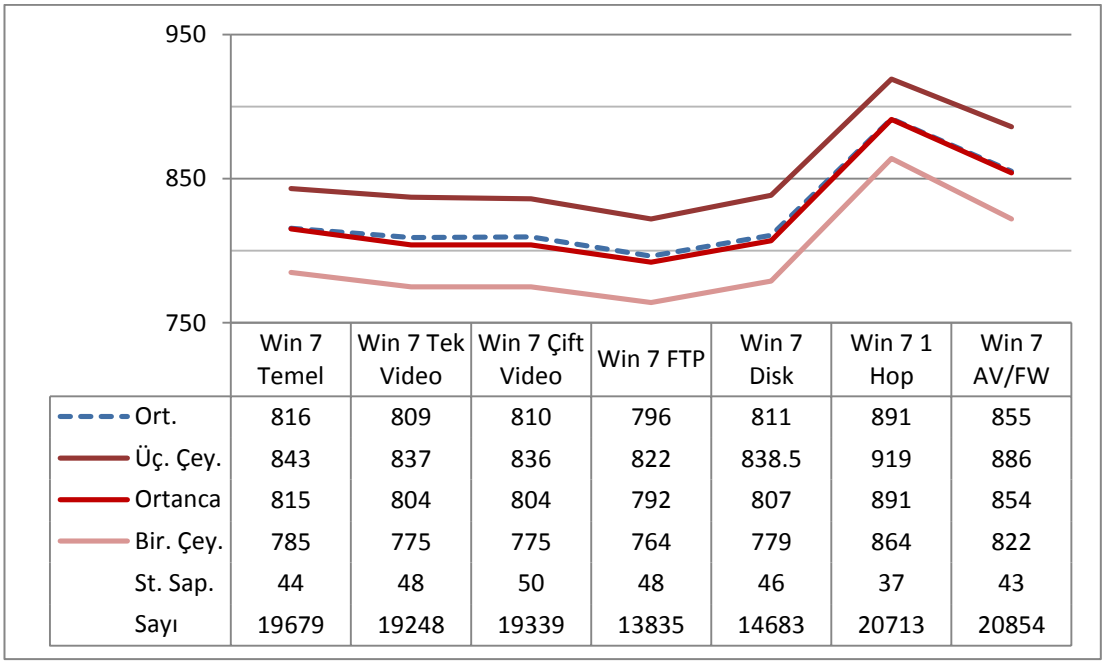
Tablo 3.19. Windows 7 Professional x86 TCP Yanıt Süreleri Histogram (µs)



Tablo 3.20. Windows 7 Pro ve XP SP1 Testleri TCP Histogram Karşılaştırma (µs)

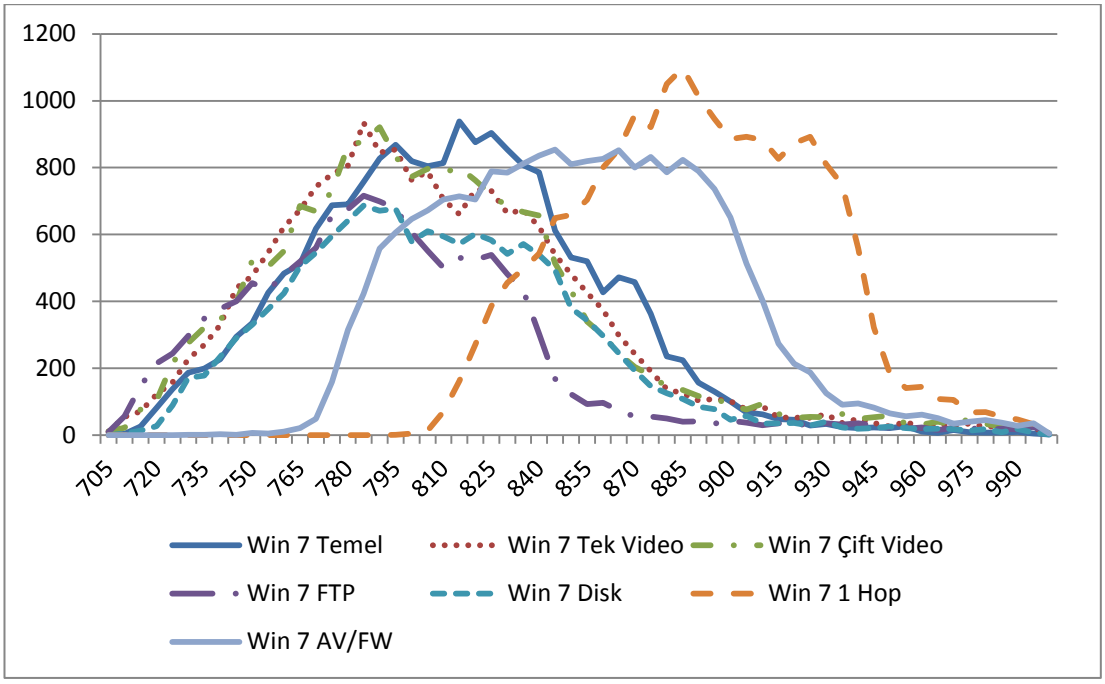


Tablo 3.21. Kablosuz Ağda Windows 7 Pro x86 TCP Test Sonuçları (µs)



	Win 7 Temel	Win 7 Tek Video	Win 7 Çift Video	Win 7 FTP	Win 7 Disk	Win 7 1 Hop	Win 7 AV/FW
Ort.	816	809	810	796	811	891	855
Üç. Çey.	843	837	836	822	838.5	919	886
Ortanca	815	804	804	792	807	891	854
Bir. Çey.	785	775	775	764	779	864	822
St. Sap.	44	48	50	48	46	37	43
Sayı	19679	19248	19339	13835	14683	20713	20854

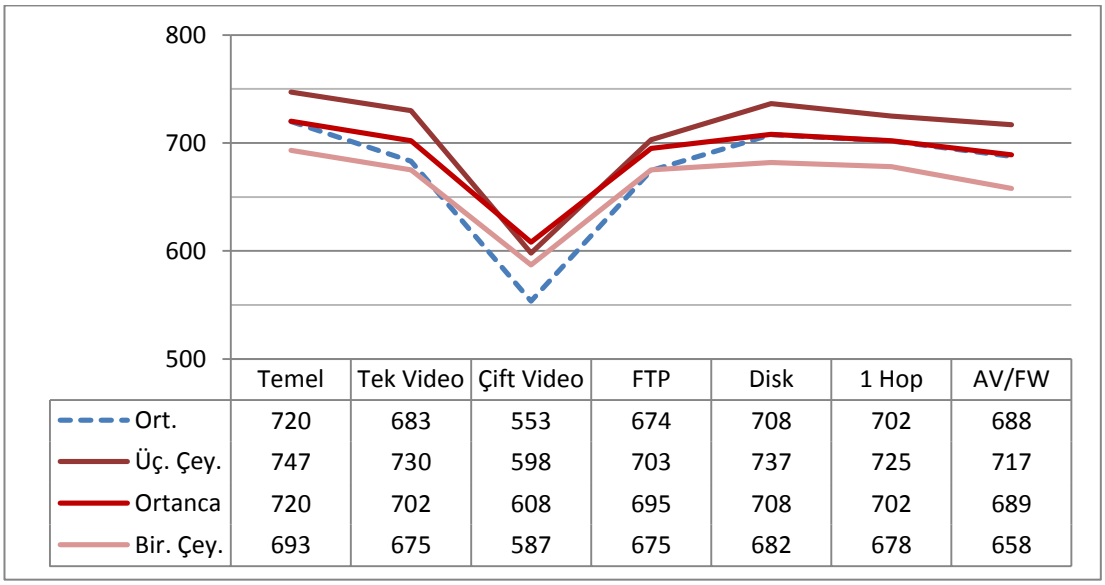
Tablo 3.22. Kablosuz Ağda Windows 7 TCP Yanıt Süreleri Histogram (µs)



Kablosuz ağ ortamında Windows 7 üzerinde yapılan ölçümlerin sonuçları Tablo 3.21'de ve histogramı Tablo 3.22'de gösterilmiştir. Bu sonuçlara bakarsak 1 hop ve güvenlik duvarı testleri hariç ayrılıkların kaybolduğu ve standart sapmaların kablosuz ağ ortamının etkisiyle yükseldiği görülmektedir. Ortalama olarak kablolu testlere göre 700 µs civarında ek gecikme ölçüldüğü sonucuna varabiliriz. Bu da DNS testleriyle paralellik göstermektedir. 1 hop testinin ortalama ek gecikme süresi 75-80 µs gibidir. Bu da kablosuz ağ ortamı üzerinden dahi bu ek değerin çok sapmadığını göstermektedir. Güvenlik duvarının ek gecikme süresi açısından etkisi ise Tablo 3.23'te verildiği gibi 40 µs olarak görünmektedir.

