IPv6 GEÇİŞ YÖNTEMLERİNİN GÜVENLİK VE PERFORMANS ANALİZİ

Beyhan ÇALIŞKAN

YÜKSEK LİSANS TEZİ
YÖNETİM BİLİŞİM SİSTEMLERİ

GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ

ŞUBAT 2010
ANKARA

Beyhan ÇALIŞKAN tarafından hazırlanan IPv6 GEÇİŞ YÖNTEMLERİNİN GÜVENLİK VE PERFORMANS ANALİZİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Prof.Dr. Bilal GÜNEŞ
Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği ile Yönetim Bilişim Sistemleri Anabilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Başkan: : Prof. Dr. Ergün Kasap

Üye : Prof. Dr. Bilal GÜNEŞ

Üye : Doç. Dr. Ali Aydın Selçuk

Üye : _____

Üye : _____

Tarih : 15.../02.../2010.

Bu tez, Gazi Üniversitesi Bilişim Enstitüsü tez yazım kurallarına uygundur.

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.

Beyhan ÇALIŞKAN

IPv6 GEÇİŞ YÖNTEMLERİNİN

GÜVENLİK VE PERFORMANS ANALİZİ

(Yüksek Lisans Tezi)

Beyhan ÇALIŞKAN

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

Şubat 2010

ÖZET

IP (İnternet Protokolü), paket anahtarlamalı bilgisayar ağları arasındaki iletişim için tasarlanmıştır. Küresel ağ Internet'in genel dili olan bu protokol, 1981 yılında RFC 791 ile tanımlanmış ve standartlaşmıştır. Günümüzde yaygın olarak kullanılan IPv4 (IP versiyon 4), internetin hızlı gelişimi ile birlikte adres uzayını hızla tüketmektedir. IETF bu sorunu çözmek ve protokole yeni fonksiyonlar kazandırmak için 1990'ların başında çalışmalarına başlamış ve 1998 yılında RFC 2460 ile IPv6 standartlarını tanımlamıştır.

Yeni protokole geçiş bir akşamda gerçekleşebilecek bir süreç değildir. Bu nedenle farklı iki protokolün uzun süre birlikte çalışması kaçınılmazdır. Protokoller arasındaki geçiş sürecinin başarısı ise, IPv4 düğümlerinin IPv6 düğümleri ile sorunsuz iletişimine bağlıdır. Bu amaçla birçok IPv6 geçiş yöntemi geliştirilmiştir.

IPv6 geçiş yöntemlerinin güvenlik ve performans analizinin yapıldığı bu çalışmanın temel amacı, yeni protokolün uygulanması sırasında izlenecek yöntem veya yöntemler hakkında bir öngörüde bulunmaktır. Güvenlik,

performans ve uygulanabilirlik ölçütleri ışığında, yapılan çok sayıda araştırma ve gerçek ağ testleri sonucunda, küçük ağlardan çok büyük ağlara kadar IPv6 geçiş yöntemleri değerlendirilmiştir. Ayrıca, daha önce detaylı araştırması yapılmamış, IPv6 geçiş yöntemlerinin servis dışı bırakma saldırıları karşısındaki durumu, gerçek ağ ortamında test edilmiştir. Bu tez çalışmasına ait bir diğer önemli konu, bazı çevirici yöntemlerinde, tek nokta hatalarının giderilmesi için önerilen mekanizmadır. Hazırlanan bu çalışmada, sırasıyla; IPv6 geçiş yöntemleri ile ilgili yapılan bazı çalışmalar hakkında bilgi verilmiş, IPv6 mimarisine ilişkin başlık yapısı ve adresleme özellikleri irdelenmiştir. Ardından mevcut IPv6 geçiş yöntemleri incelenmiş ve bu yöntemlere ilişkin güvenlik ve performans konuları araştırılmıştır. Çalışmanın son bölümde, IPv6 geçiş sürecinde kullanılması önerilen mekanizmalar, elde edilen bulgular ışığında değerlendirilmiştir.

Bilim Kodu : 902.1.063
Anahtar Kelime : IPv6, güvenlik, performans, geçiş yöntemleri
Sayfa Adedi : 93
Tez Yöneticisi : Prof.Dr. Bilal GÜNEŞ

SECURITY AND PERFORMANCE ANALYSIS OF THE IPv6 TRANSITION METHODS

(M.Sc. Thesis)

Beyhan ÇALIŞKAN

GAZI UNIVERSITY

INFORMATICS INSTITUTE

February 2010

ABSTRACT

The Internet Protocol (IP) is designed for use in interconnected systems of packet-switched computer communication networks. It is the common language of the Internet which was defined and standardized in RFC 791 at 1981. Depending on explosive growth of the Internet, currently used IP version IPv4 (IP Version 4) is exhausting its address space. To remedy the foreseen address space limitation and to provide additional functionality, IETF started to work at the early 90's and published the IPv6 standards in RFC 2460 at 1998.

The migration to a new protocol will not be achieved overnight. Therefore, IPv4 and IPv6 will have to coexist for several years. The key to a successful IPv6 transition is compatibility and interoperability between IPv4 nodes and IPv6 nodes. For this purpose, many IPv6 transition mechanisms were developed.

This study aims to provide a foresight about the IPv6 transition mechanisms according to their security and performance analysis. Depending on many research and real network tests, IPv6 transition mechanisms were evaluated from the small networks to the large networks in the view of the security, performance and applicability metrics. Furthermore, IPv6 transition mechanisms were evaluated against the denial of service attacks at the real network testbed which was didn't analyzed yet. Also, a new mechanism was

proposed to protect some of the IPv6 translator approaches against the single point of failure. This study organized as follows; First section presents the related work about the IPv6 transition mechanisms and second section presents the IPv6 header and addressing architecture. In the next sections, security and performance analysis of the IPv6 transition mechanisms were discussed. And last section dedicated to conclusions.

Science Code : 902.1.063
Key Words : IPv6, performance, security, transition methodts
Page Number : 93
Adviser : Prof.Dr. Bilal GÜNEŞ

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarından dolayı danışmanım Prof. Dr. Bilal GÜNEŞ'e, kıymetli tecrübelerinden faydalandığım uzman araştırmacı Onur BEKTAŞ'a , ayrıca manevi destekleriyle beni hiçbir zaman yalnız bırakmayan ailem ve ULAKBİM'deki çalışma arkadaşlarıma teşekkürü bir borç bilirim.

İÇİNDEKİLER

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklama**
Bit	En küçük bilgi depolama birimi
Byte	8 bitlik bellek birimi
Mbit	10^6 bit
Ms	Saniyenin binde biri (10^{-3} saniye)

Kısaltmalar	**Açıklama**
IETF	Internet Engineering Task Force
RFC	Request For Comments
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

1. GİRİŞ

IP (İnternet Protokolü), paket anahtarlamalı bilgisayar ağları arasındaki iletişim için tasarlanmıştır. Küresel ağ Internet'in genel dili olan bu protokol, 1981 yılında RFC 791 ile tanımlanmış ve standartlaşmıştır. Günümüzde yaygın olarak kullanılan IPv4 (IP versiyon 4), internetin hızlı gelişimi ile birlikte adres uzayını hızla tüketmektedir. IETF tarafından oluşturulan Address Lifetime Expectation çalışma grubuna göre, 2011 yılına kadar IPv4 adreslerinin tükeneceği öngörülmüştür. IETF bu sorunu çözmek ve protokole yeni fonksiyonlar kazandırmak için 1990'ların başında çalışmalarına başlamış ve 1998 yılında RFC 2460 ile IPv6 standartlarını tanımlamıştır.

IPv4 adreslerinin dağıtıldı ilk zamanlarda etkili bir dağıtım yöntemi gözetilmemiştir. Var olan IPv4 adreslerinin yüzde altmışı Amerika Birleşik Devletleri tarafından kullanılırken yüzde kırklık kısım Dünya'nın geri kalanı tarafından paylaşılmaktadır. Yani IPv4 adreslerinin %60'ı, Dünya nüfusunun %5'ine tahsis edilmiştir [1]. Diğer yandan, Çin ve Hindistan gibi nüfusu çok fazla olan ülkelerde IP adresi ihtiyacı hızla artmaktadır. Ayrıca, mobil cihazlara, oyun konsollarına hatta arabalara bile IP adresi verilmesi bir diğer önemli etken olmuştur. 1993 yılında bu sorunu çözmek için IETF tarafından IPng (IP next generation) çalışmaları başlatılmıştır. Yapılan çalışmalar sonucunda, sorunun çözümü için iki yöntem belirlenmiştir:

1. Mevcut protokolü değiştirmeden, adres uzunluğunu arttırmak.

2. Tamamen yeni bir protokol geliştirmek.

Çok acil bir çözüm gerekmediği için, yeni nesil protokolün geliştirilmesine başlanmıştır. İlk adı IPng olan bu protokol, sonradan IPv6 adını almıştır.

Yeni protokole geçiş bir akşamda gerçekleşebilecek bir süreç değildir. Bu nedenle farklı iki protokolün uzun süre birlikte çalışması kaçınılmazdır. Protokoller arasındaki geçiş sürecinin başarısı ise, IPv4 düğümlerinin IPv6 düğümleri ile sorunsuz iletişimine bağlıdır. Bu amaçla NGtrans çalışma grubu tarafından, IPv6 geçiş yöntemleri hakkında araştırmalar yapılmış ve çeşitli yöntemler önerilmiştir [2].

IPv6 geçiş çalışmaları, ilk olarak 6bone [3] test ağında başlatılmıştır. Özellikle adres sıkıntısı yaşayan Asya ülkeleri ve sonrasında Avrupa ve Amerika'da oluşturulan IPv6 görev gücü (task force) grupları, IPv6 geçiş çalışmalarına hız kazandırmıştır. Türkiye'de ise IPv6 çalışmaları, ULAKBIM tarafından kurulan "IPv6 Görev Gücü" tarafından başlatılmış ve "Ulusal IPv6 Protokol Altyapısı Tasarımı ve Geçişi Projesi" adı altında ulusal bir projeye dönüştürülmüştür [4].

IPv6 geçiş yöntemlerinin güvenlik ve performans analizinin yapıldığı bu çalışmanın temel amacı, yeni protokolün uygulanması sırasında izlenecek yöntem veya yöntemler hakkında bir öngörüde bulunmaktır. Güvenlik, performans ve uygulanabilirlik ölçütleri ışığında, yapılan çok sayıda araştırma ve yazarın tecrübelerine dayanarak, küçük ağlardan çok büyük ağlara kadar IPv6 geçiş

yöntemleri değerlendirilmiştir. Ayrıca, daha önce detaylı araştırması yapılmamış, IPv6 geçiş yöntemlerinin servis dışı bırakma saldırıları karşısındaki durumu, gerçek ağ ortamında test edilmiştir. Bu tez çalışmasına ait bir diğer önemli konu, bazı çevirici yöntemlerinde, tek nokta hatalarının giderilmesi için önerilen mekanizmadır.

1.1. IPv6 Geçiş Yöntemleri Üzerine Yapılan Çalışmalar

IPv6 kullanan ağların ve sistemlerin 2011 yılı sonuna kadar yaygınlaşacağı beklentisi ile birlikte, bu yeni protokolün performans değerleri üzerine çalışmalar da başlamıştır. Ioan Raicu, yaptığı tez çalışmasında, IPv6 performans ölçülmelerini IPv4 ile karşılaştırmış ayrıca IPv6'ya geçiş yöntemlerinden düğümden-düğüme paketleme (encapsulation) ve yönlendiriciden-yönlendiriciye (router-to-router) tünelleme için testler yapmıştır. Yazar, düzenlediği test ortamında performans ölçütleri olarak işlemci kullanımı, soket oluşturma süresi, saniyedeki TCP bağlantı sayısı ve C++ ile geliştirilen istemci / sunucu yapısındaki video uygulamasının throughput ve saniyedeki çerçeve (frame) sayısı değerlerini almıştır. Geçiş yöntemine ait yaptığı testlerde sadece Windows 2000 işletim sistemi kullanılmış diğer testlerde ise Windows 2000 ve Solaris 8 birlikte kullanılmıştır. Çalışma sonuçlarına göre yönlendiriciden-yönlendiriciye tünel througput değeri yalın IPv6 kullanılan duruma göre %1-%7 arasında daha düşük performans göstermiş buna karşın düğümden-düğüme paketleme değerinde, IPv4 performansına kıyasla %110 gibi bir düşüş gözlemlemiştir. İşlemci kullanımı bakımından da düğümden-düğüme yapılandırması en çok kaynak tüketen yapı olarak tespit edilmiştir. Bilgisayar ağlarındaki bir diğer performans ölçütü olan gecikme süresi (latency) için yalın IPv6 ve yönlendiriciden-yönlendiriciye tünel değerleri ortalama 40ms iken IPv4 kullanıldığında gecikme süresinin 15ms değerinde olduğu ölçülmüştür. Soket oluşturma süresi ve saniyedeki TCP bağlantı sayısı bakımından düğümden-düğüme performansı yönlendiriciden-yönlendiriciye tünel performansından daha iyi olduğu tespit edilmiştir. Son olarak, geliştirilen video uygulamasında, IPv4 kullanıldığında 66Mbit/s ve 8.9 çerçeve/s değerinin yalın IPv6 değeri olan 26Mbit/s ve 3.5 çerçeve/s değerinden oldukça fazla olduğu, benzer performansın yönlendiriciden-yönlendiriciye tünelde de görüldüğü belirtilmiştir [5].

IETF NGtrans çalışma grubu tarafından ortaya konan birçok geçiş yöntemi ile birlikte bu yöntemlerin gerçek ortamdaki performans ve güvenlik sorunları üzerine çalışmalar da hız kazanmıştır. Yao-Chung Chang ve arkdaşları tarafından yapılan çalışmada 6to4, tunnel broker ve configured tunnel yöntemleri için performans değerleri araştırılmıştır. Gerçek bir ağ üzerinden 6bone ve KAME gibi bilinen IPv6 ağlarına ICMPv6 paketleri gönderilerek yapılan testlerde throughput, işlemci kullanımı, gecikme süresi ve paket kaybı değerleri ölçüt olarak belirlenmiştir. Araştırmacıların elde ettiği sonuçlara göre, 6to4 yöntemi en az gecikme süresi ve en fazla througput değeri ile en iyi performansı gösterirken, tunnel broker ise her iki ölçüte göre en az performanslı geçiş yöntemi olarak tespit edilmiştir. İşlemci kullanımının ise 6to4 yönteminde daha fazla olduğu gözlemlenmiştir. Araştırmacılar ilk tercih olarak 6to4 yönteminin kullanılabileceğini öngörmüşler fakat QoS ve

multicast gibi teknolojiler için configuerd tunnel yönteminin daha makul olacağını belirtmişlerdir [6].

Ra'ed AlJa'afreh ve arkadaşlarının, IPv6 geçiş yöntemlerinden BDMS (Bi-Directional Mapping System) ve DSTM (Dual Stack Transition Mechanism) üzerine yaptıkları çalışma, bu iki yöntemin performansını ortaya koymaya yöneliktir. OMNeT++ simülatörü kullanılarak 3 farklı senaryoda test yapılmıştır. İlk senaryoda v4-v4, v6-v6 doğrudan bağlantı, ikinci senaryoda BDMS çevirici kullanılarak v4-v6 ve v6-v4 bağlantısı ve son senaryoda DSTM çevirici kullanılarak v6-v4 bağlantısı test edilmiştir. Test ölçütü olarak yazarlar, througput ve uçtan uca gecikme süresi değerlerini almışlardır. İlk senaryo sonuçlarına göre v4-v4 gecikme süresi v6-v6 gecikme süresinden kısa ve througput değerleri birbirine yakın bulunmuştur. Bağlantı sayısı arttıkça v4-v4 performansı her iki ölçüte göre de daha iyi sonuçlar göstermiştir. Yazarlar bunun nedenini IPv6 başlığının (header) IPv4 başlığına göre daha uzun olmasına bağlamışlardır. İkinci senaryo sonuçlarına göre ise v4-v6 gecikme süresi v6-v4 gecikme süresinden daha düşük bulunmuştur. Yazarlar bu durumu IPv6 ile IPv4 başlık uzunluğundaki farka ve giden paketlerde sağlama (checksum) işleminden kaynaklanan kayıplara bağlamışlardır. BDMS çevirici ile yapılan testlerde througput değerleri ise benzer bulunmuştur. DSTM çevirici kullanılan son senaryoda ise gecikme süresi ve througput değerlerinin BSDM çeviriciye oranla daha düşük performanslı olduğu tespit edilmiştir. Yazarlar, bu durumun tünelleme işleminden kaynaklanan ağ tıkanıklığı olabileceğini belirtmişlerdir. [7]. Yine benzer bir çalışma Eun-Young Park ve arkadaşları tarafından yapılmış ve mevcut IPv4 ağlarından IPv6 ağları ile iletişim için DSTM çevirici kullanımı araştırılmıştır [8].

IPv6 geçiş yöntemlerinden olan NAT-PT çevirici ile ilgili bir çalışma da Yong Guen Hong ve arkadaşları tarafından yapılmıştır [9]. IPv6 geçiş yöntemi olarak tasarlanan NAT-PT çeviriciler sadece IP / ICMP katmanını için çevirim yapabilmekte, uygulamaların bulunduğu transport katmanında TCP /UDP çevirimi için ALG (Application Layer Gateway) kullanılması gerekmektedir. Her uygulama farklı prosedür, farklı veri iletim metodu ve farklı veri yapısı barındırdığı için genel bir ALG yoktur. Dolayısı ile her bir uygulama için özel olarak tasarlanmış ALG yapılandırmasına gerek vardır.Yazarlar, Linux üzerinde Netfilter yapısını kullanan FTP-ALG çevirici ile testler yapmışlardır. Test senaryosunda, öncelikle ip katmanı sonrasında uygulama katmanı çevrimi yapılmış ve elde edilen sonuçlar bu senaryonun tersi ile karşılaştırılmıştır. Her iki durum için de througput değerlerinin benzer çıktığı tespit edilmiştir.

Yeterli IPv4 adresi olmayan ağlarda, genellikle NAT kullanılarak internete veya diğer ağlara erişim sağlanır [10]. Shiang-Ming Huang ve arkadaşları, NAT arkasında bulunan ağların IPv6 ağları ile iletişimi için Teredo yöntemini araştırmışlar ve ICMPv6 kullanarak performans ölçümü yapmışlardır [11]. Yazarlar, yaptıkları testlerde Linux çekirdek seviyesinde çalışan NICI-Teredo ve BSD için geliştirilen 6WIND-Teredo uygulamalarının kullanıcı seviyesinde çalışan Miredo-Teredo'dan daha performanslı olduğunu tespit etmişlerdir.

Mevcut geçiş yöntemlerine ilişkin literatür taramalarının yanı sıra [12] farklı metotlar üzerine de çalışmalar yapılmıştır. Tushar M. Raste ve D.B. Kulkarni yaptıkları çalışmada Netfilter hooks [13] kullanarak Linux çekirdek seviyesinde 4to6 tünelleme mekanizması önermişlerdir. Bu yöntemde bilgisayarlar IPv6 ağında bulunmaktadır. IPv4 ağı ile iletişime geçmek isteyen bilgisayar, yaptığı dns sorguları veya açtığı v4 soketleri yardımıyla algılanır ve oluşturulan sanal arayüz üzerinden tünellenerek yönlendiriciye bağlanır. Önerilen bu yöntemde, arada ALG tarzı cihazlar olmadığı ve sadece IPv6 ağı yönetimi gerektirdiği için yazarlar tarafından ekonomik olarak değerlendirilmiştir [14].