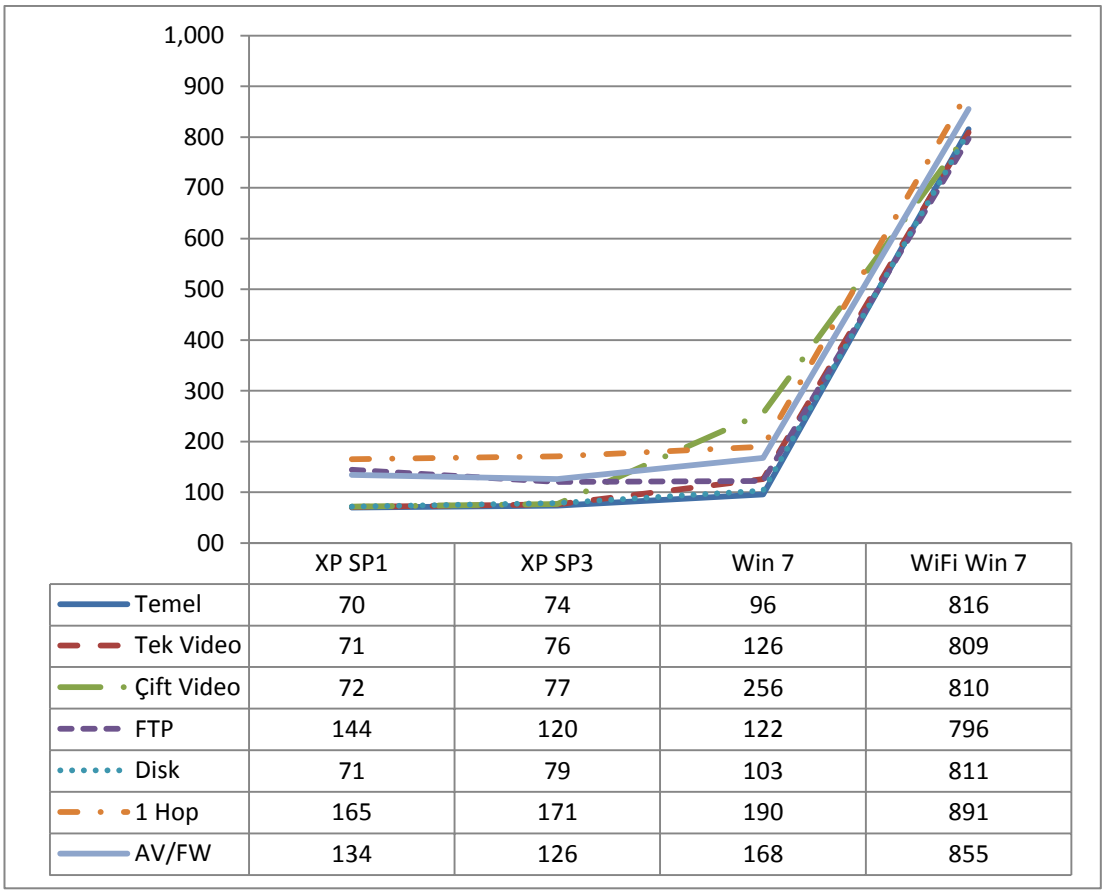
TCP testlerinin tüm ortam ve platformlar için topluca süre ortalamaları özeti Tablo 3.24'te verilmiştir. Genel olarak işletim sistemi modernleştikçe süreler yükselmekte, kablosuz ağ ortamının gecikmeyi arttırıcı etkisi görülmektedir. TCP testinde beklendiği gibi gecikmeler temel olarak ağ ortamına bağlı şekilde değişmekte, test edilen etkinlik çok önemli olmamaktadır. Bu nedenle etkinliğe bağlı anomali tespiti için DNS testinin daha uygun olduğu görülmektedir. TCP testi ağ ortamının etkilerinin azaltılması için yararlı olabilir.

Tablo 3.23. Kablosuz-Ethernet Ortamları Arasında TCP Süre Farkları (µs)



	Temel	Tek Video	Çift Video	FTP	Disk	1 Hop	AV/FW
Ort.	720	683	553	674	708	702	688
Üç. Çey.	747	730	598	703	737	725	717
Ortanca	720	702	608	695	708	702	689
Bir. Çey.	693	675	587	675	682	678	658

Tablo 3.24. Topluca TCP Test Süresi Ortalamaları (µs)



	XP SP1	XP SP3	Win 7	WiFi Win 7
Temel	70	74	96	816
Tek Video	71	76	126	809
Çift Video	72	77	256	810
FTP	144	120	122	796
Disk	71	79	103	811
1 Hop	165	171	190	891
AV/FW	134	126	168	855

4. DEĞİŞİM TESPİT YÖNTEMLERİ

Anomalilerin tespiti için DNS testlerinin daha uygun olduğunu 3. bölümde elde edilen sonuçlar göstermektedir. Bundan sonra sürekli gözlenen bir İstemci için bu test sonuçları arasında fark oluştuğu noktaların bulunması gerekmektedir. Bu amaçla literatürde daha önce çeşitli *akış verileri* (stream data) için geliştirilmiş ve test edilmiş *değişim tespit* (change detection) yöntemlerini inceledik [18,19,20,21,22,23].

İncelenen test yöntemleri arasından test edilen ve sonuçları burada verilecek olanlar,

- Wilcoxon Rank-sum Testi [24]
- Kolmogorov-Smirnov Testi [19,23]
- Kullback-Leibler Uzaklığına Dayalı Test [19,18]

olarak listelenebilir. Bu testler parametrik olmayan istatistiksel tabanlı testler olup, kendi yöntemimize uyguladığımız literatür çalışmalarında genellikle bir çatı yöntem içerisinde kullanılmaktadır.

4.1. Wilcoxon Rank-sum Testi

Wilcoxon Rank-sum testi parametrik olmayan sıralamaya dayalı bir istatistiksel testtir [24]. İlgili iki örneklemi veya aynı örnekten tekrarla alınmış iki ayrı ölçümü karşılaştırmak ve ortalamalarının ne kadar fark ettiğini ölçmek için kullanılır. Genellikle t-testi uygulanan veri kümelerinde normal dağılımın varsayılamayacağı durumlarda alternatif olarak kullanılır. Mann-Whitney-Wilcoxon olarak adlandırılan ve bağımsız farklı boyutlarda iki örneklem kullanılabilen türevi mevcuttur.

Örneğin A ve B adlı iki istatistiksel yığından n_A ve n_B adet örneklem aldığımızı düşünelim. Çeşitli X-ölçümlerinin dağılımının A'da ve B'de aynı olduğu hükümsüz önsavını (null hypothesis) test etmek istediğimizi farz edelim. Bu önsavı

$$H_0: A = B \qquad (4.1.1)$$

şeklinde yazabiliriz. Wilcoxon testinin tespit etmeye çalıştığı şey bu iki kümenin H_0 altında birbirinden ne kadar uzak olduğudur. Tek taraflı veya iki taraflı alternatifleri şu şekilde yazabiliriz:

$$H_0: A > B \tag{4.1.2}$$

$$H_0: A < B \tag{4.1.3}$$

$$H_0: A \neq B \tag{4.1.4}$$

Wilcoxon testi birleştirilen bu iki örneklemin sıralanarak oluşturulan sıra değerlerine dayalı bir testtir. Sıralamadan sonra bulunan sıra değerleri her iki örneklem için de toplanarak w_A ve w_B değerleri elde edilir. Daha sonra sırasıyla (4.1.2), (4.1.3) ve (4.1.4) için,

$$P - değeri = Pr(W_A \geq w_A) \tag{4.1.5}$$

$$P - değeri = Pr(W_A \leq w_A) \tag{4.1.6}$$

$$P - değeri = 2\,Pr(W_A \geq w_A)\ \ ya\ da\ \ 2\,Pr(W_A \leq w_A) \tag{4.1.7}$$

(4.1.4) denkleminde w_A'nın yakın olduğu kuyruğa göre iki denklemden biri seçilir. Bu denklemlerde W_A ile ifade edilen değer, A'dan yapılan rastgele gözlemlerin sıra toplamlarını ifade eden rastgele değişkendir. Bu olasılık değerleri bilgisayarda hesaplanarak çeşitli tablolar oluşturulmuştur. Yüksek değerler içinse yaklaşık değeri bir formül aracılığıyla bulunabilmektedir. Wilcoxon Rank-sum testi [0-1] aralığında bir olasılık değeri verir.