Hiç şüphesiz IPv6 ile gelen önemli bir özellik de MIPv6 (Mobile IPv6) olmuştur. Bu sayede, farklı IPv6 ağları arasında gezinen bir istemci, bağlantı ayarlarını değiştirmeksizin iletişimini sürdürebilir [15]. Su-Jin Lee ve arkadaşları gezici düğümlerin IPv4 ağları ile iletişimi için NAT-PT yönteminin geliştirilmiş bir sürümümü tasarlamışlar ve NAT-PTm olarak adlandırmışlardır. MIPv6 özelliğinden faydalanarak farklı IPv6 ağları arasında geçişin test edildiği ilk senaryoda IPv4 ağı ile iletişim için NAT-PT çevirici kullanılmıştır. Bu durumda tünelleme ve geri tünellemeden dolayı oluşan verimsiz yönlendirmelerin (routing) performans kayıplarına neden olduğu belirtilmiştir. Yazarlar tarafından önerilen NAT-PTm yönteminde ise eşleşme tablosu (mapping table) genişletilerek ev ajanının (home agent) görevi devralınmıştır. Bunun sonucunda %39'a varan performans kazancı elde edilmiştir [16].

Bilgisayar veya diğer paket anahtarlamalı ağlar üzerinde bazı servislerin hizmet kalitesini arttırmak için kaynak ayrılması QoS mekanizması sayesinde gerçekleştirilir. Bu sayede sesli görüşme gibi anlık veya kesintisiz iletişim gerektiren uygulamalar için p2p gibi gecikmeye toleranslı uygulamalara göre öncelik atanır. Christos Bouras ve arkadaşları IPv6 ağlarında QoS konusunu ele almışlar ve ikili yığın yapıda çalışan yönlendiriciler üzerinde testler yapmışlardır. Hazırlanan test ortamında sadece IPv4, sadece IPv6 ve ikili yığın için DiffServ mimarisinde [17] QoS denemeleri gerçekleştirilmiştir. Yazarlar, performans ölçütü olarak işlemci yükü, paket kaybı ve gecikme süresi değerlerini kullanmışlardır. Test sonuçlarına göre IPv6 için işlemci yükünde IPv4 işlemci yüküne oranla %5-9 arasında artış gözlemlenmiştir. Paket kayıpları ve gecikme süreleri ise her üç ortamda da benzer çıkmıştır [18].

A. Dutta ve arkadaşları, IPv6 geçiş sürecinde yazılımların durumunu irdelemişler ve Amerikan ordusunda yaygın olarak kullanılan MCS-L uygulaması üzerinden IPv4 ağları ile iletişimin nasıl sağlanacağını tartışmışlardır. Söz konusu yazılımın v4/v6 ağlarında birlikte çalışması için 4 durum belirlenmiştir. İlk olarak, yazılım bir v4 uygulamasıdır, v6 desteklemez ama istemci ikili yığın yapıda çalışır. İkinci öneride, yazılım v4 ve v6 çalışan iki eş yazılıma dönüştürülür. Üçüncü öneride ise yazılım hem v4 hem de v6 çalışması için düzenlenir. Son olarak, yazılım hem v4 hem v6 çalışması için düzenlenir, çekirdekte v6 desteği olmasa bile yazılım çalışmaya devam eder. Yazarlar, her bir durum için olası geçiş mekanizmalarını tartışmış ve uygun yöntemlerin ihtiyaca göre titizlikle seçilmesi gerektiğini belirtmişlerdir [19].

IPv6 geçiş yöntemlerinde güvenlik, hiç şüphesiz detaylı araştırılması gereken bir konudur. 6NET konsorsiyumu [20] tarafından yayınlanan "An IPv6 Deployment Guide" ve CISCO tarafından yayınlanan "IPv6 Security" kitaplarında çeşitli geçiş yöntemlerine ilişkin güvenlik sorunları irdelenmiştir. Qinhua Zheng ve arkadaşları tarafından yapılan bir çalışmada [21], öngörülenin aksine IPv6 ağlarında solucan (worm) dağılımının daha hızlı olabileceği tespit edilmiştir. İkili yığın yapıdaki ağda, W32 Blaster solucanı ile yapılan bu araştırma, IPv6 ağlarının daha güvenli olacağına ilişkin beklentilerin gerçekçi olamadığını göstermektedir.

Bilgisayar ağlarında servis dışı bırakma (DoS) saldırıları sıklıkla görülmektedir. Beyhan Çalışkan ve Onur Bektaş tarafından yapılan çalışmada ikili yığın yapıdaki Apache web sunucusunun performansı araştırılmıştır. Araştırmacılar, test sonuçlarına göre ikili yığın yapıda sunucu performansının düşüş gösterdiğini, saldırı altında ise sunucunun servis veremez hale geldiğini tespit etmişlerdir [22].

2. IPv6 ÖZELLİKLERİ

IPv4'ün halefi olarak geliştirilen yeni Internet Protokol'ü IPv6, 1998 yılında RFC 2460 ile standartlaşmıştır. Bu yeni protokolü IPv4'ten ayıran başlıca değişiklikler aşağıda sıralanmıştır:

- Genişletilmiş Adresleme Özellikleri

  IPv6 ile birlikte 32 bit olan adres uzayı 128 bite genişletilmiş, bu sayede daha çok düğüme adres sağlanmış ve daha fazla adresleme hiyerarşisi desteklenmiştir. Kolay oto konfigürasyon bir diğer yenlik olmuştur. Çoklu dağıtım (multicast) adreslerine "scope" alanı eklenerek yönlendirme ölçeklenirliği geliştirilmiştir. Bir gruptaki düğümlere paket göndermek için "anycast" adresi tanımlanmıştır.

- Basitleştirilmiş Başlık Biçimi

  Bazı IPv4 başlık alanları atılmış veya opsiyonel hale getirilmiştir. Bu sayede paket işleme maliyeti ve başlık için harcanan bant genişliği azaltılmıştır.

- Ek Başlık ve Seçenek Alanları İçin Geliştirilmiş Destek

  IP başlık seçeneklerinde yapılan değişiklikler daha verimli iletim sağlarken, seçenek alanın uzunluğu daha az kısıtlayıcıdır. Ayrıca IPv6 ile birlikte yeni seçenek alanlarının tanımlanabilmesi için daha çok esneklik sağlanmıştır.

- Akış (Flow) Etiketi Özelliği

  Gönderici tarafından belirli bir trafik için özel bir taşıma istendiğinde paketin etiketlenmesi özelliği eklenmiştir.

- Kimlik Doğrulama ve Gizlilik Özelliği

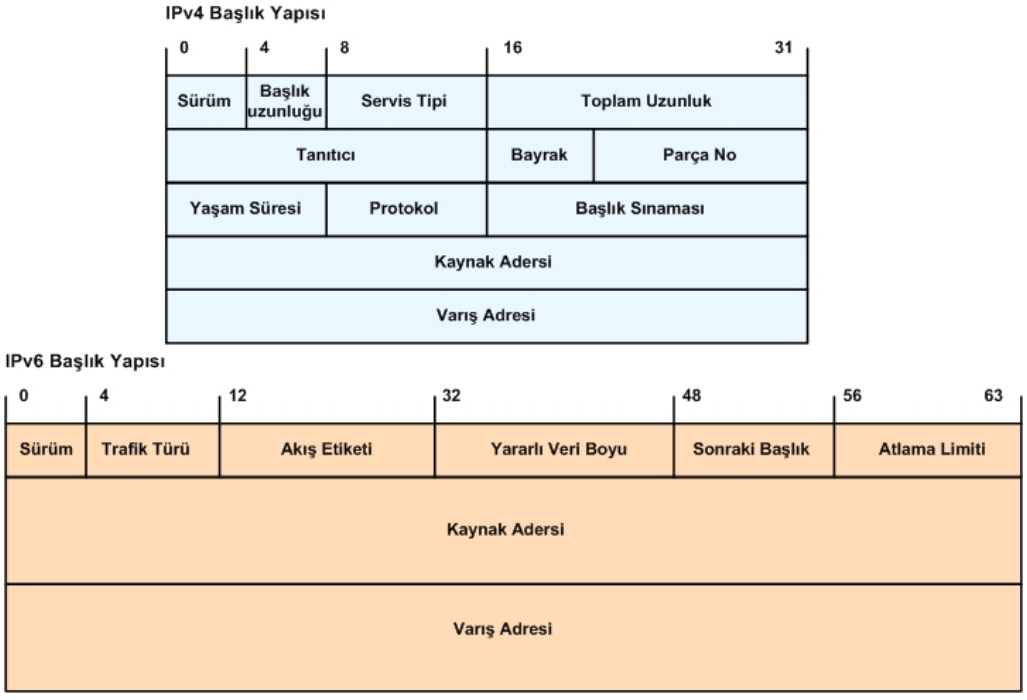
  Ek başlıklar sayesinde kimlik doğrulanma, veri bütünlüğü ve opsiyonel olarak veri gizliliği IPv6 için tanımlanmıştır.

2.1. IPv6 Protokol Yapısı

Çalışmanın bu bölümünde IPv6 başlık yapısı açıklanmış ve IPv4 başlık yapısı ile karşılaştırma yapılmıştır. Protokolün yapısını anlamak IPv6 geçiş yöntemlerinde kullanılan yaklaşımların anlaşılabilmesi için önemlidir. Bununla birlikte IPv4 protokol yapısına ait ayrıntılı detaylar çalışmanın kapsamı dışındadır.

2.1.1. IPv6 başlık yapısına detaylı bakış

IPv6 başlığı, her biri 16 byte olan kaynak adresi ve varış adresi alanları ve 8 byte'lık genel başlık bilgisi alanı ile toplam 40 byte uzunluğundadır. IPv4 başlığında bulunan, başlık uzunluğu (header lenght), tanılama (identification), bayraklar (flags), parça no (fragment offset) , başlık sınaması (header checksum) alanları IPv6 başlık yapısından çıkartılmıştır. IPv6 ve IPv4 başlık yapıları şekil 1'de gösterilmiştir.



Şekil 1. IPv6 – IPv4 başlık yapısı

IPv6 başlığında yer alan alanlar aşağıda ayrıntılı olarak açıklanmıştır:

- Sürüm (version)

  Kullanılan İnternet Protokolü'nün belirtildiği 4 bit uzunluğundaki alandır. IPv6 kullanıldığında bu alana "6" sayısı atanır.

- Trafik Türü (traffic class)

  IPv4'teki "servis tipi" alanının yerini alır ve 8 bit uzunluğundadır. Gönderilen IPv6 paketlerinin önceliği ve türü bu alanda taşınır. Trafik türü alanın kullanımı RFC 2474 ile açıklanmıştır.

- Akış Etiketi (flow label)

  Paketlerin aktarımı sırasında gönderici ve alıcı arasında aynı işleyişin gerçekleşmesi için tanımlanmış alandır. Gönderici paketleri bir takım opsiyonlarla etiketledikten sonra yönlendiricide ilgili akışa ait bu iz takip edilir ve daha etkili paket işlemesi gerçekleşir. Bu sayede, her bir paket başlığının tekrar işlenmesi gerekmemektedir. Bu alan 20 bit uzunluğundadır.

  Bu çalışma yazıldığı tarihte akış etiketi hala deneysel bir konumda bulunmakta ve bu özelliği desteklemeyen cihazlarda bu olan sıfır olarak atanmaktadır.

- Yararlı Veri Boyu (payload lenght)

  16 bit uzunluğundaki bu alan IP başlığından sonra gelen verinin büyüklüğünü gösterir. Bu alan hesaplanırken sadece başlık kısmından sonraki veri göz önünde bulundurulur. IPv4'te ise bu alan başlık + veri uzunluğu olarak hesaplanmaktadır. Yararlı veri boyu alanının 16 bit olması nedeniyle maksimum yararlı veri büyüklüğü 64KB ile sınırlanmaktadır.

- Sonraki Başlık (next header)

  8 bit uzunluğundadır. IPv6 başlığından sonra gelen ek-başlığın türünün taşındığı alandır. IPv4 protokolündeki alanlarla aynı değerleri alır.

- Atlama Limiti (hop limit)

  Bu alan IPv4 başlığındaki TTL alanı gibidir. Gönderilen pakete ait atlama limiti değeri her bir düğümde 1 azaltılır. Bu değerin 0 olması durumunda paket yok edilir ve ICMPv6 zaman aşımı mesajı gönderilir. Bu alanın uzunluğu 8 bit olarak belirlenmiştir.

- Kaynak Adresi (source address)

Gönderilen paketlerin kaynak adresi bilgisi bu alanda taşınır. 128 bit uzunluğundadır.

- Varış Adresi (destination address)

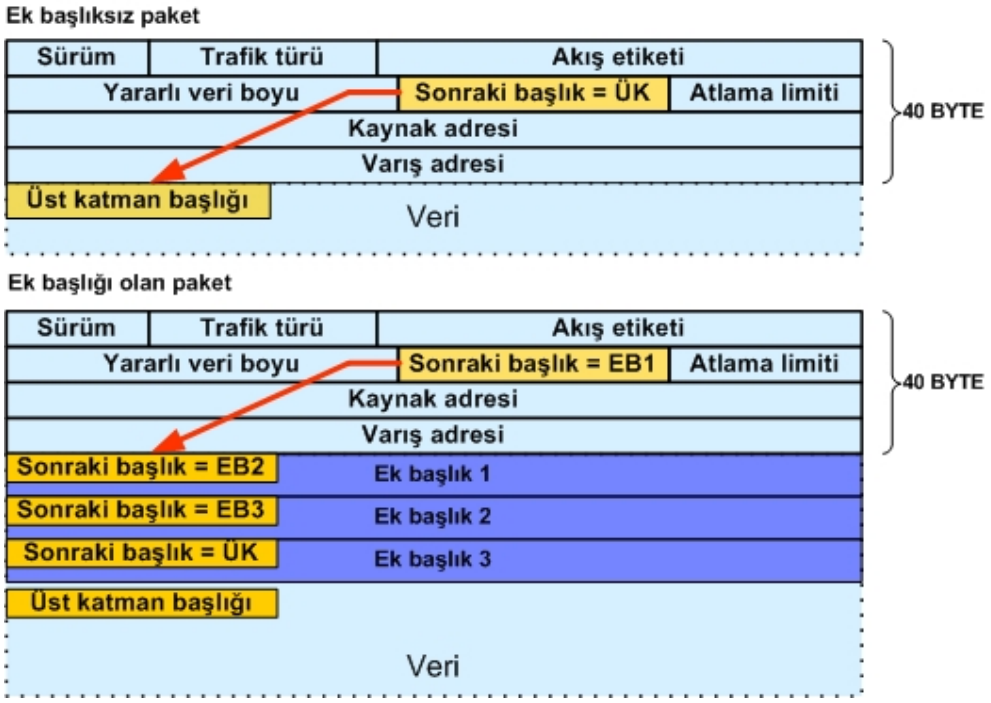
  Gönderilen paketlerin varış adresinin taşındığı alandır. 128 bit uzunluğundadır.

2.1.2. IPv6 ek başlıkları

IPv4 başlık uzunluğu, minimum 20 byte maksimum 60 byte arasında çeşitli opsiyonları belirtmek için değişiklik gösterebilir. Bu durum paket işlemede performans kayıplarına yol açmaktadır. IPv6 başlık yapısı tasarlanırken, hızlı paket işlemi için olabildiğince basit olması amaçlanmış, çeşitli opsiyon gereksinimlerine ise ek başlıklar (extension headers) tanımlanarak çözüm sağlanmıştır. Aşağıda belirtilen ilk 4 ek başlık RFC 2460 ile, kimlik doğrulama ek başlığı RFC 2402 ile ve güvenlik verisi paketleme ek başlığı RFC 2406 ile tanımlanmıştır.

- Düğümden-düğüme atlama başlığı (Hop-by-Hop Options header)
- Yönlendirme başlığı (Routing header)
- Parçalama başlığı (Fragment header)
- Alıcı Hedef başlığı (Destination Options header)
- Kimlik Doğrulama başlığı (Authentication header)
- Güvenlik Verisi Paketleme başlığı (Encapsulating Secuirty Payload header)

IPv6 paketlerinde, sıfır, bir veya birden fazla ek başlık bulunabilir. Bu ek başlıklar, IPv6 başlığı ile üst-katman (upper-layer) protokol başlığı arasında yer alır ve her biri "sonraki başlık" alanında belirtilir. IPv6 paketi ek başlıklı ve ek başlıksız olarak Şekil 2'de gösterilmiştir.



Şekil 2. IPv6 ek başlığının kullanımı

IPv6 paketlerinin uçtan-uca iletimi sırasında, arada bulunan düğümler, ek başlıklara bakmaz ve işlemez. Ek başlıklar sadece varış adresinde sırasıyla işlenmektedir. Varış adresi çoklu dağıtım adresi ise, çoklu dağıtım grubunda bulunan düğümlerin tamamı ek başlıkları işlemektedir. Bu durumun tek istisnası düğümden-düğüme atlama başlığıdır. Düğümden-düğüme atlama başlığı tanımlı ise kaynak adresi dahil tüm düğümler tarafından ek başlığa bakılmaktadır.

IPv6 ek başlıkları RFC 2460'a göre mutlaka sırasıyla işlenmelidir. Birden fazla ek başlık bulunması durumunda ek başlıkların sırası aşağıda belirtildiği gibi düzenlenmektedir:

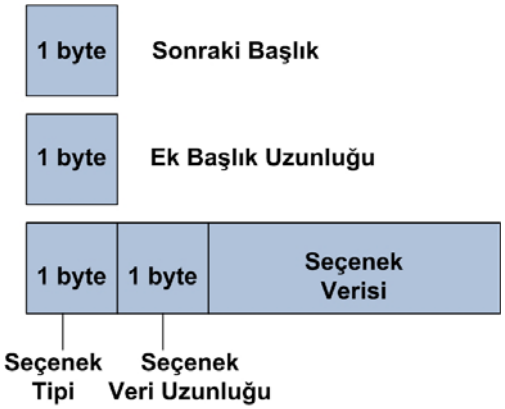
- IPv6 başlığı
- Düğümden-düğüme atlama başlığı
- Alıcı Hedef başlığı
- Yönlendirme başlığı
- Parçalama başlığı
- Kimlik Doğrulama başlığı
- Güvenlik Verisi Paketleme başlığı

- Alıcı Hedef başlığı
- Üst-Katman başlığı

Alıcı hedef başlığı en fazla 2 defa kullanılabilirken diğer tüm ek başlıklar birer defa kullanılır.

2.1.2.1. Düğümden-düğüme atlama başlığı

Düğümden-düğüme atlama başlığı IPv6 başlığından hemen sonra gelir ve "sonraki başlık" alanında 0 değeri ile gösterilir. Bu ek başlık, iletişimin yapıldığı bütün düğümler tarafından işlenmek zorundadır.



Şekil 3. IPv6 düğümden düğüme atlama başlığı

Düğümden-düğüme ek başlığı içerisinde sırasıyla aşağıda belirtilen alanlar yer alır:

- Sonraki Başlık (Next Header)

  Bu alan düğümden-düğüme atlama başlığından sonra gelecek başlık tipini tanımlar. 8 bit uzunluğundadır.

- Ek Başlık Uzunluğu (Header Extension Lenght)

  Bu alan düğümden-düğüme atlama başlığının uzunluğunu 8 byte'lık gruplar halinde gösterir. İlk 8 byte hesaplamaya dahil edilmez. Yani başlık 8 byte'tan küçük ise bu alan 0 değerini alır. 8 bit uzunluğundadır.

- Seçenek Alanı (Options)

  Bir veya birden çok seçeneğin yer aldığı bu alanın uzunluğu "ek başlık uzunluğu" alanında belirtilir. Seçenek tipi, seçenek alanının ilk 8 bitlik

kısmında yer almaktadır. Kullanılan seçeneğin, işlem yapan düğüm tarafından anlaşılamaması durumunda, yapılacak işlemi ilk 2 bit sayesinde belirtilir:

- 00: atla ve işleme devam et.
- 01: paketi yok et.
- 10: paketi yok et ve ICMP problem 2 kodu gönder.
- 11: varış adresi çoklu dağıtım adresi değilse paketi yok et ve ICMP problem 2 kodu gönder.

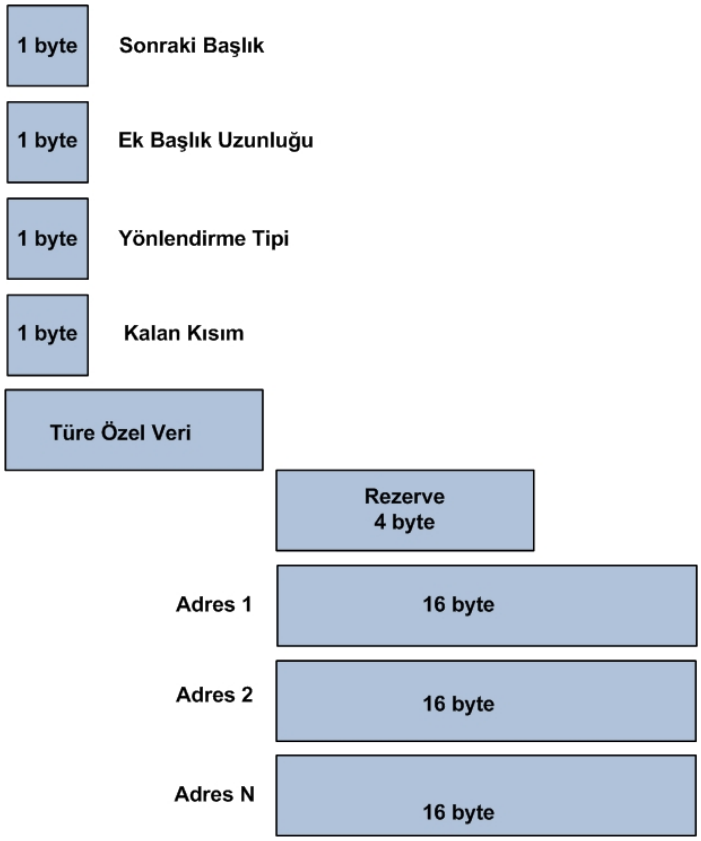
Seçenek tipi alanındaki 3. bit değeri 1 ise, seçenek bilgilerinin yönlendirme sırasında değiştirilebileceğini, 0 ise değiştirilemeyeceğini belirtir.

Düğümden-düğüme atlama ek başlığında yer alan "seçenek tipi" alanı, IPv6 Jumbo-veri (Jumbogram) göndermeye olanak sağlamaktadır [23]. Bu sayede yararlı veri boyutu maksimum 65,536 byte olan değerinden 4,294,967,295 byte değerine kadar çıkabilmektedir. Jumbo-veri göndermek için, IPv6 başlığında yer alan "yararlı veri boyutu uzunluğu" alanına 0 değeri atanır. Aynı şekilde "sonraki başlık" alanına da 0 değeri atanarak düğümden-düğüme atlama ek başlığı olduğu belirtilir. Düğümden-düğüme atlama, "seçenek tipi" alanına atanan 149 değeri ise IPv6 Jumbo-veriyi tanımlamaktadır.

Seçenek tipi alanı ayrıca RFC 2711 ile tanımlanan RSVP (Resource Reservation Protocol) ve MLD (Multicast Listener Discovery) gibi protokollerin yönlendiriciler tarafından işlenmesini sağlamak için de kullanılır.

2.1.2.2. Yönlendirme başlığı

Yönlendirme başlığı, gönderilen paketin varış adresine ulaşana kadar, üzerinden geçeceği düğümlerin listesini tanımlar. Bu ek başlık "sonraki başlık" alanına 43 değeri atanarak belirtilir.



Şekil 4. IPv6 yönlendirme başlığı

Yönlendirme ek başlığı içerisinde sırasıyla aşağıda belirtilen alanlar yer alır:

- Sonraki Başlık

  Bu alan yönlendirme başlığından sonra gelecek başlık tipini tanımlar. 8 bit uzunluğundadır.

- Ek Başlık Uzunluğu

  Bu alan yönlendirme başlığının uzunluğunu 8 byte'lık gruplar halinde gösterir. İlk 8 byte hesaplamaya dahil edilmez. 8 bit uzunluğundadır.

- Yönlendirme Tipi (Routing Type)

  Yönlendirme başlığının tipini belirten alandır. 8 bit uzunluğundadır. Varsayılan yönlendirme tipi değeri 0'dır.

- Kalan Kısım (Segments Left)

Paketin varış adresine ulaşıncaya kadar geçmesi gereken düğüm sayısını gösterir. 8 bit uzunluğundadır.

- Türe Özel Veri (Type-Specific Data)

  Bu alanın uzunluğu her zaman 8 byte'ın katları şeklindedir ve yönlendirme tipine bağlı olarak değişir.

IPv6 paketleri, düğümlerde işlenirken, yönlendirme başlığındaki yönlendirme tipi değerinin tanımlanamaması durumunda "kalan kısım" değerine bağlı olarak aşağıdaki işlemlerden biri gerçekleştirilir:

- Kalan kısım değeri 0 ise, düğüm yönlendirme başlığını önemsemez ve sonraki başlık alanında belirtilen değeri işler.

- Kalan kısım değeri 0'dan farklı ise, düğüm paketi yok eder ve ICMP parametre problemi, kod 0 mesajını kaynağa gönderir.

Eğer iletme düğümü (forwarding node) sonraki bağlantıya ait MTU (Maximum Transmission Unit) değerinin paketin boyutundan küçük olduğunu tespit ederse, paketi yok eder ve kaynak adrese ICMP "paket çok büyük" mesajı gönderir.

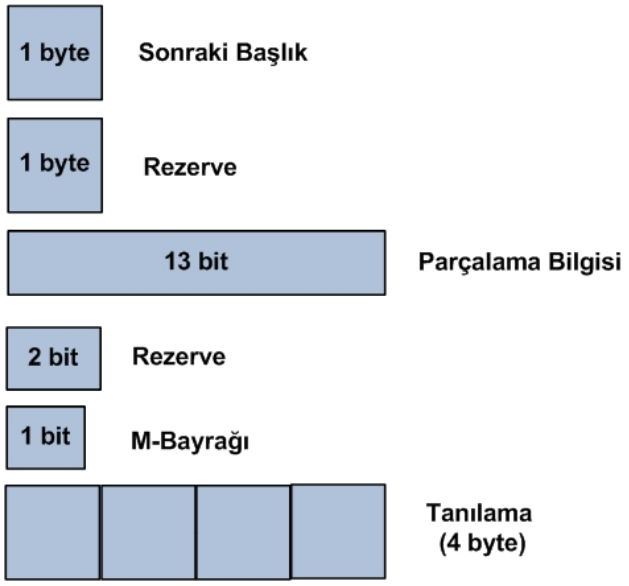
RFC 2460 ile sadece "tip 0" için yönlendirme başlığı tanımı yapılmıştır. Yönlendirme başlığını ilk işleyen düğüm, IPv6 başlığında varış adresi olarak tanımlanan düğümdür. Bu düğüm "kalan kısım" değerini 1 eksiltir ve yönlendirme başlığındaki sonraki adres değerini IPv6 başlığındaki varış adresi alanı olarak değiştirir. Bu işlem son düğüme varılıncaya kadar tekrar edilir. Yönlendirme başlığının yukarıda anlatılan algoritması Şekil 5'te örnek olarak gösterilmiştir.

	IPv6 başlığı	Yönlendirme başlığı
Kaynaktan 1. düğüme	Kaynak adresi K Varış adresi D1	Kalan Kısım = 3 Adres [1] = D2 Adres [2] = D3 Adres [3] = V
1.düğümden 2.düğüme	Kaynak adresi K Varış adresi D2	Kalan Kısım = 2 Adres [1] = D1 Adres [2] = D3 Adres [3] = V
2.düğümden 3.düğüme	Kaynak adresi K Varış adresi D3	Kalan Kısım = 1 Adres [1] = D1 Adres [2] = D2 Adres [3] = V
3.düğümden Varış adresine	Kaynak adresi K Varış adresi V	Kalan Kısım = 0 Adres [1] = D1 Adres [2] = D2 Adres [3] = D3

Şekil 5. IPv6 yönlendirme başlığının kullanımı

2.1.2.3. Parçalama başlığı

Bir IPv6 düğümü göndereceği paketin maksimum büyüklüğünü "path MTU discovery" [24] kullanarak tayin eder. Gönderilmek istenen paketin büyüklüğü desteklenen MTU değerinden büyük ise, gönderen düğüm paketi parçalara ayırır. Paketin parçalanması kaynak düğümünde, birleştirilmesi varış düğümünde gerçekleştirilir. Paketin yol boyunca üzerinden geçtiği yönlendiriciler tarafından parçalanma veya birleştirilme işlemi yapılmaz.



Şekil 6. IPv6 parçalama başlığı

Parçalama ek başlığı içerisinde sırasıyla aşağıda belirtilen alanlar yer alır:

- Sonraki Başlık

  Bu alan parçalama başlığından sonra gelecek başlık tipini tanımlar. 8 bit uzunluğundadır.

- Rezerve

  Bu alan rezerve edilmiştir ve kullanılmamaktadır. Varsayımlı 0 değerini alır ve 8 bit uzunluğundadır.

- Parçalama Bilgisi (Fragment Offset)

  Parçalanmış paketin, 8 byte'lık gruplar halinde orijinal paketten kayma bilgisini taşır. Bu alan toplam 13 bit uzunluğundadır.

- Rezerve

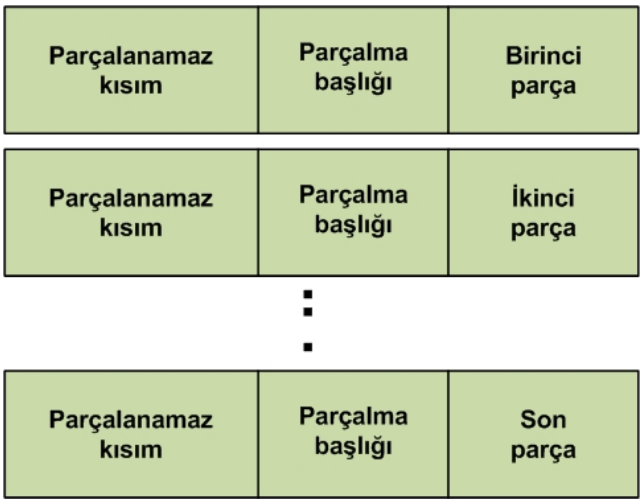
  Bu alan rezerve edilmiştir ve kullanılmamaktadır. Varsayımlı 0 değerini alır ve 2 bit uzunluğundadır.

- M-Bayrağı (M-Flag)
  1 bit uzunluğundadır. 1 değerini aldığında daha parçalanmış paket olduğunu, 0 değerini aldığında son parçayı tanımlar.

- Tanılama (Identification)

32 bit uzunluğundadır. Kaynak düğüm tarafından oluşturulur. Parçalanmış paketlerin ait olduğu orijinal paketi belirtmek için kullanılır.

IPv6 paketlerinin başlıkları ve her düğüm tarafından işlenmesi gereken ek başlıları parçalanamaz. Parçalanan kısım, son düğüm tarafından işlenen ek başlılar, üst-katman başlıkları ve anlamalı veri alanlarıdır. Şekil 7'de parçalanma işlemi gösterilmiştir.

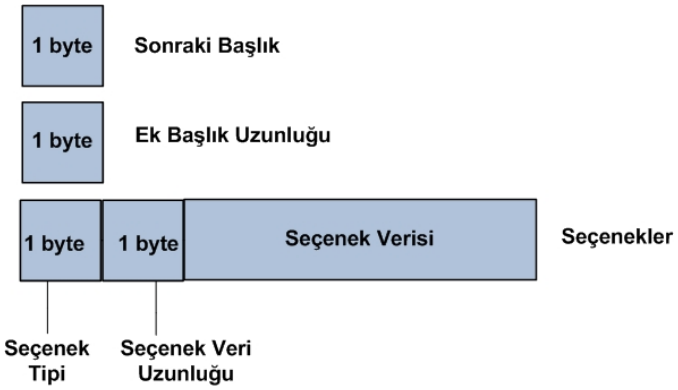


Şekil 7. IPv6 parçalama işlemi

Parçalanan paketler varış adresinde tekrar birleştirilir. Bu işlemin sorunsuz gerçekleşmesi için her bir parçadaki kaynak ve varış adresleri eşit olmalıdır. Ayrıca her parça aynı tanılama değerine sahip olmalıdır. İlk parçalanmış paketin varış adresine ulaşmasından sonra, 60 saniye içersinde birleştirme işlemini tamamlayacak kadar parça gelmediyse tüm paketler yok edilir. Varış düğümü, aldığı ilk parçada kaydırma (offset) değerini 0 olarak görürse, kaynak adresine ICMPv6 parça birleştirme süresi aşıldı (fragment reassembly time exceeded) mesajını gönderir.

2.1.2.4. Alıcı hedef başlığı

Alıcı hedef başlığında, sadece varış adresinin işlemesi için opsiyonel bilgiler taşınır. Bu ek başlık, IPv6 başlığındaki sonraki başlık alanına 60 değeri atanarak tanımlanır. IPv6 paketinde yönlendirme başlığından önce ve üst-katman başlığından önce iki defa yer alabilir. Yönlendirme başlığından önce yer alıyorsa, yönlendirme başlığında belirtilen yönlendiriciler tarafından işlenmesi için bilgi taşır. Üst-katman başlığından önce yer alıyorsa, varış düğümü için bilgi taşır.



Şekil 8. IPv6 alıcı hedef başlığı

Alıcı hedef ek başlığı içerisinde sırasıyla aşağıda belirtilen alanlar yer alır:

- Sonraki Başlık

  Bu alan alıcı hedef başlığından sonra gelecek başlık tipini tanımlar. 8 bit uzunluğundadır.

- Ek Başlık Uzunluğu

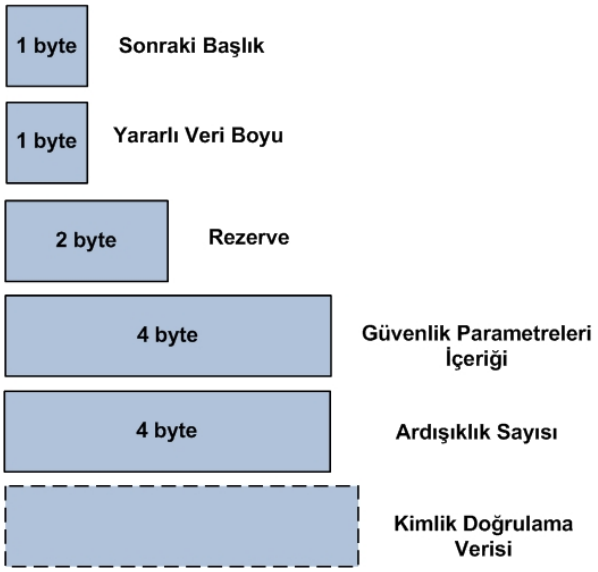
  Bu alan alıcı hedef başlığının uzunluğunu 8 byte'lık gruplar halinde gösterir. İlk 8 byte hesaplamaya dahil edilmez. Yani başlık 8 byte'tan küçük ise bu alan 0 değerini alır. 8 bit uzunluğundadır.

- Seçenek Alanı

Bir veya birden çok seçeneğin yer aldığı bu alanın uzunluğu "ek başlık uzunluğu" alanında belirtilir. Seçenek tipi, seçenek alanının ilk 8 bitlik kısmında yer almaktadır. Bu alanın kullanımı düğümden-düğüme ek başlığının kullanımı ile aynı yapıdadır.

2.1.2.5. Kimlik doğrulama başlığı

Kimlik doğrulama başlığı, düğümler arasındaki iletişimde veri bütünlüğü, veri doğruluğu ve yineleme (replay attack) engeli sağlar. Bu başlık, yalnız veya güvenlik verisi paketleme başlığı ile beraber kullanılabilir. Sonraki başlık alanında 51 değeri ile tanımlanır.



Şekil 9. IPv6 kimlik doğrulama başlığı

Kimlik doğrulama ek başlığı içerisinde sırasıyla aşağıda belirtilen alanlar yer alır:

- Sonraki Başlık

  Bu alan kimlik doğrulama başlığından sonra gelecek başlık tipini tanımlar. 8 bit uzunluğundadır.

- Yararlı Veri Boyu

  8 bit uzunluğundaki bu alan kimlik doğrulama başlığının uzunlunu belirtir.

- Rezerve

  Bu alan 16 bit uzunluğundadır. Gelecekte kullanmak üzere rezerve edilmiştir. Varsayılan değeri 0'dır.

- Güvenlik Parametreleri İçeriği (Security Parameters Index)

  32 bit uzunluğundadır. Varış düğümü tarafından güvenlik birliği (security association) sorgulanarak paketin kaynağı kontrol edilir.

- Ardışıklık Sayısı (Sequence Number)

  32 bit uzunluğundaki bu ardışık sayı, monotonic olarak artmakta ve sayaç görevi görmektedir. Bu mekanizma, eş paket verilerinin tekrarlı gönderilmemesini garanti altına alır. Bu sayede yenileme saldırıları tek kaynaklı güvenlik biriliği için engellenmiş olur. İlk gönderilen paket için

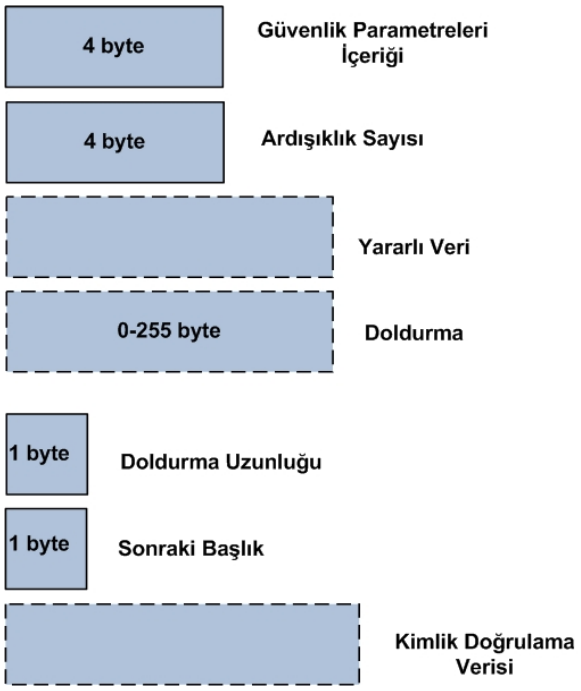
sayaç değeri 1 olarak atanır ve ardıl paketler için 232 değerine kadar birer arttırılır. Sayaç değeri 232'ye ulaştığında sayaç sıfırlanır ve tekrar başlatılır.

- Kimlik Doğrulama Verisi

  Bu alan paketin sağlama değerini (Integrity Check Value) içerir. Seçilen güvenlik birliği algoritmasına bağlı olarak değişken uzunlukta ve her zaman 4 byte'ın katları şeklindedir. Kullanılan hesaplama ve bütünlük algoritmaları RFC 4302 ile detaylı olarak anlatılmıştır.