Bu test kullanılarak yaptığımız tüm DNS ölçümleri birbirleriyle karşılıklı olarak % 95 güven aralığı ile Wilcoxon Rank-sum testine tabi tutulmuş ve hesaplanan değerler çok büyük boyutta bir tablo ortaya çıkardığı için örnek sonuçlar Tablo 4.1'de ve hata oranları Tablo 4.2'de gösterilmiştir. Bu hata oranlarında Wilcoxon testi istediğimiz doğrulukta testler arasında ayrım yapamamaktadır. Bu nedenle Wilcoxon testinin akış verisi üzerinde denenmesine gerek yoktur.

Tablo 4.1. DNS Ölçümleri için Wilcoxon Testlerinin Örnek Sonuçları (p-değeri)

			XP SP1							
			Disk				1 Hop			
			1	2	3	4	1	2	3	4
XP SP1	Video x 1	1	0,000	0,005	0,000	0,000	0,000	0,000	0,000	0,000
		2	0,000	0,000	0,442	0,000	0,000	0,000	0,000	0,000
		3	0,000	0,209	0,000	0,000	0,784	0,016	0,003	0,335
		4	0,000	0,000	0,029	0,000	0,000	0,000	0,000	0,000
	Video x 2	1	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		2	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		3	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		4	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
	FTP	1	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		2	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		3	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
		4	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
	Disk	1	-	0,000	0,000	0,000	0,000	0,000	0,001	0,000
		2	-	-	0,000	0,000	0,057	0,000	0,000	0,005
		3	-	-	-	0,000	0,000	0,000	0,000	0,000
		4	-	-	-	-	0,000	0,000	0,000	0,000
	1 Hop	1	-	-	-	-	-	0,199	0,402	0,839
		2	-	-	-	-	-	-	0,521	0,097
		3	-	-	-	-	-	-	-	0,269
		4	-	-	-	-	-	-	-	-

Tablo 4.2. Wilcoxon Testlerinin Hata Oranları

Ölçüm Sayısı	6216	**Yanlış Olumlu Oranı**	% 60,58
Olumlu	104	**Yanlış Olumsuz Oranı**	% 2,08
Olumsuz	6112	**Hassasiyet**	% 24,40

4.2. Kolmogorov-Smirnov (KS) Testi

Kolmogorov-Smirnov testi, iki örneklemin (A ve B) aynı yığından olup olmadığı anlamak amacıyla karşılaştırmak için kullanılan, parametrik olmayan bir istatistiksel testtir. Bu test örneklemlerin deneysel olasılık dağılımları arasındaki uzaklığı ölçer [23,25].

Bağımsız ve eşit dağılmış n adet gözlemi içeren X için deneysel olasılık dağılımı aşağıdaki şekilde tanımlanabilir:

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} I_{X_i \leq x} \qquad (4.2.1)$$

$F_{1,n}$ ve $F_{2,n'}$ örneklemlerin deneysel olasılık dağılımı olmak üzere KS istatistiği aşağıdaki şekilde tanımlanır:

$$D_{n,n'} = \sup_x \left| F_{1,n}(x) - F_{2,n'}(x) \right| \qquad (4.2.2)$$

Burada *sup x* işlevi uzaklıklar kümesinin *supremum*udur. $\Pr(K \leq K_\alpha) = 1 - \alpha$ olmak üzere eğer aşağıdaki ifade doğruysa H_0: $A = B$ hükümsüz önsavı reddedilir:

$$\sqrt{\frac{n\,n'}{n+n'}} D_{n,n'} > K_\alpha \qquad (4.2.3)$$

Bu testle ölçülen DNS değerleri karşılıklı olarak % 95 güven aralığı ile MATLAB işlevi kullanılarak teste tabi tutulmuştur. Buradan alınan sonuçlar için hata oranları Tablo 4.3'te gösterilmiştir. Bu hata oranlarına göre KS testi de istediğimiz hassasiyette çalışmamaktadır. Bu nedenle akış verisi üzerinde testler yapılmamıştır.

Tablo 4.3. Kolmogorov-Smirnov Testlerinin Hata Oranları

Ölçüm Sayısı	6216	**Yanlış Olumlu Oranı**	% 0
Olumlu	16	**Yanlış Olumsuz Oranı**	% 2,45
Olumsuz	6200	**Hassasiyet**	% 9,52

4.3. Kullback-Leibler (KL) Uzaklığına Dayalı Test

Toplanan DNS ölçümlerden oluşturulan bir akış verisinin değişik testler arasındaki değişim noktalarının tespiti için [19] ve [18] nolu makalelerde yer alan yöntem Java programlama diliyle gerçekleştirilmiştir.

Bu test, ister tek boyutlu ister çok boyutlu olsun büyük veri akışı kümelerinde, altta yatan dağılım bilinmeden verinin değişim gösterdiği noktaların tespiti için geliştirilmiştir. Dağılımlar arasındaki uzaklığı ölçmek için *göreceli entropi* olarak da bilinen *KL-uzaklığı*nı kullanır.

Önerilmiş olan yöntem, "değişim"in uygulayıcı tarafından tanımlanmasına izin verecek genel bir yapıya sahiptir. Çok büyük veri kümelerine ve veri akışı yapılandırmalarına uyumludur, ayrıca çok-boyutlu verilerle de çalışabilmektedir. İstatistiksel temellere dayandığı için değişimin bu açıdan çözümlenmesine açıktır. Aynı zamanda da veri üzerinde dağılım açısından bir varsayım yapılamadığı durumlarda çalışabilmektedir.

KL-uzaklığının seçilmesi için bazı önemli gerekçeler şunlar olarak verilmiştir [19]:

- Eğer bir veri için birtakım dağılımlar arasından uygun bir dağılım seçilmek isteniyorsa, bunun için en uygun seçim aynı zamanda gerçek dağılımla olan KL-uzaklığını en aza indiren seçimdir.
- KL-uzaklığı, standart olarak kullanılan t-testi, ki-kare testi, Kulldorff uzamsal tarama istatistiği gibi fark testlerinin genelleştirilmiş halidir.
- p ve q dağılımları arasında ayrım tespit etmek isteyen bir en iyi sınıflandırıcının yanlış olumlu (veya yanlış olumsuz) oranı, p'den q'ya KL-uzaklığının üstel ifadesiyle orantılı bir limite sahiptir.

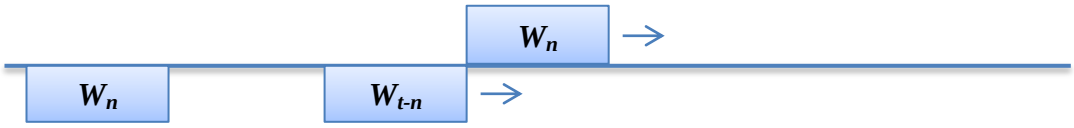
Dağılımlar için KL-uzaklığı R_n'deki öklit-uzaklığının simplex üzerindeki karşılığı gibi bir anlam ifade eder. Bu nedenle bu uzaklığa bir anlam atfetmemize izin verir. "Değişim"in tanımını verinin temsil biçiminden ayırabilir ve *tipler kuramı*nı (theory of types) kullanabiliriz.

KL-uzaklığı hesaplama açısından görece az maliyetle çok boyutlu verilere de uygulanabilmektedir. Bu özellik sıra tabanlı Wilcoxon ve Kolmogorov-Smirnov gibi testlerde mevcut değildir.

Önerilmiş olan yöntemde istatistiksel açıdan anlamlılığı ölçebilmek için güven aralığının tanımlanması amacıyla "*önyükleme yöntemi*" (bootstrap method) adı verilen bir yöntemi kullanmaktadır [26]. Bu yöntemle, verinin kendisi üzerinden (seçilen veriyi ister tekrar dâhil ederek ya da etmeyerek) tekrar eden biçimde yeniden örneklemeyle veri üzerindeki bir ölçümün anlamlı olup olmadığının tespit edilebilmesi sağlanmaktadır. Küçük veri kümelerinden istatistiksel olarak güçlü çıkarımlar yapılabilmesine olanak sağlar.