2.1.2.6. Güvenlik verisi paketleme başlığı

Güvenlik verisi paketleme başlığı, uçtan uca paket gönderiminde bütünlük, gizlilik, kimlik doğrulama, yenileme engeli ve sınırlı olarak trafik akışı gizliliği sağlar. IPv6 başlığındaki sonraki başlık alanı 50 değerine atanarak tanımlanır. Bu başlık yalnız olarak kullanılabildiği gibi "kimlik doğrulama başlığı" ile birlikte de kullanılabilir. Şekil 10'da güvenlik verisi paketleme başlığının yapısı gösterilmiştir.



Şekil 10. IPv6 güvenlik verisi paketleme başlığı

Bu başlık ile ilgili detaylı bilgi RFC 4303'te belirtilmiştir. Güvenlik verisi paketleme ek başlığı içerisinde sırasıyla aşağıda belirtilen alanlar yer alır:

- Güvenlik Parametreleri İçeriği

32 bit uzunluğundadır. Varış düğümü tarafından güvenlik birliği (security association) sorgulanarak paketin kaynağı kontrol edilir. Bu alan güvenlik birliği mekanizmasının sağlanması için bütün güvenlik parametresi içeriği uygulamalarında zorunlu tutulmuştur. Varsayılan olarak bu alanın değeri 0'dır. Gelecekte kullanmak üzere IANA 1-255 arası değerleri rezerve etmiştir.

- Ardışıklık Sayısı (Sequence Number)

  32 bit uzunluğundaki bu ardışık sayı, monotonic olarak artmakta ve sayaç görevi görmektedir. Bu mekanizma, eş paket verilerinin tekrarlı gönderilmemesini garanti altına alır. Bu sayede yenileme saldırıları tek kaynaklı güvenlik biriliği için engellenmiş olur. İlk gönderilen paket için sayaç değeri 1 olarak atanır ve ardıl paketler için 232 değerine kadar birer arttırılır. Sayaç değeri 232'ye ulaştığında sayaç sıfırlanır ve tekrar başlatılır.

- Yararlı Veri (Payload Data)

  Değişken uzunluktaki bu alanın görevi şifrelenmiş veriyi ve şifreleme başlangıç bilgisini (encryption initialization vector) taşımaktır.

- Doldurma (Padding)

  Şifreleme mekanizmasının, minimum paket boyutu gerektirdiği durumlarda, paketin boyutunu arttırmak için kullanılır. 0 ile 255 byte arasında değişen uzunlukta olabilir.

- Doldurma Uzunluğu (Pad Lenght)

  8 bit uzunluğundaki bu alan doldurma miktarının kaç byte olduğunu tanımlar.

- Sonraki Başlık

  Bu alan güvenlik verisi paketleme başlığından sonra gelecek başlık tipini tanımlar. 8 bit uzunluğundadır.

- Kimlik Doğrulama Verisi

  Bu alan paketin sağlama değerini (ICV) içerir. Seçilen güvenlik birliği algoritmasına bağlı olarak değişken uzunluktadır.

2.2. IPv6 Adresleme Yapısı

IPv6 adres uzunluğu 128 bit'tir. Bu durum teorik olarak 2^{128} adet IP adresi var olduğu anlamına gelir. IPv6 adresleri 16'lık tabanda (hexadecimal) ":" işaretiyle

ayrılmış dörtlü gruplar şeklinde yazılmaktadır. Örnek adres gösterimi aşağıdaki gibidir:

2001:DB8:0000:0000:EF01:8765:4321:ABAB

IPv6 adreslerinin yazımında kolaylık sağlamak için bazı kısaltmalar yapılabilir. Örneğin 16 bit'lik bir blokta takip eden 0'lar yazılmayabilir. Bu durumda örnek IP adresi aşağıdaki gibi olmaktadır:

2001:DB8:0:0:EF01:8765:4321:ABAB

Bir başka kısaltma da iki defa ":" kullanılarak ardışık sıfırların gösterilmesi şeklindedir. Bu kural bir IP adresi yazılırken sadece bir defa uygulanabilir. Bu durumda örnek IP adresi aşağıdaki gibi olmaktadır:

2001:DB8::EF01:8765:4321:ABAB

IPv6 protokolünde 3 tip adres yapısı mevcuttur [25]:

- Tek Alıcılı (Unicast)

  Bir düğüme ait bir arayüzü (interface) tanımlar. Tek alıcı adresine gönderilen paket o adres tarafından tanımlanan arayüze iletilir.

- Herhangi bir Alıcılı (Anycast)

  Birden çok arayüze (farklı düğümlere ait arayüzler olabilir) ait adresi tanımlar. Bu adrese gönderilen paket, adres tarafından tanımlanan arayüzlerden birine gönderilir (yönlendirme protokolüne bağlı olarak, muhtemelen en yakın düğüme).

- Çoklu Alıcılı (Multicast)

  Birden çok arayüze (farklı düğümlere ait arayüzler olabilir) ait adresi tanımlar. Bu adrese gönderilen paket, adres tarafından tanımlanan tüm arayüzlere gönderilir.

IPv6'da broadcast adresi yoktur. Broadcast adresine ait işlevler multicast adresleri tarafından gerçekleştirilir.

2.2.1. IPv6 Adres Sınıfları

IPv6 adresi almak isteyen ağ cihazları ve bilgisayarlar için tanımlanmış sınıflarla birlikte özel kullanım için ayrılmış adresler de mevcuttur. Bu adresler özetle aşağıda açıklanmıştır:

- Global Adresler (Global Unicast Address)

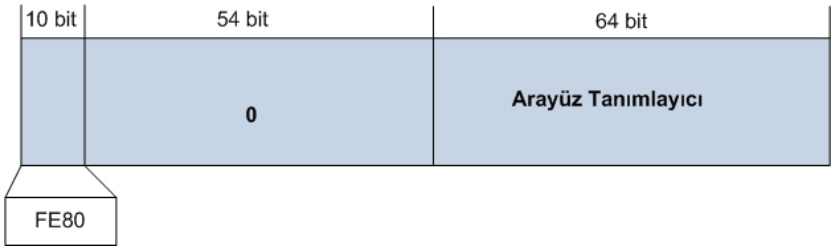
Internet ağına çıkan adresler olarak tanımlanabilir. Bu adresler dış ağlarda yönlendirilebilir niteliktedir. Şekil 11'de global adres yapısı gösterilmiştir.

Şekil 11. IPv6 global adres yapısı

Global adreslerin ilk 3 bit'i "001", global yönlendirme takısı 45 bit ve arayüz tanımlayıcı değeri 64 bit olarak tanımlanmıştır [26].

- Link-Local Adresler (Link Local Addresses)

Bu adresler aynı ağ / bağlantı üzerindeki komşu düğümlerle iletişim için tasarlanmıştır ve asla dış dünyaya yönlendirilmemelidir. Bu adres yapısı global adreslerdeki gibi global yönlendirme takısına ihtiyaç duymaz. Link-local adresler otokonfügirasyon mekanizması ve komşu keşfi (neighbor discovery) için yönlendirici olamayan ağlarda kullanılabilir [25]. Şekil 12'de link-local adres yapısı gösterilmiştir.

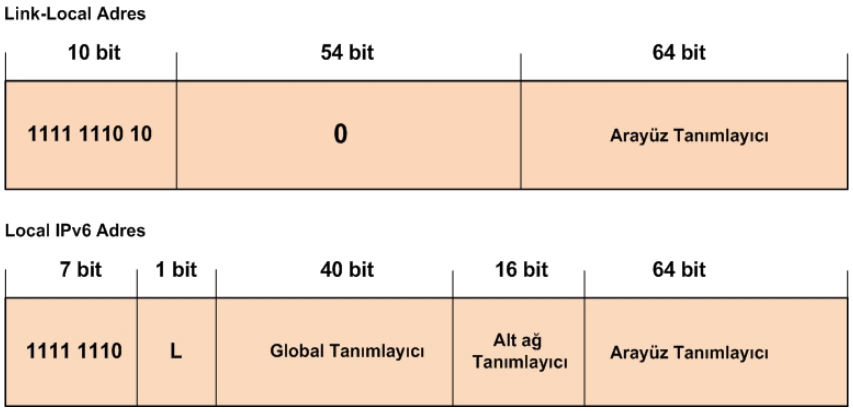


Şekil 12. IPv6 link-local adres yapısı

Bu adres yapısının ilk 10 bit'i hexadecimal "FE80" (1111111010) olarak tanımlanır. Sonraki 54 bit 0 değerinde ve son 64 bit arayüz tanımlayıcısıdır.

- Local IPv6 Adresi (Unique Local IPv6 Unicast Address)

Bu adresler global olarak eşsiz olmalarına karşın dış ağa yönlendirilmeleri beklenmez. Kullanım amaçları aynı ağ veya alandaki düğümler arasında iletişimin gerçekleştirilmesidir. Çok kısıtlı bir alan içinde kalmak koşulu ile yönlendirilebilirler [27]. Şekil 13'de local IPv6 adres yapısı gösterilmiştir.



Şekil 13. IPv6 local adres yapısı

Bu adres yapısının ilk 7 bit'i hexadecimal "FC00" (11111110) olarak tanımlanır. Sonraki 1 bit değeri varsayımlı olarak 1'dir ve 0 değeri gelecekte kullanım için ayrılmıştır. Global tanımlayıcı değeri 40 bit uzunluğunda ve eşsiz global takıyı (global unique prefix) belirtir. Alt-ağ (subnet) tanımlayıcı 16 bit uzunluğundadır. Son olarak arayüz tanımlayıcı değeri 64 bit uzunluğundadır.

Bu adres yapısı, RFC 3513 ile tanımlanan Site-Local adresi yerine geçmiştir [28].

- Özel Adresler

  IPv6'da tüm alanların 0 olduğu adres (0:0:0:0:0:0:0:0) belirtilmemiştir. Bu adres herhangi bir arayüze tanımlanamaz veya varış adresi olarak IPv6 başlığında yer alamaz.

  Bir diğer özel adres "0:0:0:0:0:0:0:1" olarak gösterilen loopback adresidir. IPv4'te bulunan "127.0.0.1" adresi gibi kullanılır.

IPv6 adreslerinden bir kısmı, geçiş yöntemlerinde kullanılmak üzere rezerve edilmiştir. Bu adresler, çalışmanın geçiş yöntemleri kısmında irdelenmiştir.

3. IPv6 GEÇİŞ YÖNTEMLERİ

IPv6'ya geçiş sürecinde hiç şüphesiz v6 / v4 ağları uzun süre birlikte bulunacaktır. Bir gecede tüm IPv4 ağını IPv6'ya geçirmek ya da mevcut uygulamaları IPv6 destekler hale getirmek mümkün olmayacaktır. Bu nedenle, küçük ağlardan çok geniş ağlara kadar IPv6'ya geçiş zor ve karmaşık bir hal alabilir. Planlanmış ve ihtiyaçlar doğrultusunda analiz edilmiş geçiş yöntemleri uygulamak bu sürecin daha sorunsuz geçmesine yardımcı olacaktır. Çalışmanın bu bölümünde, IPv6 geçiş yöntemleri irdelenmiş ve olası avantaj ve dezavantajları araştırılmıştır.

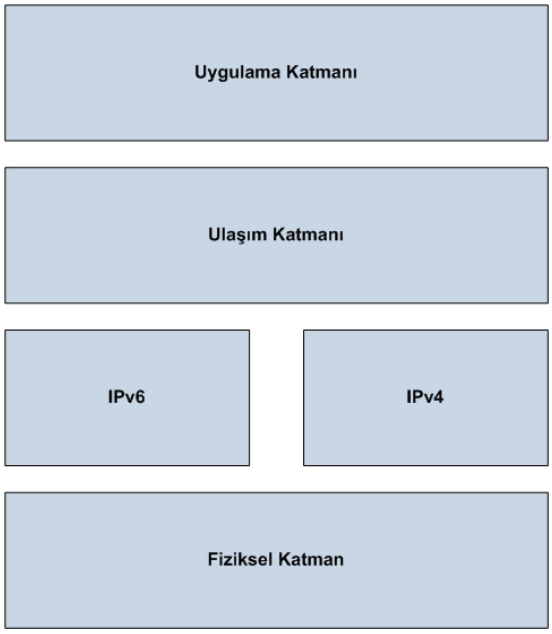
IPv6 geçiş yöntemlerini 3 ana grupta toplamak mümkündür:

- İkili Yığın Yöntemleri (Dual Stack Techniques)
- Tünelleme Yöntemleri (Tunneling Techniques)
- Çevirici Yöntemleri (Translation Techniques)

Bu yöntemler ihtiyaca göre yalnız veya birlikte kullanılabilir.

3.1. İkili Yığın Yöntemi

İkili yığın yöntemi, IPv6 geçiş sürecindeki en kolay yöntemdir. Bu yöntemde tüm ağ cihazları hem IPv4 hem IPv6 adresini aynı anda bulundurur. IPv4 ağı ile iletişim için IPv4 yığını (stack) IPv6 ağları ile iletişim için IPv6 yığını kullanılır. Şekil 14'te ikili yığın yapının mimarisi gösterilmiştir.



Şekil 14. İkili yığın mimarisi

Bu yöntem uygulanırken, farklı iki ağın yönetimi gerektiği göz önünde bulundurulmalıdır. Yani her iki protokol için ağ cihazlarında yönlendirme (routing) yapılmalı, ateş duvarlarında (firewall) gerekli kurallar her iki ağ için tekrar yazılmalıdır. Bu durum ağın yönetim maliyetini arttırmaktadır. Bir diğer önemli nokta, iki farklı yığının aynı anda çalışmasından kaynaklanan işlemci ve hafıza tüketimidir [29].

3.2. Tünelleme Yöntemleri

Tünelleme yöntemleri genellikle mevcut ağın dış ağlarla iletişiminde IPv6 desteğinin olmadığı durumlarda kullanılır. Yani uzaktaki bir IPv6 düğümüyle iletişim, IPv4 ağı

üzerinden taşınarak gerçekleştirilir. Bu işlem, tünelleme veya paketleme (encapsulation) adı verilen, bir protokole ait bilgilerin bir başka protokolün içine paketlenerek gönderilmesi ile gerçekleşir. Tünelleme işlemine ait 3 bileşen vardır:

- Tünel başlangıcında paketleme (encapsulation)
- Tünel bitişinde paket açma (decapsulation)
- Tünel yönetimi

Tünelleme yönteminde 2 adet tünel bitiş noktası gerekmektedir. Genellikle ikili yığın çalışan yönlendirici cihazlar bu görevi üstlenmektedir. Bir tünel 4 farklı şekilde kurulabilir:

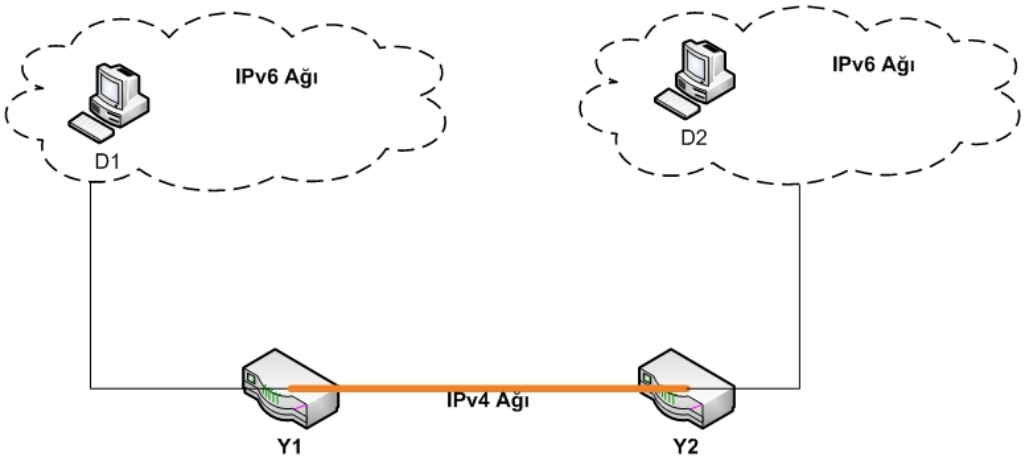
1. Yönlendiriciden-yönlendiriciye (router-to-router), muhtemelen en genel yöntemdir.

2. Düğümden-yönlendiriciye (host-to-router)

3. Düğümden-düğüme (host-to-host)

4. Yönlendiriciden-düğüme (router-to-host)

Tünelleme yöntemleri, otomatik ve elle (manual) yapılandırılmış tünel olarak iki genel başlık altında toplanabilir. Otomatik yapılandırılmış tüneller, tünelin kurulması için bir betiğin (script) çalıştırıldığı durumları ifade etmektedir. Aşağıda kısaca tünelleme işleminin genel çalışma yapısı açıklanmıştır.



Şekil 15. Tünelleme örneği

Şekil 15'te gösterilen D1 düğümü IPv6 ağında bulunmakta ve bir diğer IPv6 ağındaki D2 düğümü ile paket alışverişi yapmak istemektedir. Fakat iki ağa arasındaki bağlantı sadece IPv4 protokolünü desteklemektedir. Bu durumda Y1 yönlendiricisinden Y2 yönlendiricisine tünel açılarak iletişim gerçekleştirilir. Sırasıyla bu işleme ait adımlar aşağıda belirtilmiştir [29]:

1. Y1 tünel başlangıç noktasında IPv6 atlama limiti bir azaltılır, IPv6 paketi IPv4 başlığı içine paketlenir ve tünel boyunca iletilir. Eğer gerekli ise IPv4 paketi parçalanabilir.

2. Y2 tünel çıkışında paketlenmiş IPv4 paket alınır ve kaynak adresi kontrol edilir. Eğer paket parçalanmış ise Y2'de paket yeniden birleştirilir. Sonrasında Y2 tarafından IPv4 başlığı atılarak orijinal IPv6 paketi işlenir. Böylece paket D2 düğümüne iletilmiş olur.

Şekil 16'da IPv6 paketinin IPv4 paketi içerisine paketlenmesi gösterilmiştir.

Şekil 16. IPv4 içine IPv6 paketleme

IPv4 üzerinden tünelleme yapıldığında IPv4 başlığındaki protokol alanı 41 değerini almaktadır. Bu değer paketlenmiş bir veri olduğunu ve içerisinde IPv6 paketi taşıdığını belirtir. IPv4 başlığı kullanımından kaynaklanabilecek parçalanma ve ICMP mesajları ile ilgili kompleks sorunlar RFC 4213'te tartışılmıştır.

3.2.1. Configured tunneling

RFC 4213 ile tanımlanan bu yöntemde, tünel bitiş noktalarına ait IPv4 adresleri karşılıklı olarak tanımlanır ve IPv6 paketleri IPv4 paketleri içersinde taşınır. Tünel bitiş noktaları arasında oluşan bu sanal bağlantı IPv6 ağlarının IPv4 altyapısı üzerinden konuşmasına olanak tanır. Adından da anlaşılabileceği gibi bu yöntemde konfigürasyonlar elle tanımlanır. Yani ihtiyaç duyulan her bir tünel için ayrı yapılandırma gerekir, otomatik tünel oluşturma mekanizması yoktur. Bu durum yönetimsel maliyeti arttırmasına karşın güvenlik nedeniyle tercih edilebilir bir yöntemdir. Configured tunneling yöntemi için "6 in 4" adı da kullanılmaktadır.

Bu yönteme ilişkin filtreleme, ICMPv4, ICMPv6, MTU boyutu, parçalama, komşu keşfi ve güvenlik gibi konular RFC 4213'te tartışılmıştır. Bu çalışmayı ilgilendiren güvenlik konusu bir sonraki bölümde ele alınacaktır.

3.2.2. Automatic tunneling

Bu yöntem, IPv6/IPv4 desteğine sahip düğümlerin, tünel bitiş noktası önceden tanımlanmadan, IPv4 altyapısı üzerinden iletişim kurmasını sağlamaktadır. Tünel bitiş düğümlerinde IPv4-uyumlu IPv6 adresleri (IPv4-compatible IPv6)

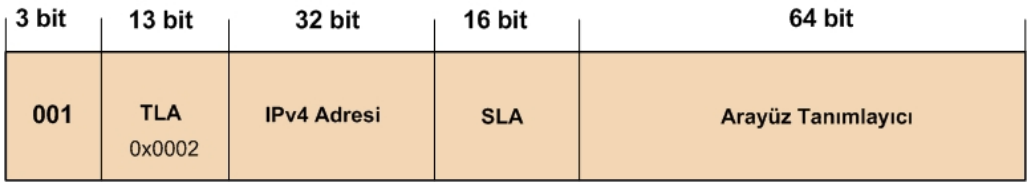
kullanılmaktadır [30]. Böylece alıcı adres, paketlenmiş veri içinde tanımlanmış olmaktadır. Bu yöntem sadece yönlendiriciden-yönlendiriciye ve düğümden-düğüme iletişim olanağı sağlamaktadır.

Automatic tunneling yöntemi ve IPv4-uyumlu IPv6 adresi tanımları RFC 4213 ile kaldırılmıştır. Bu yöntemin kullanılması artık önerilmemektedir.

3.2.3. 6 to 4

IPv6 ağlarının IPv4 ağı üzerinden iletişimi için tasarlanmış bu yöntemde önceden tanımlı tünellere gerek yoktur. 6to4 yöntemi için 6in4 yönteminin otomatik hali demek kısmen doğru olacaktır. IPv6 ağları diğer IPv6 ağları ile konuşurken 6to4 yönlendiricileri kullanmaktadır. Bazı kaynaklarda 6to4 ağ geçidi (gateway) olarak adlandırılmaktadır. Bu yöntemin en önemli avantajlarından biri 6to4 ağında bulunan düğümler için yapılandırma gerektirmemesidir.

IANA tarafından 6to4 yöntemi için 2002::/16 ağı rezerve edilmiştir. Diğer IPv6 ağları ile iletişimin kurulması için 6to4 yönlendiricide en az bir global IPv4 adresinin tanımlı olması gerekmektedir. 6to4 yönlendiricisine ait IPv6 adresi, sahip olduğu IPv4 adresinden türetilir [31]. Şekil 17'de 6to4 IPv6 adres şeması gösterilmiştir.



Şekil 17. 6to4 IPv6 adres şeması

6to4 adres yapısında takı (prefix) uzunluğu 48 bit olarak belirlenmiştir ve formatı 2002:V4ADDR::/48 şeklindedir. İlk 3 bit biçim takısı (format prefix) devamındaki 13 bit TLA (top level aggregator) alanıdır [32]. Sonraki 32 bit'lik kısım, 6to4 yönlendiricinin IPv4 adresine ait hexadecimal değeridir ve son 64 bit arayüz tanımlayıcısını belirtir.

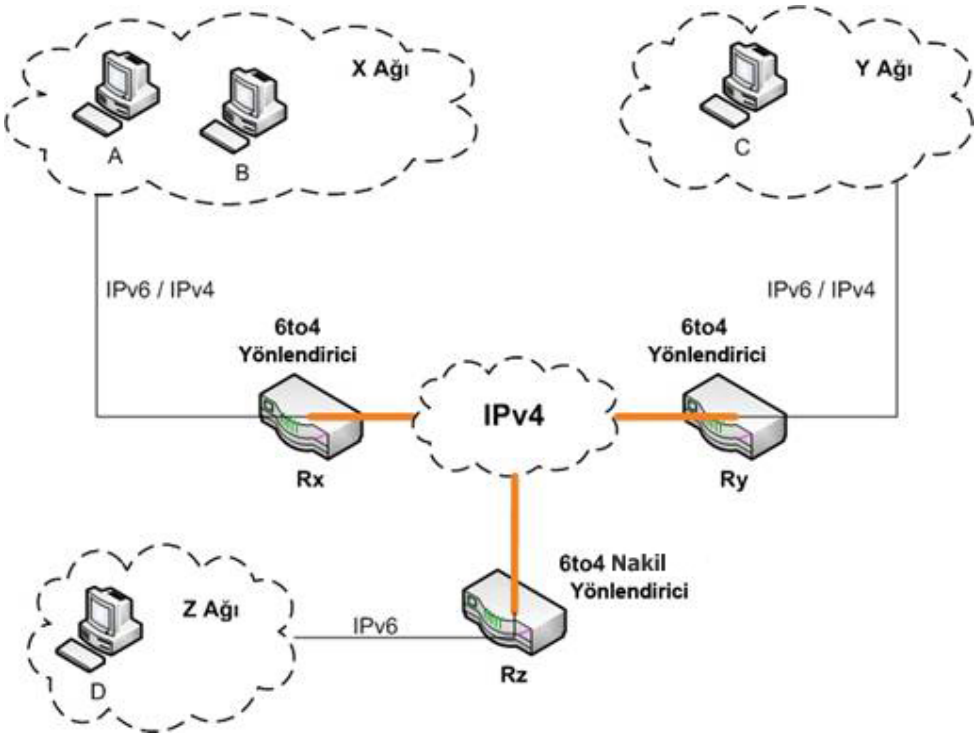
6to4 ağındaki bir düğüm diğer bir 6to4 ağındaki düğüm ile konuşmak istediğinde, elle tünel konfigürasyonuna gerek yoktur. Çünkü tünelin başlangıcındaki yönlendirici IPv6 varış adresinden tünelin bitişine ait IPv4 adresini öğrenmektedir. Bu mekanizma Şekil 18'de gösterilen topoloji üzerinden sırasıyla anlatılmıştır:

- X ağında, A ve B düğümleri kendi aralarında IPv6 kullanarak iletişim sağlayabilir..

- X ağındaki bir düğüm Y ağındaki C düğümü ile iletişim kurmak istediğinde, paket ilk önce Rx yönlendiricinse ulaşır. Rx yönlendiricisi, IPv6 varış adresinden Ry yönlendiricisin IPv4 adresini türetir ve gelen paketi IPv4 paketi içine paketleyerek Ry yönlendiricisine gönderir. Son

olarak Ry yönlendiricisi gelen paketi açarak orijinal IPv6 paketini C düğümüne iletir.

- 6to4 ağındaki herhangi bir düğümün, sadece IPv6 destekli Z ağındaki bir düğümle iletişime geçmesi biraz daha karmaşık bir işlemdir. Bu durumda aradaki bağlantı 6to4 nakil (relay) yönlendirici sayesinde gerçekleşir. 6to4 nakil yönlendirici üzerinde, en az bir 6to4 adresi ve en az bir IPv6 adresi tanımlı olmalıdır. 6to4 ağları arasındaki yönlendirme IPv4 kaynaklı olduğu için herhangi bir tanım yapılmasına gerek yoktur. Fakat yalın IPv6 ağı ile 6to4 ağı arasında, 6to4 nakil yönlendirici tarafından 2002::/16 ağının dış dünyaya anons edilmesi gereklidir. Aynı şekilde dış ağda kullanılan IPv6 bloğu 6to4 ağına yönlendirilebilir. Bahsedilen yönlendirme farkı hariç, Rx ile Rz arasındaki tünelleme işleminde bir değişiklik yoktur.

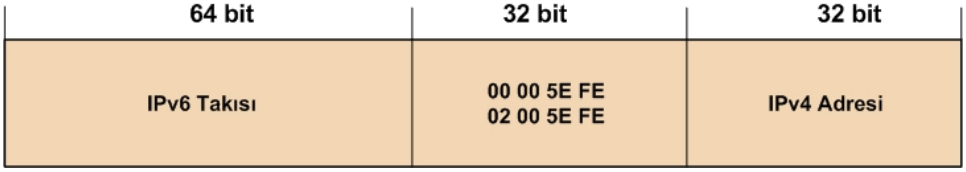


Şekil 18. 6to4 mekanizması

6to4 yönlendiricilerinin, 6to4 nakil yönlendiricilerine ulaşması için varsayılan ağ geçidinin (deafult route) tanımlı olması gerekmektedir. Bu tanımı kolaylaştırmak ve 6to4 nakil yönlendiricilerini daha kolay bulmak için RFC 3068 ile tekli dağıtım adresi tanımlanmıştır. IANA tarafından bu adres için 192.88.99.0/24 bloğu rezerve edilmiş ve 6to4 nakil yönlendiricileri için 192.88.99.1 adresi atanmıştır. Bu sayede 6to4 yönlendiriciler mevcut en yakın 6to4 nakil yönlendiricisine ulaşabilmektedir.

3.2.4. ISATAP (Intra-Site Automatic Tunnel Addresing Protocol)

ISATAP yöntemi, ikili yığın çalışan düğümlerin IPv6 bağlantısını IPv4 ağları üzerinden gerçekleştirmesi için tasarlanmıştır. Bu yöntemde düğümlere ait IPv4 adresleri özel (private) [33] veya global adresler olabilir. ISATAP ağında bulunan bütün düğümler ISATAP desteğine sahip olmalıdır. Şekil 19'da ISATAP adres yapısı gösterilmiştir.



Şekil 19. ISATAP adres yapısı

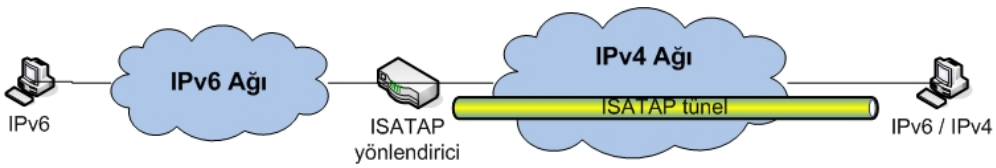
İlk 64 bit'lik kısımda IPv6 adres takısı yer almaktadır. Bu takı link-local, local IPv6, 6to4 takısı veya global IPv6 adreslerinden birini kullanabilir. Sonraki 24 bitlik kısım IANA OUI [34] "00-00-5E" bilgisini taşır ve ardından 8 bit hexadecimal 0xFE değeri gelir. 0xFE değeri, paketin içinde IPv4 adresi olduğunu belirtmek için kullanılır. Son 32 bit ise IPv4 adresini gösterir [35].

Bu tanıma göre 2001:DB8::/64 bloğuna sahip bir ağda IPv4 adresi 192.168.100.56 olan bir düğümün ISATAP adresi 2001:DB8::5EFE:C0A8:6438 olarak atanmaktadır.

ISATAP destekli bir düğüm yapılandırılırken aşağıdaki adımlar izlenir:

1. Düğüme bir IPv4 adresi atanır.
2. ISATAP yönlendiricisine ait IPv4 adresi tanımlanır.
3. ISATAP yönlendiricisinden gelen yönlendirici tavsiyesi (Router Advertisement) [35] mesajlarından IPv6 adresine ait ilk 64 bit'lik takı oluşturulur.
4. RFC 2461 ile açıklanan ISATAP adres yapısına uygun IPv6 adresi oluşturulur.

ISATAP düğümleri IPv4 destekli iç ağda sorunsuzca iletişim kurabilir. Fakat dış ağlarla iletişim kurmak için, ISATAP veya 6to4 yönlendiricisi gereklidir. Şekil 20'de örnek ISATAP topolojisi gösterilmiştir.

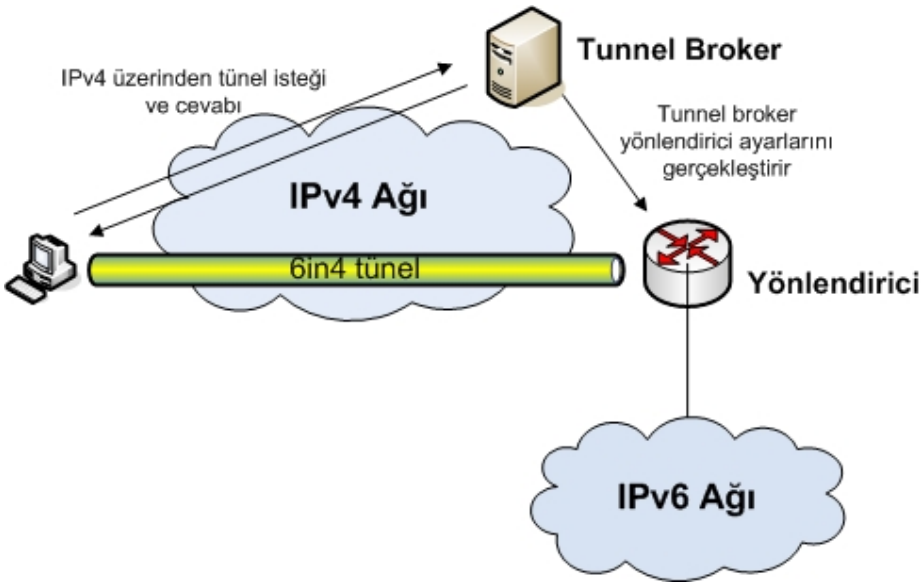


Şekil 20. ISATAP mimarisi

3.2.5. Tunnel Broker

IPv4 bağlantısı olan ikili yığın düğümlerin, IPv6 ağları ile iletişimi için geliştirilmiş bir yöntemdir. Bu yöntem, Configured tunnel mekanizmasına benzer şekilde 6in4 tünelleri oluşturmaktadır. Configured tunnel yönteminden farklı olarak, tüneller elle değil bir betik yardımı ile otomatik olarak oluşturulur.

IPv6 bağlantısı kurmak isteyen düğüm, bu isteğini Tunnel Broker'a iletir. Tünel oluşturulması, silinmesi ve yönetilmesi işlemleri düğüm adına Tunnel Broker tarafından yapılır. Tunnel Broker'ın IPv4 adresi üzerinden ulaşılabilir olması gerekmektedir. Tunnel Broker mekanizması Şekil 21'de detaylı olarak gösterilmiştir.



Şekil 21. Tunnel Broker mekanizması

İstemciler, IPv6 bağlantısı talep etmek için, genellikle bir web sunucusu üzerinden kimlik doğrulama işlemi gerçekleştirirler. Bu web sunucu Tunnel Broker üzerinde de bulunabilir. Sonrasında istemciye tünel oluşturabilmesi için gerekli bilgiler iletilir veya tünel oluşturma işlemini otomatik gerçekleştirecek bir betik yollanır. Aynı zamanda tünel bilgileri Tunnel Broker tarafından tünel sunucusuna veya yönlendiriciye iletilir. Tünel sunucusu, ikili yığın yapıda çalışan ve yönlendirme işlemi yapabilen bir cihaz olabilir [36].

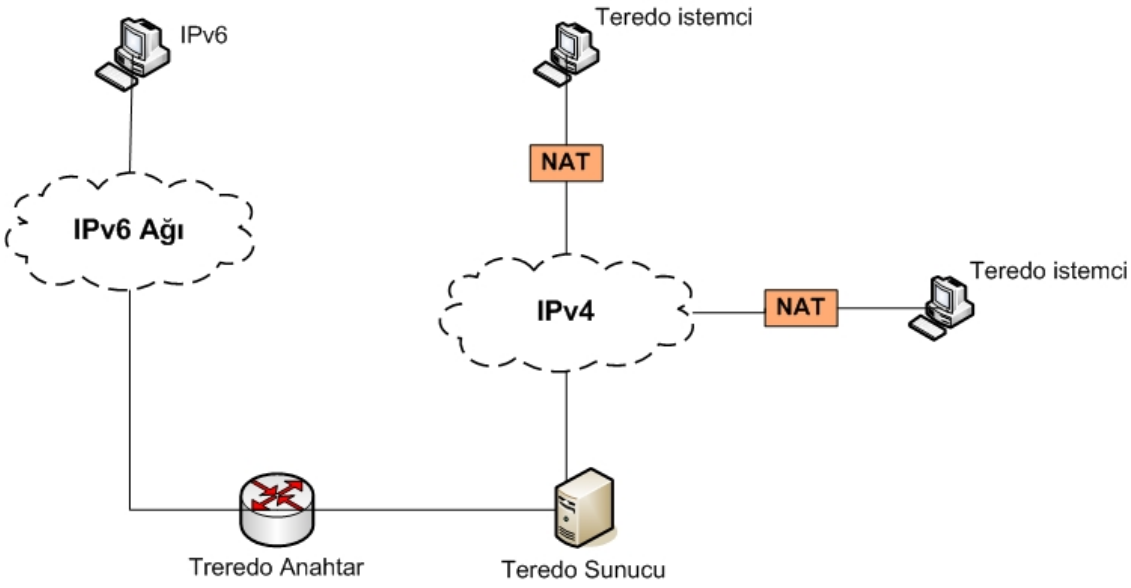
Tunnel Broker yöntemi sadece global IPv4 adresine sahip düğümler tarafından kullanılabilir. Özel bir IPv4 adresine sahip veya NAT arkasında bulunan istemciler Teredo benzeri yöntemler kullanmalıdır.

3.2.6. Teredo

Teredo yöntemi, ikili yığın çalışan ve NAT arkasında bulunan düğümlerin, IPv6 bağlantısını sağlamak için geliştirilmiştir. Bu yöntem düğümden-düğüme otomatik

tünelleme sayesinde bir veya birçok NAT arkasında bulunan istemcilerin IPv6 bağlantısını sağlamaktadır. Bu işlem IPv6 paketlerinin IPv4 UDP (User Datagram Protocol) mesajları içine paketlenmesi ile gerçekleşir [37].

Birçok Internet kullanıcısı bağlantısını NAT arkasından gerçekleştirmektedir. IPv6 protokolünde çok sayıda adres olduğu için NAT gibi bir mekanizma yoktur. Fakat geçiş sürecinde bu konuya çözüm gerekmektedir. NAT yapan cihazların IPv4 yararlı veri yükü alanında filtreleme uygulaması problem oluşturmaktadır. Bunun nedeni, IPv6 paketlerinin tünel içerisinde IPv4 yararlı veri yükü olarak taşınması ile açıklanabilir. Bir diğer sorun NAT arakasındaki adreslerin global olmayışıdır. 6to4 gibi mekanizmalar global IPv4 adresine ihtiyaç duyduğundan bu ortamlarda kullanılması zordur. Ancak NAT yapan cihazın 6to4 yönlendiricisi olması durumunda işleyiş sağlanabilir.



Şekil 22. Teredo mekanizması

Teredo mimarisi Şekil 22'de gösterilmiştir. Bu yönteme ait terimler aşağıda kısaca açıklanmıştır [37]:

- *Teredo Servisi:* IPv6 paketlerinin UDP üzerinden taşınması.
- *Teredo İstemcisi:* IPv6 bağlantısı talep eden düğüm.
- *Teredo Sunucu:* Teredo istemcilere IPv6 bağlantısını sağlayan düğüm.
- *Teredo Nakil (relay):* Teredo istemcilere gelen trafiği, Teredo servisini kullanarak yönlendiren IPv6 yönlendirici.

- *Teredo IPv6 Servis Takısı:* IANA tarafından atanan, Teredo istemcilerine ait IPv6 adres bloğudur (2001:0000::/32).

- *Teredo UDP Port:* Teredo sunucuların dinlediği UDP 3544 portudur.

- *Teredo Balon (Bubble):* Yararlı veri taşımayan, asgari uzunluktaki IPv6 paketi. Teredo nakil ve istemcileri bu paket sayesinde NAT içersinde eşleştirme yapar.

- *Teredo Servis Portu:* Teredo paketlerinin gönderildiği port olarak tanımlanır. Bu port Teredo istemcinin IPv4 adresini kullanarak işlem yapar.

- *Teredo Sunucu Adresi:* Teredo sunucusuna ait IPv4 adresidir.

- *Teredo –eşleme Adresi ve Teredo-eşleme Portu:* NAT sonucu çevrilen global IPv4 adresi ve UDP portu ve Teredo istemciye ait Teredo servis portu bilgileri. İstemci bu bilgileri Teredo protokolü sayesinde elde eder.