4.3.1. Yöntem

$x_1, x_2, \ldots$ bir nesne akışı olsun. Burada x_i'yi R^d'de bir nokta olarak varsayalım. $W_{i,n}$ olarak tanımladığımız pencere x_i ile biten n adetlik nokta sırasını ifade eder. Ölçeceğimiz uzaklıklar W_t ve $W_{t'}$ pencerelerinden oluşturulan dağılımlar arasında olacaktır.



Şekil 4.1. Kayan ve sabit-kayan pencereler

Temel olarak iki kayan pencere modeli kullanılacaktır. Bunlar:

- *Kayan pencereler modeli:* Aralarındaki uzaklığın ölçüleceği pencereler W_t ve W_{t-n} olacak şekilde kaymaktadır. Bu model şu anda olan değişimi daha iyi ölçebilmektedir. (Burada *t* zamanı göstermektedir.)
- *Sabit-kayan pencereler modeli:* Bir adet W_n sabit penceresi ile W_t kayan penceresi arasındaki uzaklık ölçülmektedir. Bu modelde ise zaman içinde biriken değişim daha iyi ölçülebilmektedir.

Her pencere bir deneysel olasılık dağılımı, F_t tanımlamaktadır. $d_t = d(F_t, F_{t'})$ olarak ifade edilen değer, F_t'den $F_{t'}$'ne olan KL-uzaklığını belirtmektedir (Tanım 4.1).

Tanım 4.1. İki olasılık yoğunluk fonksiyonu *p(x)* ve *q(x)* arasındaki *göreceli entropi* ya da *Kullback-Leibler uzaklığı* aşağıdaki şekilde tanımlanır:

$$D(p||q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}$$

KL-uzaklığı olasılık yoğunluk fonksiyonları üzerinden tanımlanmıştır. Bu nedenle akıştan alınan noktaların dağılımlara dönüştürülmesi gerekir. Bu noktada *tipler kuramı* (theory of types) kullanılmaktadır. $w = \{a_1, a_2, \ldots a_n\}$ sonlu $\mathcal{A}$ alfabesinden harflerin bir çoklu kümesi olsun. *w*'nun tipi P_w $\mathcal{A}$'daki her elemanın *w*'daki göreceli oranını temsil eden bir vektör olarak tanımlanır:

$$P_w(a) = \frac{N(a \mid w)}{n}$$

Böylece her *w* kümesi için bir P_w deneysel olasılık dağılımı tanımlanmış olur. Her bir küme için karşılık gelen deneysel dağılım hesaplanarak bu iki dağılım arasındaki uzaklık bulunabilir. *d*-boyutlu veri için "alfabe" verinin tutulacağı quad ağacının (quad tree) her bir yaprağından tanımlanan harflerden oluşacaktır. Bu şekilde yapılan hesaplamada ufak bir sorun $q = 0$ olduğunda p/q oranı tanımsız çıkmaktadır. Bunun için Krichevsky ve Trofimov tarafından önerilen düzeltmeyle $P_w(a)$ aşağıdaki şekilde güncellenir:

$$P_w(a) = \frac{N(a \mid w) + 0{,}5}{n + |\mathcal{A}|/2}$$

W_1 ve W_2 olarak verilen iki pencere ve bunlara karşılık quad ağacı yapraklarından oluşturulan w_1, w_2 çoklu kümeleri için W_1'den W_2'ye olan uzaklık şu şekilde bulunabilir:

$$D(W_1||W_2) = \sum_{a \in \mathcal{A}} P_{w_1}(a)\ log \frac{P_{w_1}(a)}{P_{w_2}(a)}$$

Elimizdeki pencerelerden elde ettiğimiz dağılımlar arasındaki farkı test etmek için hükümsüz önsav aşağıdaki şekilde oluşturulmaktadır:

$$H_0 : F_t = F_{t'} \qquad (4.3.1.1)$$

Bundan sonra H_0'ın doğru olduğu durumda bir d_t değerinin ölçülme olasılığının hesaplanması gerekmektedir. Bu noktada önyükleme yöntemi (bootstrap method) olarak adlandırılan bir yöntemi kullanacağız. Bu yöntem bir test istatistiğinin anlamlılığını belirlemek, yanlılığı (bias) ortadan kaldırmak ve güven aralıklarını geliştirmek için kullanılan bir yöntemdir. Bu yöntemle bir test istatistiğinin standart hatası, yanlılığı ve güven aralıkları tahmin edilebilir.

Veri değişimini belirlemek için bir önsav testinde hükümsüz önsav iki F ve G dağılımının denk olup olmadığını sorgular:

$$H_0: \mathrm{F} = \mathrm{G}$$

Bir gözlem yapılarak, $\hat{F}$ ve $\hat{G}$, F ve G'nin deneysel dağılımları olmak üzere $\hat{d} = D(\hat{F}||\hat{G})$ hesaplandığında gözlemin *ulaşılabilir anlamlılık seviyesi* (achievable significance level - ASL), $\hat{d}^*$ H_0 altında d'yi ölçen bir rastgele değişken olmak üzere,

$$\Pr_{H_0}(\hat{d}^* \geq \hat{d})$$

üzerinden tanımlanır. Burada $\hat{d}^* = 0$ demek hükümsüz önsava denk bir ifadedir. Dolayısıyla eğer $1-\propto$ olasılıkla $\hat{d}$ değerinin içinde yer alacağı [0, d_{hi}] aralığını tanımlarsak, bu $\propto$ seviyesinde bir ASL anlamına gelecektir. Bu yönteme *yüzdelik yöntemi* (percentile method) denir.

Önyükleme işlemi şu şekilde gerçekleştirilmektedir: P'den ölçülen $\hat{P}$ deneysel dağılımı verildiğinde, bu dağılımdan S_1, S_2, ... S_k olmak üzere k adet kümeyi örnekliyoruz. İlk n öğe olan S_{i1}'i F dağılımından geliyor gibi, kalanlara da G dağılımından geliyor gibi düşünüyoruz. Buradan önyükleme tahminleri olan $\hat{d}_i = D(S_{i1}||S_{i2})$ değerleri hesaplanır. İstenilen ASL seviyesi olan $\propto$ değerine göre bu önyükleme tahminlerinin $(1-\propto) - yüzdeliği$ d_{hi} olarak belirlenir. Daha sonra bu tahminlerden (d_{hi}, ∞) kritik bölgesini oluşturabiliriz. Eğer d_t bu bölgeye düşerse, H_0'ın geçersiz olduğuna karar verilecektir. Daha sağlam bir değişim tespit yöntemi kurabilmek için değişim sinyali sadece art arda γn defa d_{hi}'dan büyük uzaklık görüldüğünde verilecektir. Böylece sadece uzun süren değişimler için sinyal üretilmektedir. Burada γ değeri *sürerlik katsayısı* olarak belirlenmektedir.

Önyükleme yöntemi için yapılan deneylere göre yaklaşık 500-100 örnek iyi çalışan değerler üretmektedir [19].

Değişim tespit algoritması aşağıda verilmiştir:

Algoritma 4.1. Değişim Tespit Algoritması

$t \leftarrow 2n$;
$t' \leftarrow n$;
W_t *ve* $W_{t'}$ *pencerelerini oluştur*;
$d_t = d(F_t, F_{t'})$ *değerini hesapla*;
$\hat{d}_i, i = 1, \ldots, k$ *önyükleme değerlerini ve kritik bölgeyi* (d_{hi}, ∞) *hesapla*;
$c \leftarrow 0$;
while *akış sonuna gelinmediyse* ***do***
  if $d_t > d_{hi}$ ***then***
    $c \leftarrow c + 1$;
    if $c \geq \gamma n$ ***then***
      değişim sinyali ver;
      başlangıcı başla;
    end if
  else
    $c \leftarrow 0$;
  end if
  W_t *penceresini güncelle* (*gerekirse* $W_{t'}$ *penceresini de*);
  d_t *değerini güncelle*;
end while

"*Tipleri*" belirlemek için veri uzayını hücrelere bölen bir *alan-parçalama* (space partitioning) şemasına gereksinim duyuyoruz. Bunun için hem boyutla hem de veri sayısıyla iyi bir şekilde ölçeklenebilen bir veri yapısına gereksinimimiz vardır. Quad ağacı [27] tarafından üretilen hücreler yüksek boyutlarda iyi ölçeklenememekte, k-d-ağacı [28] tarafından üretilen hücrelerse sayıyla iyi ölçeklenememektedir. Bu

nedenle bu iki veri yapısının özelliklerini birleştiren bir yapıyı oluşturmamız gerekiyor. Yöntemde önerilen yapı türü kdq-ağacı olarak adlandırılmıştır [19].

İki boyutta bu yapıyı tanımlarsak: Bir kdq-ağacı her bir düğümünün bir kutuyla ilişkili olduğu ikili bir ağaçtır. Kök *v* ile ilişkili kutu tüm alanı temsil eder. Daha sonra bu alan öncelikle dikey sonra yatay bu şekilde değişen biçimlerde merkezden ikiye bölüne bölüne ağaç oluşturulur. Özyineleme bir kutudaki öğe sayısı τ'nun altına inerse veya kutunun tüm kenarları δ değerinin altına ulaşmışsa sonlandırılır. Buradaki bu değerler kullanıcı tarafından belirlenir [19].

N noktadan ve d boyuttan oluşan bir kdq-ağacının özellikleri şöyle verilebilir:

- En fazla $O(dn \log(\frac{1}{\delta})/\tau)$ mertebesinde düğümden oluşur.
- Yüksekliği en fazla $O\left(d \log\left(\frac{1}{\delta}\right)\right)$ mertebesindedir.
- $O\left(d \log\left(\frac{1}{\delta}\right)\right)$ mertebesinde bir zamanda oluşturulabilir.
- Herhangi bir hücrenin en boy oranı (aspect-ratio) en fazla 2'dir.

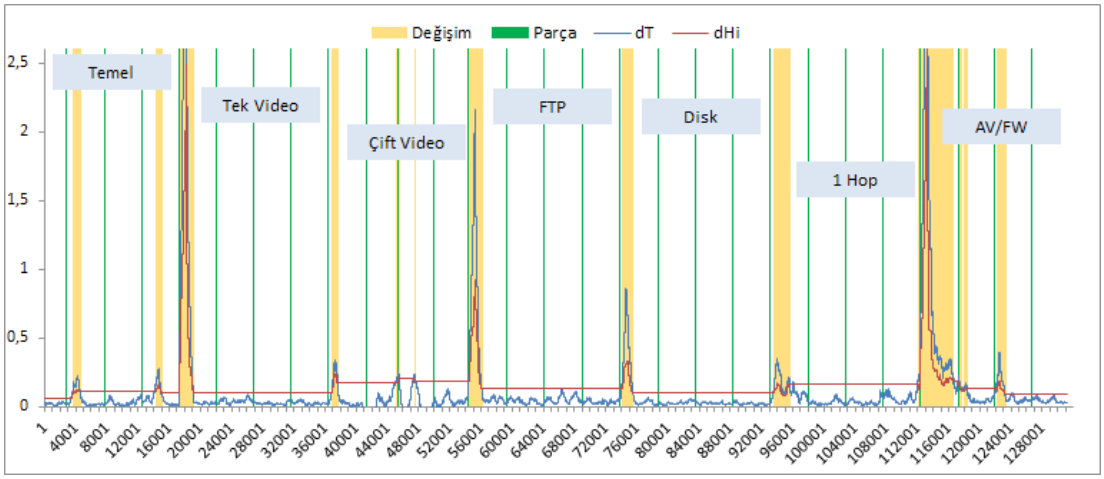
Bu nedenle kdq-ağacı verinin sayısıyla ve boyutuyla doğrusal olarak ölçeklenmekte olduğunu görebiliriz [19].

İlk pencere W_1 üzerinden kdq-ağacı oluşturulmakta, daha sonra W_1 ve W_2 deneysel dağılımlarını çıkarmak için bu yapı kullanılmaktadır. Bir değişiklik sinyali üretildiğinde ise tüm yapı yeniden oluşturulmaktadır. Aynı yapı önyükleme değerlerini hesaplarken de kullanılmaktadır.

4.3.2. Testler

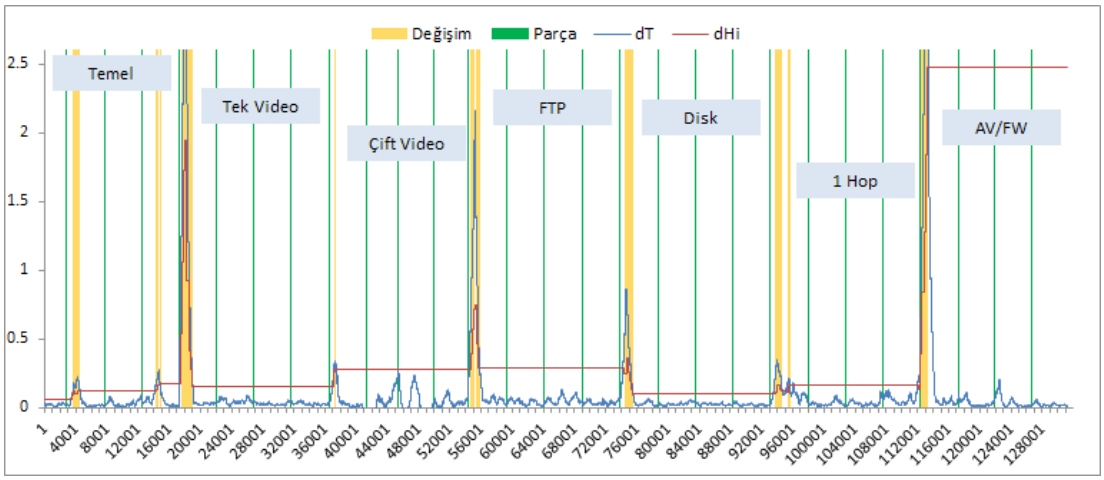
[19] nolu makalede önerilen yöntemin gerçekleştirimi Java programlama dilinde uygulanmış ve burada da makalede tanımlanan tek boyutlu kdq-ağacı kullanılmıştır. Ölçümlerden elde edilen verilerden bu ağaç çeşitli parametreler ve pencere boyları ile test edilmiş ve aşağıdaki sonuçlar alınmıştır. Tüm testler için aynı olan parametre değerleri: $\tau = 100$, $\delta = 50$.

Çizelge 4.4. Windows XP SP1 DNS Ölçümleri için KL-testi (1)



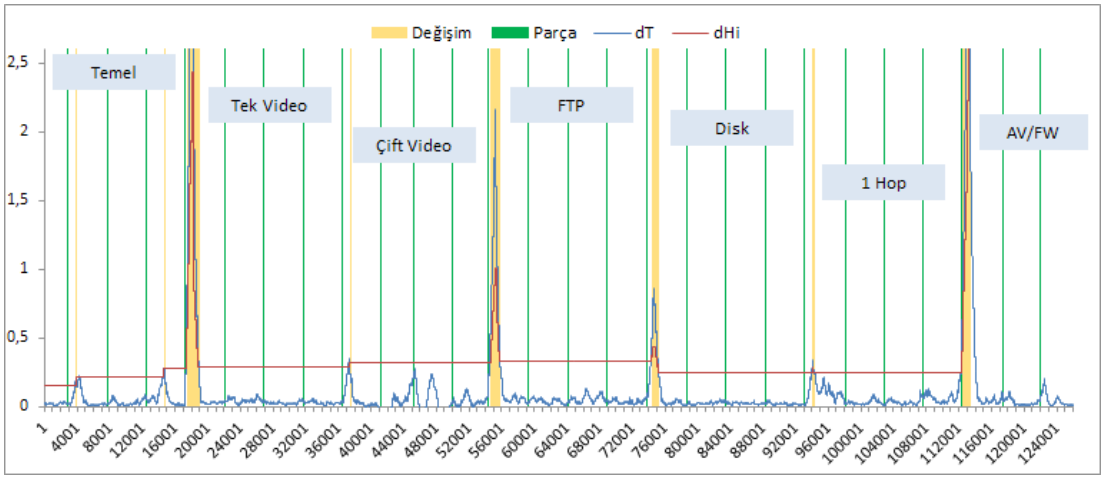
Windows XP SP1 için pencere boyutu *(n)* 1000, önyükleme örnek sayısı *(k)* 500, ASL *(α)* 0,001 ve PF *(γ)* 0,1 olmak üzere yapılan KL-testinin sonuçları Çizelge 4.4'te verilmiştir. Burada 4 parçada bir test değişmekte ve bu noktalarda değişimin algılanması beklenmektedir. Görüldüğü gibi testler arasındaki değişimler algılanmış ve fazladan yanlış olumlu sinyaller verilmiştir, özellikle güvenlik duvarı testi bu konuda değişken bir dağılım göstermektedir.