- *Teredo IPv6 İstemci Takısı:* Teredo IPv6 servis takısı ve Teredo sunucu adresinden oluşturulan global IPv6 adres takısıdır.

- *Teredo Düğüm Tanımlayıcı (Node Identifier):* Teredo servisi ile erişilebilir bir istemciye ait, 64 bit uzunluğundaki IPv4 adresi ve UDP port bilgilerini taşıyan alandır.

- *Teredo IPv6 Adresi:* Teredo IPv6 istemci takısı ve Teredo düğüm tanımlayıcısı tarafından oluşturulan adrestir.

- *Teredo Yenileme Süresi (Refresh Interval):* Bir Teredo IPv6 adresinin yenilenen trafikte geçerlilik süresini belirtir. Varsayılan değer 30 saniyedir.

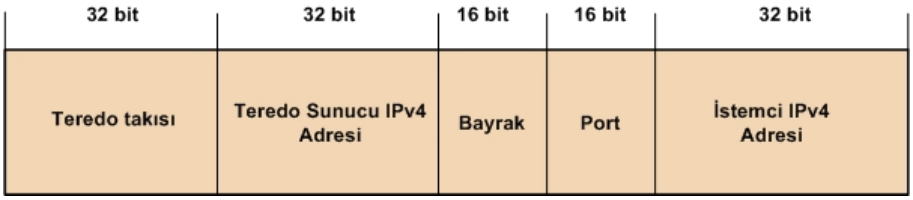
- *Teredo İkincil Port:* Teredo yenileme süresi bilgisinin gönderildiği ve alındığı bir UDP portunu ifade eder. Bu port üzerinden Teredo trafiği geçmez.

- *Teredo IPv6 Keşif Adresi:* "224.0.0.253" olarak tanımlanmış, ağdaki diğer Teredo istemcilerini bulmak için kullanılan IPv4 adresidir.

Teredo işleyişine geçmeden, Şekil 23'de gösterilen Teredo adres yapısını anlamak önemlidir.

Teredo IPv6 adres yapısında, 32 bit uzunluğundaki Teredo servis takısı 2001:0000::/32 olarak rezerve edilmiştir. Sonraki 32 bit'lik alanda, Teredo sunucuya ait IPv4 adresinin bilgisi taşınmaktadır. Takip eden bayrak alanında NAT tipine ait bilgi vardır. Sonraki 16 bit, NAT cihazı tarafından kullanılan global UDP portunu hexadecimal olarak ve gizleyerek (bit flipping) gösterir. Son 32 bit ise Teredo sunucu

tarafından tespit edilen NAT cihazına ait global IPv4 adresinin hexadecimal ve gizlenmiş halidir.



Şekil 23. Teredo adres yapısı

Teredo istemcilerin bu yöntemi kullanabilmesi için önceden yapılandırma gereklidir. İlk adım Teredo istemciye Teredo sunucunsa ait IPv4 adresinin tanıtılmasıdır. İstemci, link-local IPv6 adresinden ağdaki yönlendiricilere yönlendirici isteği (Router Solicitation) [35] mesajı göndererek Teredo sunucuya ait IPv4 adresini öğrenir. Ayrıca, yönlendirici tavsiyesi mesajından da Teredo IPv6 servis takısına ait bilgi elde edilir. Bir sonraki adımda istemci, rezerve edilmiş adres ve port bilgilerinden yararlanarak Teredo IPv6 adresini oluşturur.

Teredo sunucusu, Teredo istemcisinden gelen IPv6 paketlerini UDP içine paketler. Bu işlem sırasında paketin varış düğümüne ait IPv4 adresi ve UDP port numarası IPv6 varış adresinden türetilir. Giden pakette kaynak adresi olarak kendi IPv4 adresini ve kaynak port olarak Teredo UDP portunu (3544) gönderir. Teredo nakil ise Teredo servis takısnı dış dünyaya anons eden bir IPv6 yönlendiricidir.

3.2.7. Dual Stack Transition Mechanism (DSTM)

DSTM, sadece IPv6 bulunan ağlarda IPv4 ağı ile iletişim için tasarlanmıştır. Bu yöntemde IPv4 paketleri IPv6 içine paketlenerek taşınır. DSTM istemcisi sadece IPv6 adresine sahiptir, IPv4 ağındaki bir düğümle iletişime geçmesi için geçici olarak IPv4 adresi alır [38]. Bu yönteme ait elementler aşağıda açıklanmıştır:

- *DSTM Alanı (domain):* IPv4 ağları ile iletişimde DSTM kullanılan, IPv6 / IPv4 düğümlerinin bulunduğu ağ.

- *DSTM İstemci:* DSTM istemci yazılımına ve IPv6/IPv4 desteğine sahip düğüm.

- *DSTM Sunucu:* İkili yığın yapıda çalışabilen, DSTM sunucu yazılımına sahip düğüm. DSTM sunucunun görevi, DSTM istemcilere IPv4 adresi sağlamak ve istemcilerde tünel bitiş parametrelerini (tunnel endpoint parameters) ayarlamaktır.

- *Tünel Bitiş Noktası:* IPv4 paketi taşıyan IPv6 paketlerinin varış noktasını belirtir. DSTM alanı ile istenen IPv4 ağı arasındaki yönlendirme işlemini yapmaktadır. DSTM ağ geçidi olarak tanımlanabilir.

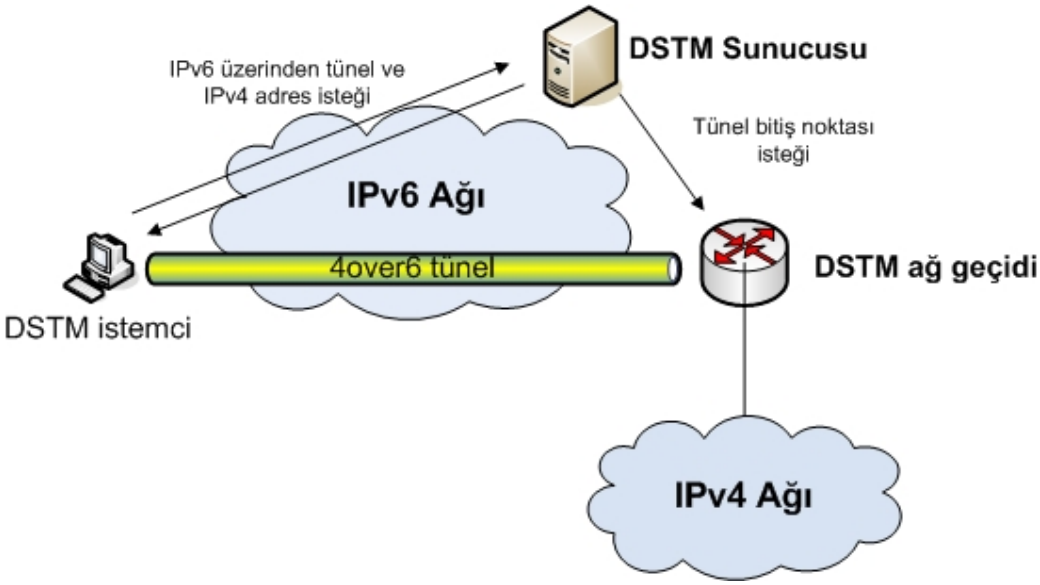
- *4over6 Tünel:* IPv4 paketlerinin IPv6 içine paketlenerek taşınmasıdır.

IPv4 ağı ile iletişime geçmek isteyen DSTM istemci, öncelikle DSTM sunucusundan geçici olarak bir IPv4 adresi isteğinde bulunur. Bu istek IPv6 protokolü üzerinden gerçekleştiği için DHCPv6 [39] gibi bir protokol kullanımı gereklidir.

IPv4 adresi isteğinden sonra DSTM sunucusu, DSTM ağ geçidinden, istekte bulunan istemci için tünel bitiş noktası talebinde bulunur. Tünel bitiş noktası başarılı bir şekilde kurulduğunda, DSTM sunucusu aşağıdaki bilgileri istemciye bildirir:

- Geçici olarak atanan IPv4 adres bilgisi.
- IPv4 adresinin geçerlilik periyodu.
- Tünel bitiş noktasına ait IPv4/IPv6 adres bilgisi.

DSTM sunucusundan istemciye aktarılan bu bilgiler sayesinde, istemci ile ağ geçidi arasında 4over6 tüneli kurulur. DSTM mimarisine ait örnek gösterim Şekil 24'te verilmiştir.



Şekil 24. DSTM mimarisi

3.2.8. Cisco 6PE

Cisco 6PE, IPv6 iletişimi için IPv4 MPLS (Multi Protocol Label Switching) ağları üzerinden haberleşmeye olanak sağlayan bir yöntemdir. Bu yöntemde altyapıda herhangi bir yenilemeye veya yönlendirici cihazların yeniden konfigüre edilmesine gerek yoktur. Çünkü yönlendirme işlemi IP başlığı ile değil, etiket temelli yapılmaktadır. Bu yöntem, "IPv6 over MPLS" olarak adlandırılmaktadır.

MPLS, özetle OSI 2.katmandaki anahtarlama (switching) ve 3. katmandaki yönlendirme işlemlerinin bütünleştirilmesi olarak açıklanabilir. MPLS teknolojisindeki temel amaç, ağın giriş noktalarında bir defa yönlendirme işlemi yaptıktan sonra çıkış noktalarını belirleyip, ağın içinde anahtarlama yapmaktır. MPLS ile ilgili detaylar RFC 3031 ile açıklanmıştır.

IPv6 over MPLS yöntemleri Cisco tarafından detaylı açıklanmıştır [40].

3.2.9. VLAN ile IPv6/IPv4 birlikteliği (VLANs for IPv4-IPv6 Coexistence)

VLAN [41] kullanımı bilgisayar ağlarında oldukça yaygındır. IPv6 desteği olamayan yönlendirici ve anahtarlama cihazlarının bulunduğu bir ağda VLAN kullanarak IPv6 iletişimi sağlamak mümkündür. VLAN sayesinde birçok ağ tek bir ağ üzerinden, Ethernet çerçeveleri (frame) etiketlenerek taşınabilir. Bu noktadan hareketle RFC 4554'te, IPv6 paketlerinin IPv4 ağında taşınması açıklanmıştır.

3.2.10. 6over4

IPv6 ağlarının IPv4 ağları ile iletişimi için tasarlanmıştır. Bu yöntem, IPv6 link-local adreslerinin IPv4 çoklu dağıtım adresleri üzerinden iletimini tanımlamaktadır. RFC 2529 ile detaylı olarak anlatılan bu yöntem, IPv4 çoklu dağıtım mimarisinin birçok ağda desteğinin bulunmamasından dolayı etkili olarak kullanılamaz.

3.2.11. Diğer tünelleme yaklaşımları

Kapsamlı IPv6 geçiş yöntemlerinin yanı sıra, bir veya birkaç düğümün IPv6 iletişimi kurması için çeşitli tünelleme mekanizmaları da mevcuttur. Aşağıda bu yöntemlere kısaca değinilmiştir.

Proto 41 forwarding [42] NAT arkasında bulunan düğümlerin IPv6 iletişimi kurması için tasarlanmıştır. Yalın IPv6 yönlendiricileri veya 6to4 yönlendiricileri ağda yer alıncaya kadar kullanılabilecek, geçici bir çözüm olarak öne sürülmüştür.

GRE (Generic Routing Encapsulation), configured tünel yöntemi gibi her bir bağlantı için elle yapılandırma gerektirmektedir. Bu yöntemde, IPv6 paketleri IPv4 paketleri içinde taşınmaktadır. RFC 2784 ile detaylandırılan yöntem, NAT arkasında bulunan düğümlerin iletişimini sağlayamaz.

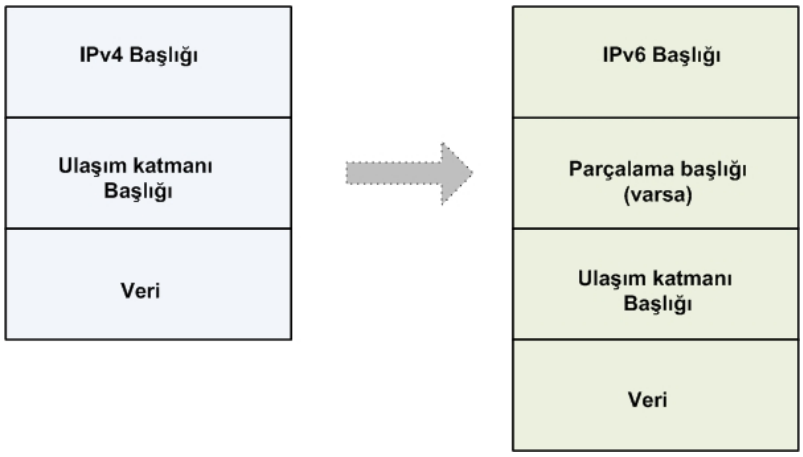
3.3. Çevirici Yöntemleri

Çevirme yöntemleri, sadece IPv6 desteğine sahip düğümlerin sadece IPv4 desteğine sahip düğümler ile iletişimi veya bu durumun tersi için tasarlanmıştır. Çevirme terimi ile vurgulanmak istenen, adres ve protokol dönüşümüdür. Bu dönüşümler IPv4 başlığı ile IPv6 başlığı arasında veya IPv4 adresi ile IPv6 adresi arasında olabilir. Çevirme işleminin hangi OSI katmanında yapıldığına bağlı olarak değişik yöntemler mevcuttur.

3.3.1. SIIT (Stateless IP/ICMP Translation)

OSI ağ katmanında çalışan bu çevirici yöntemine "başlık çevirici" demek mümkündür. Bu yöntem, IP ve ICMP paketlerine ait başlıların her iki protokol tarafından anlaşılabilir olması için çevirme işlemi yapmaktadır. TCP ve UDP başlıkları genellikle çevirici tarafından değiştirilmez. Bu duruma istisna olarak, UDP sağlama işlemi gerektiren IPv6 paketleri ve ICMPv6 sağlama işlemi gerektiren ICMPv4 paketleri gösterilebilir [43].

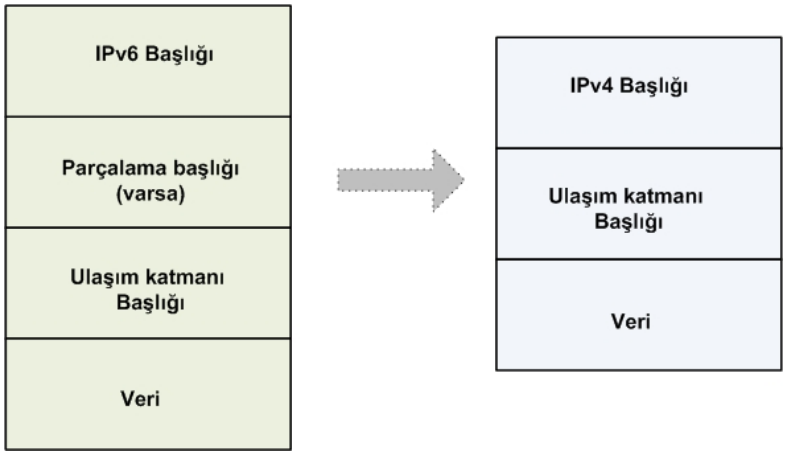
IPv4-to-IPv6 çeviriciler, varış adresi bulunulan IPv4 ağının dışındaki bir hedef için IPv4 paketi aldığında, bu pakete ait başlığı, IPv6 başlığı haline dönüştürürler. Sonrasında bu paket, IPv6 varış adresine gönderilir. Bu işleme Şekil 25'te gösterilmiştir.



Şekil 25. IPv4 – IPv6 başlık çevirme işlemi

ICMPv6 protokolü gereği, tüm ICMP mesajları için, çevirici tarafından sağalama alanı güncellenmektedir. Ayrıca tüm ICMP paketlerine ait "tip" alanına ve ICMP hata mesajında yer alan IP başlığına çevirme işlemi uygulanmaktadır.

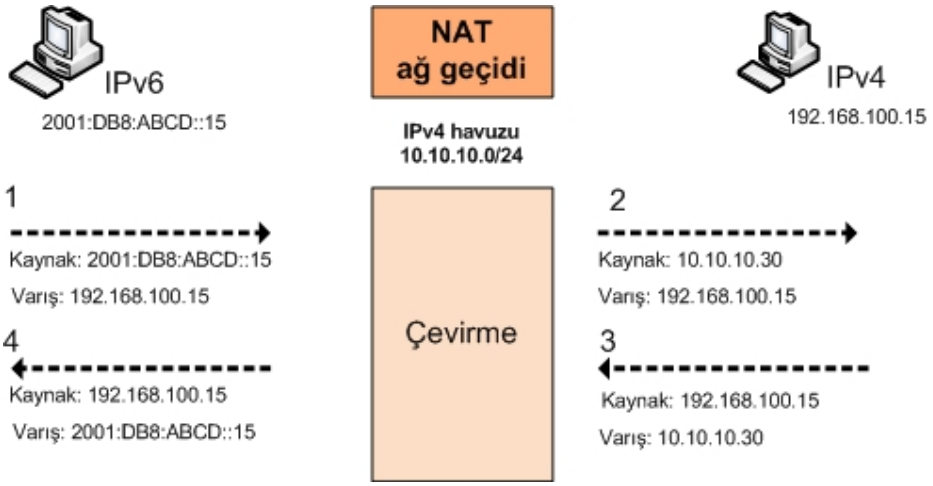
IPv6-to-IPv4 çeviriciler de tersine benzer şekilde, IPv6 başlığını atıp yerine IPv4 başlığı getirerek IPv6 düğümleri ile IPv4 düğümleri arasındaki iletişimi sağlamaktadır. Bu mekanizma Şekil 26'da açıklanmıştır.



Şekil 26. IPv6 – IPv4 başlık çevirme işlemi

3.3.2. NAT-PT (Network Address Translation-Protocol Translation)

Bu yöntem SIIT mekanizmasının uygulamasıdır. NAT yapan cihaz tarafından global IPv4 adres havuzundan bir adres IPv6 adresi ile ilişkilendirilir. Bu mekanizma Şekil 27'de görselleştirilmiştir.

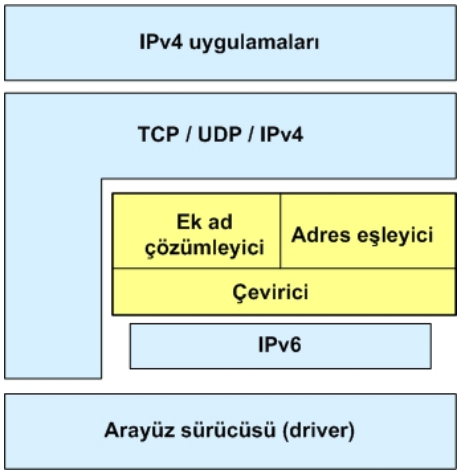


Şekil 27. NAT-PT mekanizması

RFC 2766 ile tanımlanan bu yönteminin, bir sonraki bölümde irdelenen güvenlik sorunları nedeniyle kullanılması önerilmemektedir. Ayrıca NAT-PT başlık çevirimi sırasında bazı alanların çevrilememesi ve her uygulama için ayrı ALG gereksiniminden dolayı efektif bir yöntem değildir.

3.3.3. Bump in the Stack (BIS)

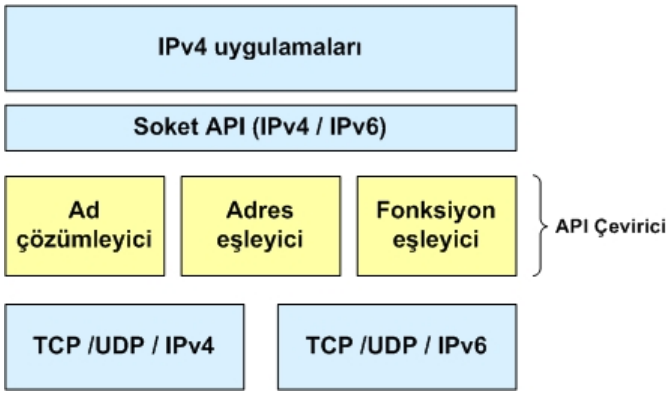
BIS yöntemi temel olarak NAT-PT yöntemi ile benzer işleyiştedir. Aralarındaki fark, BIS yönteminde çevirme işleminin düğümün kendi içinde, yani işletim sistemi seviyesinde gerçekleşmesidir. BIS, IPv4 uygulamaları ile IPv6 ağı arsında bir çevirici arayüz gibi davranmaktadır. Bu yöntemde ağ arayüzünde sadece IPv6 adresi tanımlıdır, yani tüm iletişim IPv6 protokolü üzerinden gerçekleştirilmektedir. BIS, düğüm dışına çıkmayan bir IPv4 adres havuzuna sahiptir ve bu adresleri IPv6 adresleri ile ilişkilendirerek farklı iki protokolün iletişimine olanak sağlamaktadır. İkili yığın yapıdan farklı olarak, ağ arayüzü için IPv4 adresine ihtiyaç duymaz, dışarıdan yalın IPv6 düğümü gibi görünür. Bununla birlikte BIS mimarisinde IPv6 yığını bulunmamaktadır. Kullanılan çevirme algoritması SIIT yöntemi ile aynıdır [44]. Daha çok erken geçiş evreleri için tasarlanmış, sürekli kullanımı önerilmeyen bir yöntemdir. BIS mimarisi Şekil 28'de gösterildiği gibidir.



Şekil 28. BIS mimarisi

3.3.4. Bump in the API (BIA)

BIA çevirici yöntemi BIS yöntemine oldukça benzerdir. Aralarındaki temel fark, BIS yönteminde IP başlıkları çevrilirken BIA yönteminde API (application programming interface) soket (socket) çevrimi yapılır. BIA yöntemi, soket API modülü ile TCP/IP modülü arasına API çevirici eklemektedir. Böylece IPv4 soket API işlevleri ile IPv6 soket API işlevleri arasında dönüşüm yapılabilmektedir. Bu durum, daha karmaşık ve bütünüyle gerçekleştirilemeyen IP başlık çevirimini ortadan kaldırmaktadır [45]. BIA mimarisi Şekil 29'da gösterilmiştir.



Şekil 29. BIA mimarisi

BIA yöntemi ile, IPv4 desteği bulunan fakat IPv6 desteği bulunmayan uygulamaların erişilebilir olması amaçlanmıştır. Daha çok erken geçiş evreleri için tasarlanmış, sürekli kullanımı önerilmeyen bir yöntemdir.

3.3.5. Transport Relay Translator (TRT)

TRT, yalın IPv6 düğümleri ile yalın IPv4 düğümleri arasındaki iletişim için tasarlanmıştır. TRT ikili yığın yapıda çalışır ve farklı protokoller arasında ulaşım katmanı için (TCP, UDP) dönüştürme yapar. IPv6 protokolüne sahip istemcinin IPv4 uygulaması ile iletişimi için tüm paketlerin TRT üzerinden geçmesi gereklidir. Bir TCP bağlantısı kurulduğunda, TRT IPv6 istemcisi ile açılan oturumu kapatarak IPv4 uygulamasına doğru yeni bir TCP oturumu açar. Bu durumda TRT açılan iki oturumu birbirine çevirmektedir [46].

Çevirme yöntemlerinin en son çözüm olarak uygulanması önerilmektedir. TRT yöntemine ilişkin sorunlar güvenlik bölümünde tartışılmıştır.

3.3.6. SOCKS

Bu yöntemi, SOCKS [47] protokolü temelli IPv6/IPv4 ağ geçidi olarak tanımlamak mümkündür. Bu yöntem uygulama katmanında gerçekleşmekte ve yaygın olarak kullanılan vekil (Proxy) sunucu yapısına benzemektedir.

SOCKS mekanizmasında, SOCKS sunucusu ve istemci kütüphanesi olmak üzere iki temel bileşen bulunmaktadır. Sunucu bileşeni uygulama katmanında yer alırken, istemci bileşeni, istemci uygulaması ile ulaşım katmanı arasında yer almaktadır.

Bu yönteme ait detaylar RFC 3089'da anlatılmıştır.

3.4. Bölüm Değerlendirmesi

İkili yığın, kolay ve esnek bir geçiş yöntemidir. IPv4 düğümleri ile iletişim için IPv4 yığını, IPv6 düğümleri ile iletişim için IPv6 yığını kullanılır. IPv4 protokolüne

ihtiyaç kalmadığı zaman, kolayca devre dışı bırakılarak, tamamen IPv6 çalışan bir ağa sahip olmak mümkündür. Modern işletim sistemleri ve firmalara ait birçok yönlendirici çözümleri ikili yığın yapıyı desteklemektedir. Bu nedenle oldukça ekonomik ve esnek bir çözüm oluşturmaktadır. Ayrıca ikili yığın düğümler, tünelleme ve çevirici gibi IPv6 geçiş yöntemlerinin de merkezinde bulunmaktadır. Bu yönteme ilişkin en önemli dezavantaj ağın yönetim maliyetidir. Çünkü her iki protokol için de yönlendirme, anahtarlama ve filtreleme gibi işlemler ayrı ayrı gerçekleştirilmelidir. Ayrıca, düğümler üzerinde iki farklı yığına ait fonksiyonların yerine getirilmesi işlemci ve hafıza gereksinimini arttırmaktadır. Sunucu cihazlarının ikili yığın yapıda çalıştırılması, servis dışı bırakma saldırıları karşısında sunucu kaynaklarının hızla tükenmesine yol açacaktır [22].

Tünelleme teknikleri sayesinde bir veya birçok düğüm IPv6 ağları ile iletişim kurabilmektedir. Tünelleme yaklaşımının önemli bir avantajı, mevcut dış bağlantı ve altyapının herhangi bir değişikliğe ihtiyaç duymamasıdır. Servis sağlayıcıların omurga bağlantısında sadece IPv4 desteği bulunsa bile IPv6 ağları ile iletişim kurulabilmektedir. Tünelleme yöntemlerinin tamamında "paketleme" ve "paket açma" işlemlerinden kaynaklanan, işlemci gücü kaybı ve gecikme süresi artışı meydana gelmektedir. Bu durum, yönlendirici cihazlarının üzerinde yük artışı ile sonuçlanmaktadır. Ayrıca ağda hata tespiti oldukça zor bir hal almaktadır. Çünkü, tünel başlangıç ve bitiş noktası arasında kalan düğümlerden kaynaklanan sorunlar tespit edilemez. Parçalama sorunu veya MTU değeri buna örnek olarak gösterilebilir. Diğer önemli bir sorun ise tünellerin tüm ağı etkileyecek tek arıza noktası (single point of failure) olmalarıdır. Bununla birlikte, ağ yönetimi, filtreleme ve analiz işlemlerinin paketlemeden dolayı yapılması zorlaşmaktadır.

Çevirme yöntemleri, IPv6 geçiş yöntemleri arasından tercih edilecek son ve geçici çözüm olmalıdır. Yani tünelleme veya ikili yığın yapının mümkün olmadığı durumlarda kullanılmalıdır. Çünkü çevirme yöntemleri IPv6 protokolüne ait gelişmiş özellikleri desteklememektedir. Mevcut ağın genişletilmesinde ve yeni topolojilerin oluşturulmasında çevirici cihazın konumu sınırlayıcı olabilmektedir. Bu durumun nedeni, giden ve gelen paketlerin aynı çevirici üzeriden geçmesi zorunluluğudur. Yine tünellerde olduğu gibi çevirici cihazlar da tek arıza noktası olarak ağı etkileyebilmektedir. Ayrıca çevirici yöntemlerine ait bütün RFC belgelerinin "deneysel" veya "bilgi amaçlı" kategorisinde olduğu göz önünde bulundurulmalıdır.

4. IPv6 GEÇİŞ YÖNTEMLERİNDE GÜVENLİK

Günümüzde bilgisayar ve ağ güvenliği hayati öneme sahiptir. E-devlet, sağlık, finans ve eğitim gibi birçok hizmetin Internet üzerinden verilmesi konun önemini arttırmaktadır. Bu bakış açısıyla, yeni bir protokole geçişte mevcut güvenlik politikalarının işlevselliği ve uygulanabilirliği ayrıntılı olarak araştırılmalıdır. Çalışmanın bu bölümünde IPv6 geçiş yöntemlerine ait güvenlik konuları araştırılmış ve çeşitli güvenlik çözümleri önerilmiştir. IPv6 ile ilgili genel güvenlik konuları RFC 4942'de tartışılmıştır ve bu çalışmanın kapsamı dışındadır.

4.1. İkili Yığın Yönteminde Güvenlik

İkili yığın mimarisi, diğer IPv6 geçiş yöntemlerinin de merkezinde bulunduğu için güvenlik açısından önemlidir. Bu nedenle diğer yöntemlerin güvenlik politikaları tasarlanırken bu bölümde irdelenen noktalar dikkate alınmalıdır.

Modern işletim sistemlerinde (Microsoft Vista, Linux, Mac OS X, vb) IPv6 desteği varsayılan olarak açıktır. IPv4 protokolü için gerekli kısıtlama ve güvenlik politikalarını uygulayan ağ yöneticilerinin, IPv6 protokolünün varlığından haberdar olmaması, ikili yığın yöntemi için en önemli güvenlik problemidir. İkili yığın ağların veya düğümlerin Internet bağlantısı bulunuyorsa, IPv4 için alınan bütün güvenlik önlemleri IPv6 için de alınmalıdır. Yazılan tüm erişim kontrol listeleri (ACL) ve ateş duvarı kuralları IPv6 protokolüne uygun olarak dönüştürülmelidir. IPv6 ve IPv4 topolojilerinin birbirinden farklı olduğu ağlarda bu işlemler daha karmaşık ve daha zor bir hale gelebilmektedir.

IPv4 ağının NAT arkasında bulunduğu yani Internet bağlantısı için global adresinin olmadığı durumlarda, yönlendirilebilir global IPv6 adresleri sayesinde dışarıya erişim sağlanabilir. Böyle bir bağlantı tünelleme yöntemleri ile gerçekleştirilebilir. Bu durumda, NAT yapan cihazlar üzerinde, filtreleme ve erişim önlemlerin alınması gerekmektedir.

Diğer önemli bir nokta ise, IPv6 desteği bulunmayan ağlarda bile, ikili yığın düğümlerin IPv6 saldırılarına açık olmasıdır. Bu durum aşağıdaki örnekle izah edilmiştir.

- Saldırgan, varsayılan IPv6 desteğinin açık olduğunu bildiği aynı yerel ağdaki düğümün yönlendirici isteği (RS) mesajına yönlendirici tavsiyesi (RA) ile cevap verir. Bu durumda otomatik adres yapılandırması (stateless address autoconfiguration) [48] ile hedef düğüm IPv6 adresini oluşturur.
- Bir sonraki adımda hedef düğüm, oluşturduğu IPv6 adresinin eşsiz olduğunu doğrulamak için DAD (Duplicate Address Dedecrion) [49] mesajı gönderir. Saldırgan bu mesajla kurbanın IPv6 adresini öğrenir.

Bu saldırı mekanizması sadece saldırganla kurbanın aynı yerel ağda bulunması durumunda uygulanabilir. Kurban düğüm, sadece IPv4 için güvenlik prosedürlerini uyguladıysa, bu mekanizma sayesinde IPv6 üzerinden saldırılara açık olmaktadır. Var olan özelliklerin aktive edilerek kullanılması, gizli IPv6 saldırılarının temelini oluşturmaktadır.

Diğer yandan IPv6 özelliklerinin aktive edilerek kullanılması, IPv6 yığının IPv4 yığınına göre saldırıya daha açık olması durumunda önemlidir. Bu çalışmanın yapıldığı tarihte, aşağıda belirtilen durumlardan ötürü, IPv6 ağları saldırılara daha açıktır:

- Bazı güvenlik çözümlerinin (ağ saldırı tespit sistemleri (NIDS), ateş duvarları ve kişisel güvenlik yazılımları) ve NAT cihazlarının IPv6 desteğinin olmayışı.
- IPv6 desteğine sahip güvenlik çözümlerinin, nasıl yapılandırılacağının bilinmemesi veya IPv6'nın varlığından bihaber olunması.
- Uygulamalara IPv6 desteği verilirken eklenen kodlarda, çeşitli güvenlik açıkları bulunabilmesi.

Stabil ve güvenli seviyede bulunan işletim sistemleri ve değişik uygulamalara, IPv6 desteği için kodlar eklenmesi, güvenlik sorunu oluşturabilmektedir. 2008-2009 yılları arasında tespit edilen ve Mitre CVE tarafından yayınlanan, IPv6 ile ilgili güvenlik problemleri ve yazılım hataları Çizelge 1'de gösterilmiştir.

CVE	Tarih	Uygulama / OS
2009-3641	10/28/2009	Snort
2009-3164	09/10/2009	Sun Solaris10, OpenSolaris
2009-2698	08/27/2009	Linux çekirdeği
2009-2208	06/25/2009	Freebsd 6.3, 6.4, 7.1, 7.2
2009-2187	06/25/2009	Sun Solaris10, OpenSolaris
2009-1906	06/03/2009	IBM DB2
2009-1360	04/22/2009	Linux çekirdeği
2009-0634	03/27/2009	Cisco IOS 12.3 - 12.4
2009-0633	02/04/2009	Cisco IOS 12.3 - 12.4
2009-0418	02/04/2009	HP –UX B.11.(11-23-31)
2009-0304	01/27/2009	Sun Solaris10, OpenSolaris
2008-3816	10/23/2008	Cisco ASA 5500, PIX 7.2.4.(9-10)
2008-4404	10/03/2008	IBM zSeries servers
2008-2476	10/03/2008	FreeBSD, OpenBSD, NetBSD, Force10 FTOS, Juniper JUNOS ve Wind River
2008-3530	09/05/2008	FreeBSD, NetBSD
2008-3686	08/14/2008	Linux çekirdeği
2008-1576	06/02/2008	Mac OS X
2008-2136	05/16/2008	Linux çekirdeği
2008-2085	05/12/2008	SIPp 3.1
2008-1153	03/27/2008	Cisco IOS 12.(1,2,3,4)
2008-1057	02/28/2008	OpenBSD 4.2
2008-0177	02/07/2008	KAME Project
2008-0630	02/06/2008	MPlayer
2008-0352	01/08/2008	Linux çekirdeği

Çizelge 1. IPv6 ile ilgili güvenlik problemleri ve yazılım hataları (2008-2009)

Tablo 1 incelendiğinde, işletim sistemleri, yönlendiriciler, güvenlik cihazları ve güvenlik yazılımlarının IPv6 yığını nedeniyle güvenlik zafiyetine uğradığı görülmektedir.

İkili yığın düğümlerini korumak için, aşağıda sıralanan bazı önlemler dikkate alınmalıdır [50]:

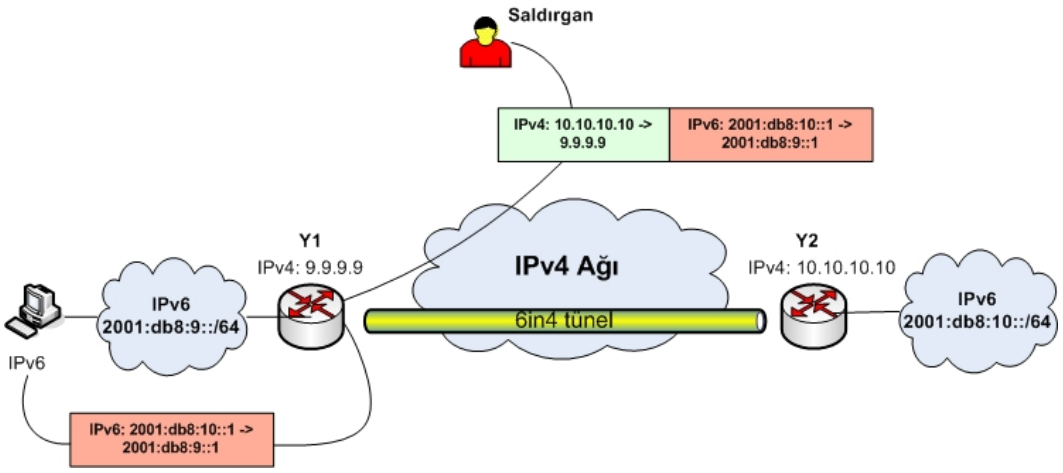
- IPv6 destekli, kişisel ateş duvarlarının kullanılması.
- Ağın tamamen yalın IPv6'ya geçirilmesi ve güvenlik politikalarının bu doğrultuda planlanması.
- IPv4 ağlarında, IPv6 kaynaklı saldırıları engellemek için anahtarlama cihazlarında 0x86dd Ethernet tipinin engellenmesi.
- Varsayılan durumu aktif olan IPv6 yığınının kapatılması.
- IPv6 ile ilgili güvenlik yamalarının takip edilmesi ve güncelleştirmelerin uygulanması.

4.2 Tünelleme Yöntemlerinde Güvenlik

Genel olarak tünelleme yöntemi kullanan tüm geçiş mekanizmaları ağda güvenlik sorunları oluşturmaktadır. Çünkü tünelleme mekanizmalarının hiçbirinde kimlik doğrulama, bütünlük kontrolü ve gizlilik öğeleri yerleşik olarak bulunmaz. Erişim kontrol listeleri ve ateş duvarı kurallarıyla korunan bir ağa, başlığı bu kurallara uyan bir paket tünel içerisinden gönderilirse, iç ağa erişim sağlanabilir. Çünkü bu filtreler paketin başlığına bakar fakat içeriğini kontrol etmez. Böyle bir trafik, korunan ağın içindeki bir tünel bitiş noktasına ulaşırsa, paket açılır ve iç ağa sızan potansiyel bir tehdit oluşabilir. Tünelleme mekanizmaları için oluşabilecek genel sorunlar aşağıda anlatılmıştır:

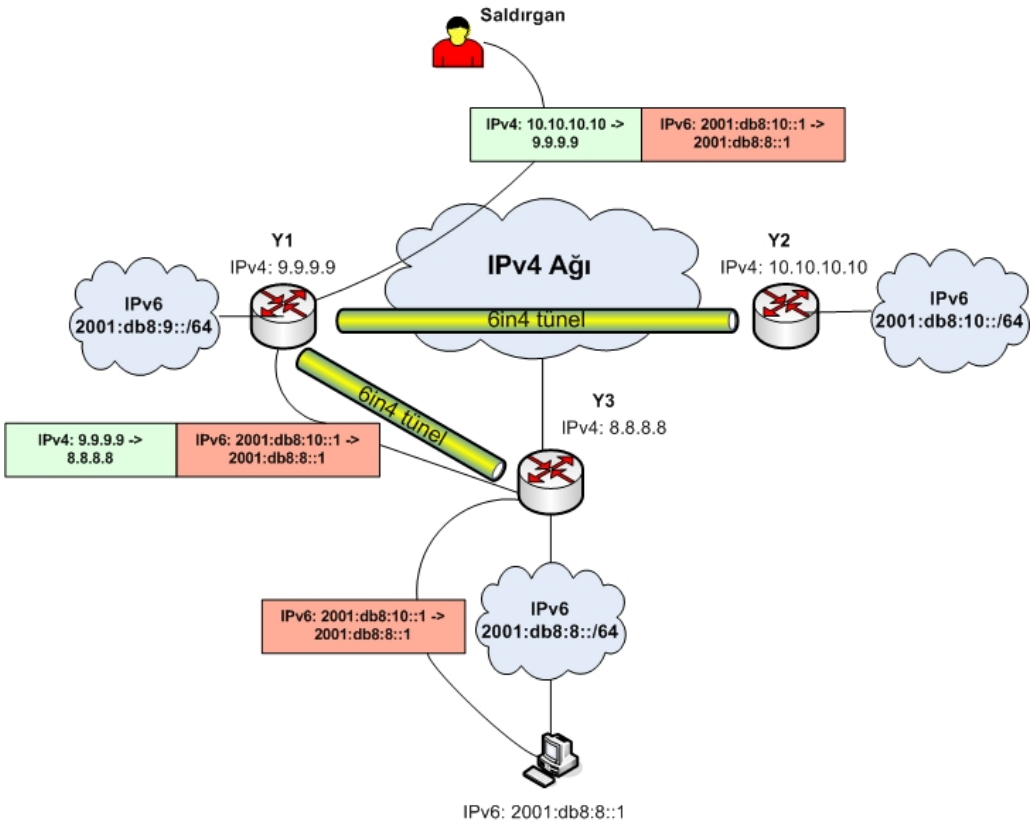
- *Tünel enjeksiyon (tunnel injection):* Saldırgan, IPv4 ve IPv6 adresini değiştirerek (spoof) kendini uygun bir trafik olarak gösterebilir. Bu sayede Şekil 30'da gösterildiği gibi hedef ağa paket gönderebilir.