Çizelge 4.5. Windows XP SP1 DNS Ölçümleri için KL-testi (2)



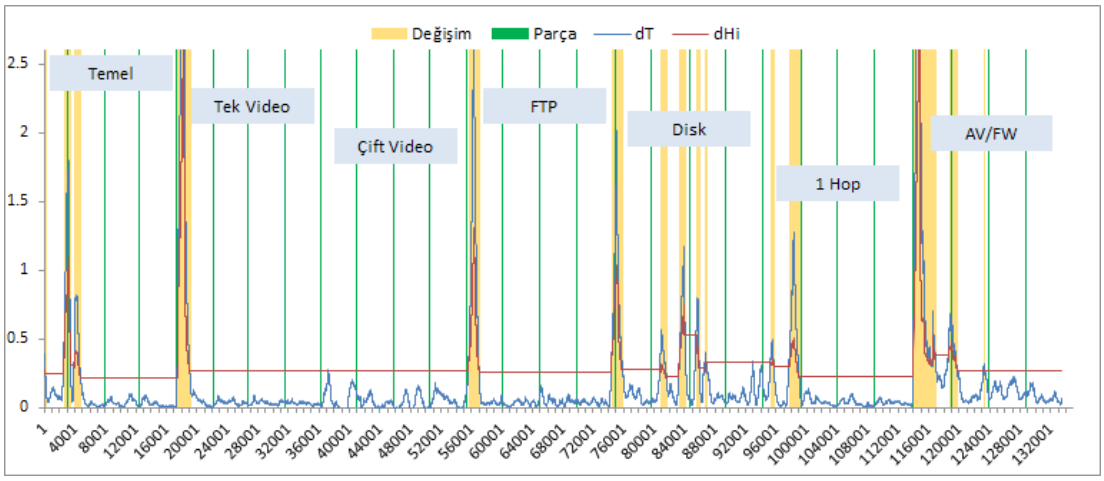
Diğer parametreler sabit tutularak PF değeri 0,2'ye çıkartıldığında (Çizelge 4.5).

Çizelge 4.6. Windows XP SP1 DNS Ölçümleri için KL-testi (3)



PF 0,1 iken d_{Hi} değerlerine fazladan 0,1 eklendiğinde ortaya çıkan test sonuçları Çizelge 4.6'da gösterilmektedir. Bu sonuçlarda temel testler arasında 2 yanlış olumlu sinyal dışında istenmeyen bir durum bulunmamaktadır. En zayıf geçişlerin aynı tür testler olan tek video ve çift video testleri arasında olduğu ve yine dağılımları birbirine benzer çıkan disk ve 1 hop testlerinin de zayıf geçiş gösterdiğini görüyoruz.

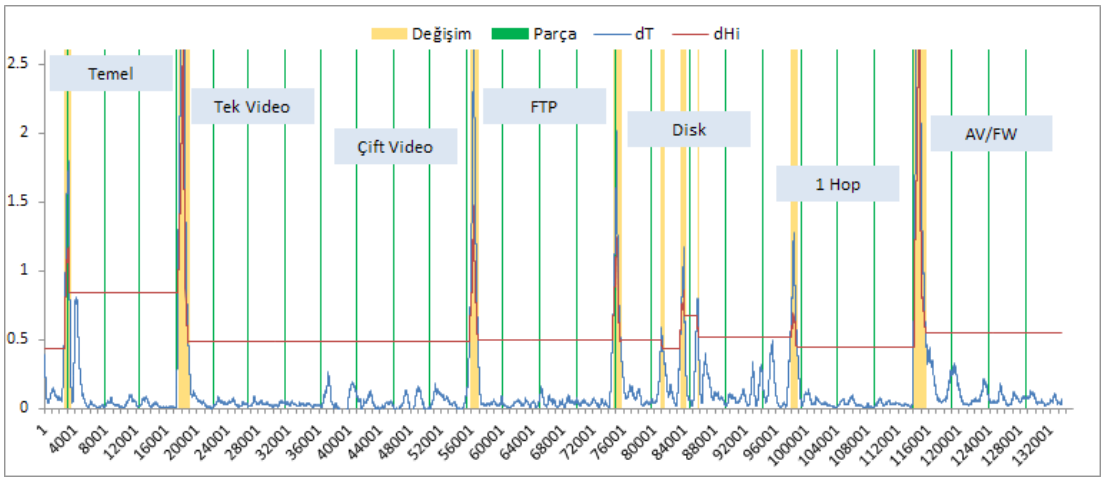
Çizelge 4.7. Windows XP SP3 DNS Ölçümleri için KL-testi (1)



Windows XP SP3 için pencere boyutu 1000, önyükleme örnek sayısı 500, ASL 0,001 ve PF 0,05 olmak üzere yapılan KL-testinin sonuçları Çizelge 4.7'de verilmiştir. Burada temel, disk, 1 hop ve beklendiği gibi güvenlik duvarı testlerinde test parçaları

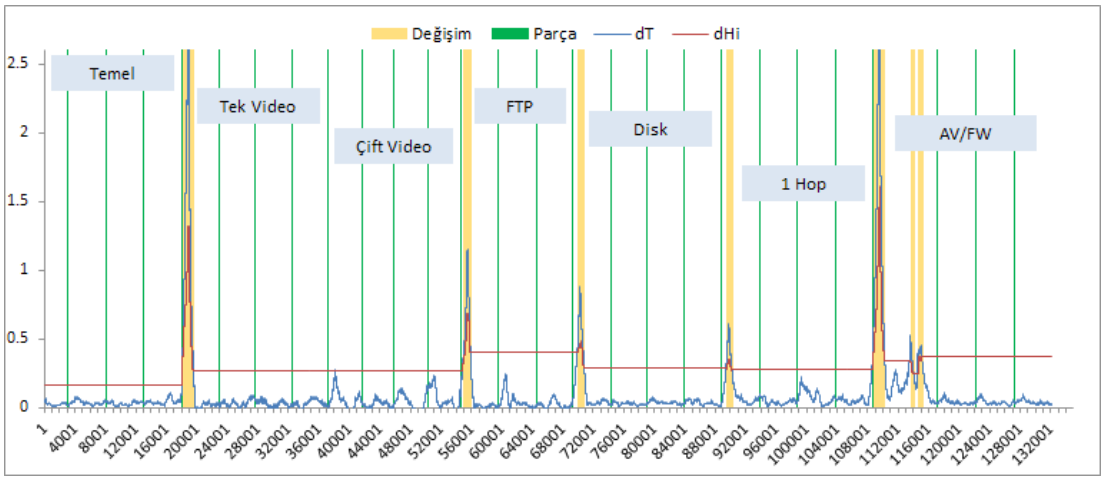
arasında değişimler algılanmıştır. Video testleri arasındaki geçiş ise çok düşük şiddettedir.

Çizelge 4.8. Windows XP SP3 DNS Ölçümleri için KL-testi (2)



PF değeri 0,05'te kalmak üzere fazladan d_{Hi} üzerine 0,25 eklenerek yapılan test sonuçları Çizelge 4.8'de gösterilmektedir.

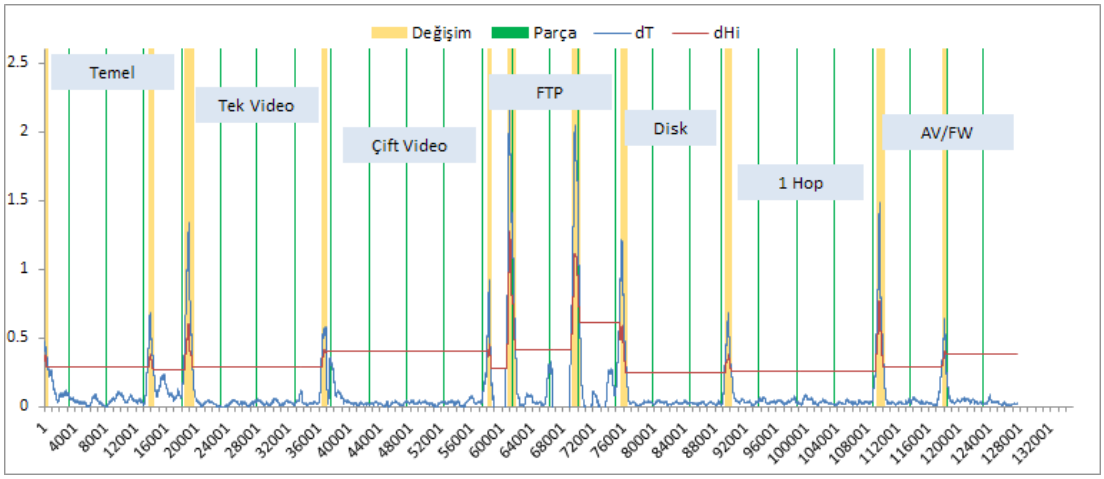
Çizelge 4.9. Windows 7 DNS Ölçümleri için KL-testi (1)



Windows 7 için pencere boyutu 1000, önyükleme örnek sayısı 500, ASL 0,001 ve PF 0,1 olmak üzere d_{hi} değerlerine fazladan 0,1 eklenerek yapılan KL-testinin sonuçları Çizelge 4.9'da verilmiştir. Burada da benzer bir şekilde video testleri arasında değişim

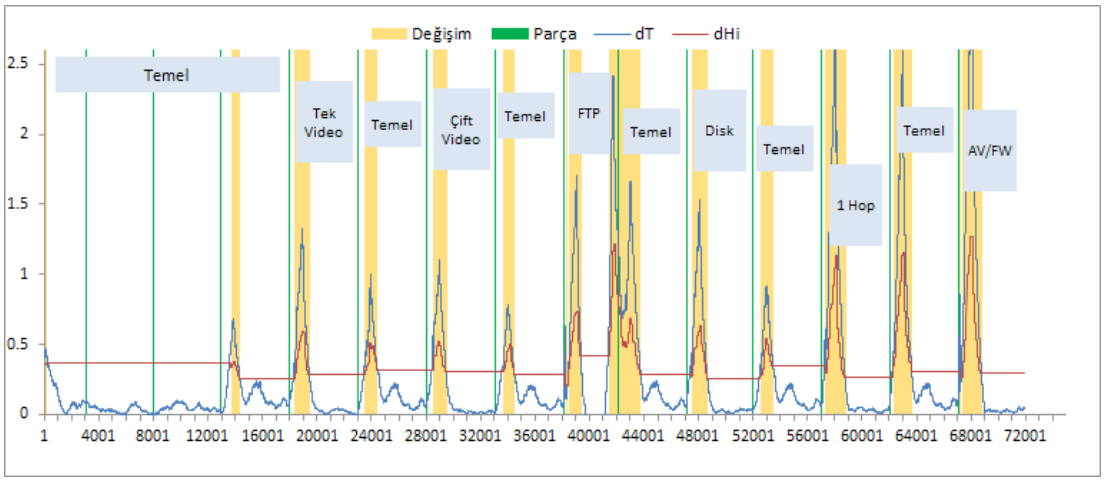
ayırt edilememekte, Güvenlik duvarı testinde ise düşük şiddette de olsa fazladan değişimler tespit edilmektedir.

Çizelge 4.10. Windows 7 Kablosuz Ağ DNS Ölçümleri için KL-testi (1)



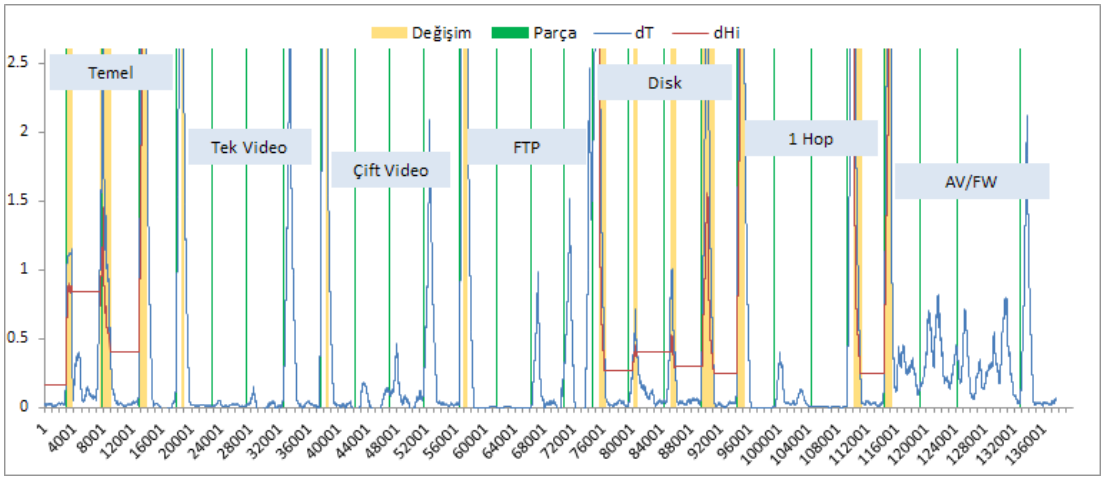
Kablosuz ağda Windows 7 için yapılan ölçümlerle pencere boyutu 1000, önyükleme örnek sayısı 500, ASL 0,001 ve PF 0,1 olmak üzere d_{hi} değerlerine fazladan 0,1 eklenerek yapılan KL-testinin sonuçları Çizelge 4.10'da verilmiştir. Kablosuz ağ üzerinde beklendiği gibi daha fazla test dağılımları birbirine yakın seyrettiği için yanlış sinyaller üretilmiştir. Özellikle FTP testi yoğun ağ kullanımı nedeniyle bundan etkilenmiştir. Bu testlerin her birinin temel teste göre karşılaştırması da Çizelge 4.11'de gösterilmiştir.

Çizelge 4.11. Windows 7 Kablosuz Ağ DNS Ölçümleri Temel Teste göre KL-testi (1)



Tüm test gruplarının her bir test için art arda (Windows XP SP1, SP3, Windows 7 ve Windows 7 WiFi olarak sıralanmıştır) DNS ölçümlerinin sonuçları Çizelge 4.12'de gösterilmiştir. Buna göre temel test içerisinde sistem değişimi sinyal üretirken, Video testlerinde sadece kablosuz ağa geçerken değişim tespit edilebilmektedir. FTP testlerinde Windows 7 (hem Ethernet hem kablosuz ağ için) yüksek değerler üretmekte, disk testinde tüm sistemler sinyal üretmekte, 1 hop testi beklendiği gibi sistemler arasında kablosuz ağ dışında fark etmemektedir. Güvenlik duvarı testinde ise kablosuz ağdaki değişim diğerlerinden yüksek olmakla birlikte tüm sistemler için tespit edilen değerler test içi değerlerden ayırt edilememektedir. Dolayısıyla temel test ve disk testinde sistemlerin değişimi sorunsuz olarak algılanabilmekte, diğer testlerde ya hiç sinyal üretilmemekte yani sistemler arası fark etmemektedir.

Çizelge 4.12. Tüm sistemlerin art arda DNS ölçümü KL-testi sonuçları



5. SONUÇLAR

Çalışmamızda ağ üzerinden özel paket etkileşimlerinden çıkartabildiğimiz yanıt verme süresindeki gecikmeleri ölçerek makinelerde meydana gelebilecek anomalilerin tespit edilmesi amaçlanmıştır. Bunlar, donanımsal ve yazılımsal değişiklikler, bozukluklar, sürücü ve donanım yazılımı hataları, kötü amaçlı yazılımların bulaşması gibi değişik etkenler olabilir. Bu etkenler genel olarak sistem kaynaklarını tüketerek ağ başarımında azalmaya yol açmakta ve yanıt süreleri değişmektedir. Yüksek MİB ve G/Ç kullanımı temel etki yapan unsurdur. Bu nedenle bu etkinliğin ölçülmesi ve çeşitli testlerle birbiriyle kıyaslanması gerekmektedir.

Bir bilgisayarın uzaktan, sürekli olarak bu anomaliler açısından izlenebilmesi için sık kullanılan ve dışarıdan ölçülebilecek ağ paketleri düşünüldüğünde en uygun adaylar olarak TCP 3-yönlü el sıkışma ve DNS sorgusuna sunucudan gelen yanıta karşılık bu adresin ziyaret edilmesi görülmüştür. Bu gecikme değerlerinin ağ üzerinden izleyici bir makine aracılığıyla yakalanıp saklanabileceği bir test ortamı oluşturulmuş ve önemli etkenlere göre bazı testler tasarlanmıştır. Temel olarak sistem değişiklikleri ve ağ ortamı değişiklikleri altında bu testler her grup için tekrarlanmış ve kaydedilmiştir. Bu testler için bir adres listesini İstemci makine tekrarlayarak ziyaret etmiş, Gözlemci makine üzerinden ağ trafiği saklanmıştır. Sonuçlar bu ağ protokollerini çözümleyip hesaplamaları yapan bir uygulama yazılarak çıkartılmış ve analiz edilmiştir.

Analiz edilen sonuçlara göre bu tür etkinliklerin beklendiği gibi ağ başarımı üzerinde etkileri dışarıdan ölçülebilecek düzeyde olmuş ve birçok testte testler arasındaki fark ayırt edilebilir düzeyde çıkmıştır. Sistem değişimi, yüksek MİB kullanımı, FTP ile yüksek ağ trafiği oluşturulması, disk üzerinde dosya kopyalama ile yüksek G/Ç etkinliği, aradaki İstemcinin atlanarak bir sonraki makineden ölçüm alınarak fazladan 1 hop üzerinden ölçüm, güvenlik duvarı ve anti-virüs uygulaması kurulması, ağ ortamının Ethernet ve kablosuz arasında değiştirilmesi ile bunların etkileri ölçülmüş ve sonuçlarda görülmüştür.

İzlenen makineden alınan veri akışı üzerinden değişimin tespit edilebilmesi amacıyla çeşitli değişim tespit yöntemleri incelenmiş ve KL-uzaklığına dayalı [19] nolu çalışmada ortaya konulmuş olan yöntemin en iyi sonuçları verdiği görülmüştür. Bu yöntemde önerilen veri yapısı ve algoritma gerçekleştirilerek sonuçlar bu sistemle incelenmiş ve birçok test arasında değişimin tespit edilebildiği ve yöntemin görece daha az noktada yanlış olumlu değişim sinyali verdiği görülmüştür. Bu yöntem kullanılarak test grupları arasındaki değişimler tespit edilebilmektedir.

Yapılan ölçümler ve incelenen yöntemler önerilen sistemin çalışabileceğini göstermektedir. Bu sistem daha da geliştirilerek ve gerçek kullanım koşullarına göre eniyileştirilerek ağ üzerinden anomali tespiti yapan bir ürün geliştirilebilir ve dışarıdan bir ağ içerisindeki makinelerin bu şekilde incelenebilmesi, ileri düzey kötü amaçlı yazılımların tespiti gibi önemli bir sorunun çözümüne katkı sağlayabilir. Ayrıca oluşabilecek diğer anomalilerden ağ yöneticileri haberdar edilerek bunlara karşı önlemler ve çözümler geliştirmeleri sağlanabilir.

KAYNAKLAR

[1] Memon, N., Sencar, H.T., Shanmugasundaram, K., 2009, Network-Based Infection Detection Using Host Slowdown, *Patent US20090126019*, ABD.

[2] Security Intelligence Report, Microsoft, 2011.

[3] Horn, D., "Timing Rootkits" erişim adresi: http://diablohorn.wordpress.com/2008/10/28/timing-rootkits/.

[4] Hoglund, G., Butler, J., Rootkits: Subverting the Windows Kernel, *Addison-Wesley*, 2005.

[5] Marx, A., Morgenstern, M., "Anti-Stealth Fighters: Testing for Rootkit Detection and Removal" erişim adresi: http://www.av-test.org/down/papers/2008-04_vb_rootkits.pdf.

[6] Wang, Y., Vo, B., Roussev, R., Verbowski, C., Johnson, A., Strider GhostBuster: Why It's A Bad Idea For Stealth Software To Hide Files, Microsoft Research, 2004.

[7] Wang, Y., Beck, D., Vo, B., Roussev, R., Verbowski, C., Detecting Stealth Software with Strider GhostBuster, Microsoft Research, 2005.

[8] "Alexa Top 1,000,000 Sites" erişim adresi: http://www.alexa.com/topsites, erişim tarihi: Nisan 2010.

[9] "Page Update Watcher" erişim adresi: http://sourceforge.net/projects/pwatcher/, erişim tarihi: Nisan 2010.

[10] "Microsoft MSDN Library - Internet Explorer Object" erişim adresi: http://msdn.microsoft.com/en-us/library/aa752084%28v=VS.85%29.aspx, erişim tarihi: Haziran 2010.

[11] "Wget for Windows" erişim adresi: http://gnuwin32.sourceforge.net/packages/wget.htm, erişim tarihi: Temmuz 2010.

[12] "TCPDUMP Man Page" erişim adresi: http://www.tcpdump.org/tcpdump_man.html, erişim tarihi: Nisan 2010.

[13] "jNetPcap Open Source Protocol Analysis Library Project" erişim adresi: http://jnetpcap.com/, erişim tarihi: Temmuz 2010.

[14] "KMPlayer Video Oynatıcı" erişim adresi: http://www.kmplayer.com/forums/.

[15] "Agnitum Outpost Security Suite Pro" erişim adresi: http://www.agnitum.com/products/security-suite/.

[16] "Microsoft TechNet - Next Generation TCP/IP Stack" erişim adresi: http://technet.microsoft.com/en-us/network/bb545475.

[17] "Next Generation TCP/IP Stack in Windows Vista and Windows Server 2008" erişim adresi: http://technet.microsoft.com/tr-tr/library/bb878108(en-us).aspx.

[18] Dasu, T., Krishnan, S., Lin, D., Venkatasubramanian, S., Yi, K., Change (Detection) You Can Believe in: Finding Distributional Shifts in Data Streams, Proceedings of the 8th International Symposium on Intelligent Data Analysis: Advances in Intelligent Data Analysis VIII (IDA '09), 21-34, 2009.

[19] Dasu, T., Krishnan, S., Venkatasubramanian, S., Yi, K., An Information-theoretic Approach to Detecting Changes in Multi-dimensional Data Streams, Proc. Symp. on the Interface of Statistics, Computing Science, and Applications, 2006.

[20] Kifer, D., Ben-David, S., Gehrke, J., Detecting Change in Data Streams, Proceedings of the Thirtieth international conference on Very large data bases (VLDB '04), vol. 30, 180-191, 2004.

[21] Kleinberg, J., Bursty and Hierarchical Structure in Streams, KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 91-101, 2002.

[22] Muthukrishnan, S., Berg, E., Wu, Y., Sequential Change Detection on Data Streams, ICDMW '07 Proceedings of the Seventh IEEE International Conference on Data Mining Workshops, 550-551, 2007.

[23] Sheskin, D., Handbook of Parametric and nonparametric statistical procedures, *Chapman and Hall/CRC*, 2000.

[24] Wilcoxon, F., Individual Comparisons by Ranking Methods, Biometrics Bulletin, 1(6), 80-83, 1945.

[25] Sheskin, D., Handbook of parametric and nonparametric statistical procedures, *Chapman & Hall/CRC*, 2000.

[26] Johnson, R.W., An Introduction to the Bootstrap, Teaching Statistics, 23(2), 49-54, 2001.

[27] Samet, H., Foundations of Multidimensional and Metric Data Structures, *Morgan Kaufmann*, 28-47, 2006.

[28] Samet, H., Foundations of Multidimensional and Metric Data Structures, *Morgan Kaufmann*, 48-89, 2006.

[29] "WinDump - TCPDump for Windows" erişim adresi: http://www.winpcap.org/windump/.

[30] "The Wilcoxon Rank-Sum Test" erişim adresi: http://www.stat.auckland.ac.nz/~wild/ChanceEnc/Ch10.wilcoxon.pdf.

[31] Ozonat, K., An Information-Theoretic Approach to Detecting Performance Anomalies and Changes for Large-scale Distributed Web Services, The 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2008, 522-531, Anchorage, Alaska, USA, 2008.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı	: ÜNLÜ, Seçkin Anıl
Uyruğu	: T.C.
Doğum Tarihi ve Yeri	: 01.01.1986
Medeni Hali	: Bekâr
Telefon	: 0 (312) 292 4290
Faks	: 0 (312) 292 4290
E-Posta	: sunlu@etu.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Y. Lisans	TOBB ETÜ Bilgisayar Mühendisliği	2011 (beklenen)
Lisans	TOBB ETÜ Bilgisayar Mühendisliği	2008

İş Deneyimi

Yıl	Yer	Görev
2008 – 2011	TOBB ETÜ	Ders Asistanlığı
2008 – 2008	Solveka Yazılım Ltd.	Yazılım Programlama
2007 – 2008	TOBB ETÜ	Yazılım Programlama
2006 – 2006	ASAŞ Ambalaj A.Ş.	Bilgi İşlem

Yabancı Dil

İngilizce (ileri seviye)

Yayınlar

1. Konferanslar

1. Unlu, S.A.; Bicakci, K., "NoTabNab: Protection Against The 'Tabnabbing Attack'," 5th APWG eCrime Researchers Summit, IEEE, October 18-20 2010, Dallas, Texas, USA