- *Tünel dinleme (tunnel sniffing):* IPv4 tünel yolu üzerinde bulunan saldırgan, paketlenmiş IPv6 içeriğine erişebilir. Bu durumun tersi de mümkündür.



Şekil 30. Tünel içine paket enjeksiyonu

Tünel enjeksiyon sayısı çoğaltılarak yansılama (reflection) saldırısı oluşturulabilir. Yansılama saldırısı, kaynak adresi kurbanın IP adresi olacak şekilde değiştirilmiş, çok sayıda paketin farklı düğümlere gönderilmesi ile gerçekleşmektedir. Böyle bir paket alan düğümler, bağlantının kurulması için kaynak adresine doğrulama paketleri göndererek cevap verir. Çok sayıda düğümden gelen paketler sonucunda kurbanın ağ ve sistem kaynakları tükenebilir. Bu tarz saldırılara taşırma (flooding) saldırıları da denir. 6in4 tünelleri kullanılarak yapılan yansılama saldırısı Şekil 31 ve 32'de gösterilmiştir. Yansılama, tünel bitiş noktasında gerçekleşebileceği gibi arada bir düğümde de gerçekleşebilir.



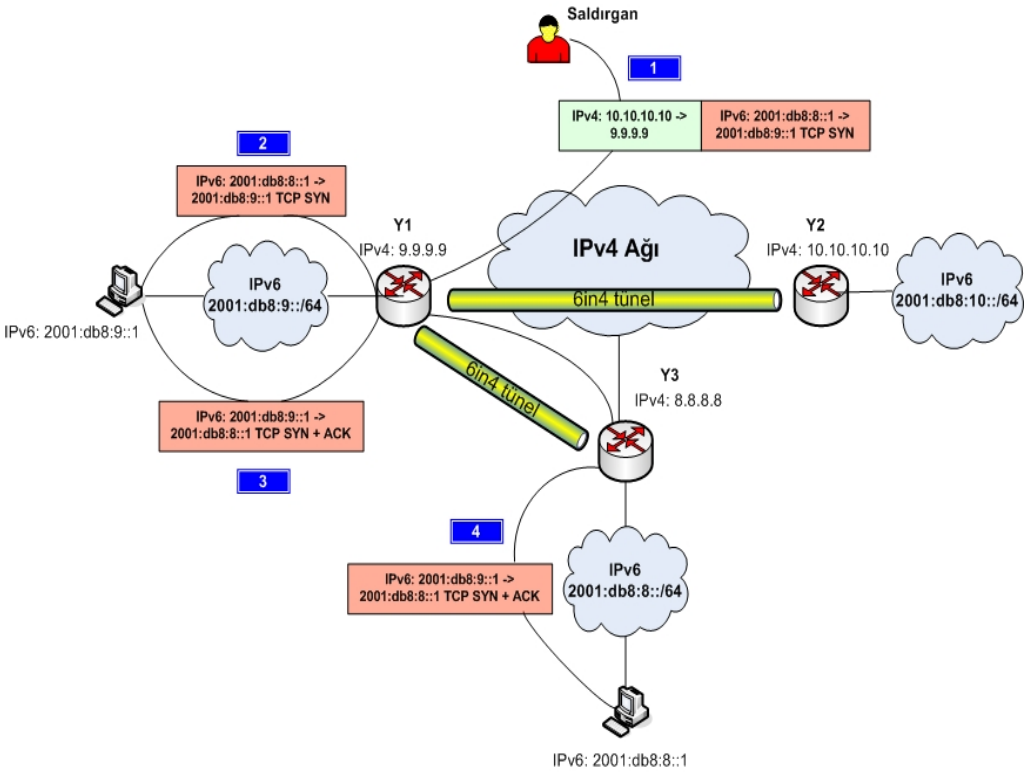
Şekil 31. Yansılama saldırısı

Tünel bitiş noktasında yansılama saldırısı, Şekil 31'de gösterilmiştir:

1. Saldırgan, IPv4 içine paketlenmiş IPv6 içeren paketi oluşturur.

2. Enjekte edilen paket önce Y1 yönlendiricisine ulaşır. Burada paket açılır, yönlendirilir ve tekrar paketlenerek Y3 yönlendiricisine gönderilir.

3. Y3 tünel bitiş noktasında paket tekrar açılarak varış adresine iletilir.



Şekil 32. Yansılama yöntemi ile SYN flood saldırısı

Aradaki bir düğümde yansılama işleminin gerçekleştiği tünel enjeksiyonu, Şekil 32'de gösterilmiştir:

1. Saldırgan, TCP SYN içeren IPv6 paketinin bulunduğu IPv4 paketini oluşturur. Oluşturulan bu paketin IPv6 varış adresi, yansılama yapacak düğümün adresidir.

2. Saldırgan tarafından gönderilen paket Y1 yönlendiricisinde açılarak hedef düğüme gönderilir.

3. İçeriğinde TCP SYN olan paketi alan düğüm, TCP SYN+ACK cevabını, kurbanın IPv6 adresine gönderir.

4. Bu son paket Y1 ve Y3 yönlendiricileri arasındaki tünelden kurbana ulaşır.

Yansılama saldırısına uğrayan kurban, saldırganın IP adresi yerine, meşru bir ağdan gelen paketleri görebilir. Bu durumda saldırgan kendini gizlemiş olur [50].

4.2.1. Configured tunnelling yönteminde güvenlik

Elle yapılandırılmış configured tunnel ve GRE yöntemleri, tünel enjeksiyon ve tünel dinleme saldırılarına hedef olabilmektedir. Saldırgan herhangi bir yerden, tünel bitiş

noktasına 6in4 paketi göndererek ağın içine sızabilir. Bu tarz enjeksiyon saldırılarını engellemek için:

- IPv4 paketlerinde, kaynak adresi tünel bitiş noktalarına ait olamayan paketler engellenmelidir. Saldırgan ağın içine sızmak için, tünel bitiş noktalarına ait adresleri bilmediği sürece 6in4 paketleri gönderemez.

- Geçersiz IPv6 kaynak adresleri tünel bitiş noktalarında engellenmelidir. Bu geçersiz IPv6 adresleri aşağıda verilmiştir:

    - tüm çoklu dağıtım adresleri (FF00::/8)
    - loopback adresi (::1)
    - tüm IPv4 uyumlu IPv6 adresleri (::/96), bu engellemeye DAD adresleri dahil edilmemelidir (::/128)
    - tüm IPv4 eşlenmiş IPv6 adresleri (::ffff:0:0/96)

- Tünellenmiş IPv6 paketlerinin, ayarlanmış arayüzden gelip gelemediği erişim kontrol listeleri ile denetlenmelidir.

- RPF [51] teknikleri kullanılarak, değiştirilmiş adresler engellenmelidir (antispoof).

- IPsec [52] veya GRE gizli anahtar [53] kimlik doğrulama yöntemleri kullanılmalıdır. RFC 4891 ile 6in4 tünellerin IPsec ile güvenli hale getirilmesi incelenmiştir.

Configured tunnel ve GRE yöntemleri elle konfigüre edildiği için, yukarıda bahsedilen güvenlik önlemlerinin alınması oldukça kolaydır. IPsec, uçtan uca trafiği şifrelediği için tünel dinleme saldırılarını engelleyebilir. Fakat ağ içinden adresi değiştirilmiş paketlerin gönderilmesini engelleyemez. Bu tür paketlerin kaynağı solucan veya truva atı benzeri yazılımlar olabileceği gibi kötü niyetli kullanıcılar da olabilir. Bu durumda IPsec ve RPF teknikleri birlikte kullanılmalıdır [29][50].

Genel olarak 6in4 kullanan ağlarda, tünel bitiş noktasından sonra IPv6 desteğine sahip bir ateş duvarı kullanılması önemlidir. Tünel bitiş noktasında açılan paket, ateş duvarı kurallarına göre iç ağa gönderilebilir veya engellenebilir.

4.2.2. 6to4 yönteminde güvenlik

6to4 yöntemi, tünel enjeksiyonu, servis dışı bırakma saldırıları ve yetkisiz kullanım gibi birçok güvenlik problemine sahiptir. Bu yönteme ait iki özellik birçok güvenlik sorununun temelini oluşturmaktadır:

1. Tüm 6to4 yönlendiricileri, diğer 6to4 yönlendiricileri ve 6to4 nakil yönlendiricilerinden gelen IPv4 paketlerini kabul etmek zorundadır.

2. 6to4 nakil yönlendiricileri, IPv6 düğümlerinden gelen trafiği kabul etmek zorundadır.

Yukarıda belirtilen bu iki özellikten ötürü, 6to4 yönteminde güven sağlamak için belli bir mekanizmanın olması gerektiği görülmektedir. Ayrıca IPv6 paketlerinin içeriği ile ilgili kesin bir sınırlama olmayışı önemlidir.

6to4 ağlarında olası güvenlik tehditleri üç genel başlık altında toplanabilir:

1. Servis dışı bırakma saldırıları.

2. Yansılama servis dışı burkama saldırıları (Reflection DoS).

3. Yetkisiz servis / hizmet kullanımı.

Saldırılar hedefe göre 6to4 ağlarına, IPv6 ağlarına ve IPv4 ağlarına diye üç grupta sınıflandırılabilir.

6to4 nakil ve yönlendiricileri kendilerine gelen IPv4 paketlerinin gerçekten 6to4 yönlendiricilerinden mi yoksa herhangi başka bir IPv4 düğümünden mi geldiğini tespit edemez. Bununla birlikte IPv4 düğümlerinden gelen bütün trafiği işlemek zorundadır. Bu özelliklerden hareketle, 6to4 ağına aşağıda belirtilen çeşitli yöntemlerle saldırılabilir:

- Komşu keşfi (ND) mesajlarıyla saldırı

- Adres değiştirilmiş trafik ile saldırı

- 6to4 düğümleri ile yansılama saldırısı

- Yerel IPv4 broadcast saldırısı

6to4 yönteminde, yalın IPv6 ağlara yapılan saldırıların merkezinde 6to4 nakil yönlendiriciler bulunmaktadır. Bütün 6to4 nakil yönlendiricileri için en önemli güvenlik önlemi: IPv4 paketleri açılırken 2002:V4ADDR::/48 adresinde, V4ADDR ile tanımlanan adresin paketin kaynak adresi ile eşleşmesidir. Bu kontrolü sağlamayan paketler atılmalıdır.

6to4 yönteminde IPv4 ağlarını hedef alan saldırılar, yansılama ve adres değiştirme olarak gruplanabilir. Bu saldırılar yalın IPv6, 6to4 veya IPv4 düğümleri tarafından gerçekleştirilebilir.

Genel olarak aşağıda sıralanan önlemler 6to4 ağlarını korumak için kullanılabilir:

- Komşu keşfi mesajları erişim kontrol listeleri ile engellenmelidir.

- ICMP yeniden yönlendirme (redirect) mesajları engellenmelidir.

- Link-local ve site-local adresleri engellenmelidir.
- RFC 1918 ile belirtilen özel IPv4 adreslerinden 6to4 adresi oluşturulamaz. Bu adresler erişim kontrol listeleri ile engellenmelidir.
- 6to4 yönlendiriciler, varış adresi sadece kendi IPv6 bloğu olan paketleri kabul etmelidir.
- Adres değiştirme saldırılarına karşı RPF mekanizması kullanılmalıdır.

RPF mekanizması ve erişim kontrol listeleri yetkisiz kullanım veya paket enjeksiyonunu engelleyemez. Fakat 6to4 ağları arasında adres değiştirme, yansılama saldırıları ve bazı istenmeyen paketleri engelleyebilmektedir.

6to4 yönteminde yerleşik kimlik doğrulama mekanizması olmadığı için yetkisiz servis kullanımının önüne geçmek zordur. Elle yapılandırılmış tünellerin aksine, IPsec kullanarak erişim kontrolü sağlamak, sadece aynı yönetimsel ağda bulunan 6to4 yönlendiricileri arasında mümkündür. Çünkü IPsec mekanizması bazı ortak konfigürasyonların paylaşımını gerektirmektedir.

6to4 IPv6 adresleri global IPv4 adreslerinden türetildikleri için, saldırganların sadece 32 bit'lik bir adresi taramaları düğüm keşfi için yeterli olacaktır. Bu durumda 6to4 arkasındaki düğümleri tespit etmek, IPv4 ağlarında düğüm aramaktan farksız olmaktadır.

6to4 ağlarında güvenlik konusu RFC 3964 ile detaylı olarak açıklanmıştır.

4.2.3. ISATAP yönteminde güvenlik

ISATAP ağlarında IPv4 ve IPv6 güvenliğinin birlikte sağlanması önemlidir. 6to4 ağlarına benzer biçimde ISATAP yönteminde de trafik enjeksiyonu ve yetkisiz kullanım gibi sorunlar mevcuttur.

ISATAP yönlendirici ve sunucuları sadece iç ağdan gelen tünel isteklerine cevap vermelidir. Bunun için IPv4 ateş duvarı kuralı yazmak yeterlidir. Bir diğer önemli güvenlik tedbiri ise trafik enjeksiyonuna karşı protokol 41 paketlerinin filtrelenmesidir. Bu işlem, IPv4 uç yönlendiricisinde, kaynak ve varış adresi bilinen tünel alanlarına ait olmayan protokol 41 paketleri engellenerek gerçekleştirilebilir. Bu sayede hem ISATAP sunucuları hem de ISATAP istemcileri korunabilir. Diğer tünelleme yöntemlerinde olduğu gibi adres değiştirme saldırılarına karşı RPF mekanizması kullanılmalıdır [50][54].

Genellikle DHCP gibi yöntemlerle adres dağıtan ISATAP sunucuları mümkün olduğu kadar iyi yönetilmelidir. İç ağdaki bir saldırgan ISATAP sunucusu gibi davranıp IPv6 adresi dağıtabilir ve ortadaki adam saldırısını gerçekleştirmek için tünel bitiş noktasını belirleyebilir. Ağ yöneticileri PRL (potential router list)

listelerinin güncelliğini kontrol etmeli ve yetkisiz adres dağıtımına karşı önlem almalıdır.

ISATAP adresleri 6to4 adresleri gibi ağ tarama saldırılarına maruz kalabilir. IPv6 adresi 2001:db8:abcd::5efe:0a0a:0a01 olan bir düğümün ISATAP adresi olduğu 0x5efe alanından anlaşılabilir. Bu durumda saldırgan iç ağda 10.10.10.1 – 254 arasındaki düğümleri 2001:db8:abcd::5efe:0a0a:0a01 - 2001:db8:abcd::5efe:0a0a:0aFF adres aralığını tarayarak bulabilir.

IPsec ile paket enjeksiyonu ve dinleme saldırılarına karşı ISATAP ağının IPv4 tarafı korunabilir. Fakat IPv6 tarafı için bu koruma sağlanamaz [50].

4.2.4. Tunnel Broker yönteminde güvenlik

Tunnel broker yöntemine ait olan bütün bileşenler arasında güvenlik sağlanmalıdır. İstemciler ile tunnel broker arasındaki bağlantı SSL gibi mekanizmalarla şifrelenerek gönderilmelidir. Bu sayede tünel kurulumu için gerekli olan kimlik doğrulama verileri ve tünel oluşturma betikleri korunabilir.

Tunnel broker ile tunnel broker sunucu arasındaki iletişim SNMPv3 gibi şifreleme tekniklerini kullanabilen bir protokol yardımıyla gerçekleştirilmelidir. Aradaki bu iletişim şifrelenerek, dinleme saldırıları ile tünel oluşturma mekanizmasına ait verilerin ele geçirilmesi engellenmelidir.

Tunnel broker tarafından istemcilerin tünel oluşturması için bir betik sağlanıyorsa, bu betiğin istemci üzerinde sadece ağ arayüzünde değişiklik yapabilmesi sağlanmalıdır.

Kötü amaçlı istemciler, tunnel broker sunucuna çok sayıda tünel açarak servis dışı bırakma saldırısı gerçekleştirebilirler. Bu durumun önüne geçmek için istemci başına açılabilecek tünel sayısı kısıtlanmalıdır [36].

4.2.5. Teredo yönteminde güvenlik

Teredo düğümleri IPsec mekanizmalarından herhangi bir kısıtlama olmaksızın faydalanabilir. Bu durum şüphesiz ağı daha güvenli hale getirmektedir. Fakat Teredo yöntemi ile ortaya çıkabilecek güvenlik sorunları göz ardı edilmemelidir. Bu sorunlar dört grup altında toplanabilir [37]:

1. *NAT üzerinden sızmak:*

   Teredo servisini kullanan düğümler, bir veya birden çok NAT katmanının arkasından IPv6 erişimi sağlayabilmektedir. Bu erişimin sağlanabilmesi için NAT cihazı üzerinde bulunan ateş duvarının Teredo servislerine izin vermesi gerekmektedir. Yani Teredo istemciler IPv6 üzerinden gelen bütün trafiğe açıktır ve olası saldırıların hedefi olabilmektedirler. Bu soruna çözüm olarak, bütün Teredo istemcilerin kendi üzerinde kişisel ateş duvarı bulundurulması

önerilmektedir. Modern işletim sistemlerinin çoğunda yerleşik ateş duvarı yazılımı mevcuttur ve bu çözüm kolayca uygulanabilir.

Diğer bir güvenlik önlemi, Teredo üzerindeki link-local adreslere ait trafiğin engellenmesidir. Çünkü Teredo servisleri tarafından link-local adresleri kullanılmamaktadır.

Son olarak IPsec servislerinin kullanılması önerilmektedir. Bu sayede istemciler ile Terdeo servisini sağlayan sunucular veya yönlendiriciler arasında bir koruma sağlanabilir.

2. *Teredo servisini arada adam saldırısı (men-in-the-middle) için kullanmak:*

   Saldırgan Teredo istemcilerinden gelen yönlendirici isteklerini engelleyerek, istemciye farklı bir yönlendirici tavsiyesi mesajı gönderebilir. Bu durumda Teredo istemcisi istenenden farklı bir IP adresi alabilir. Bu tip bir saldırı, istemciye erişim imkanı olmayan bir IP adres atayarak servis almasını engellemek veya istemcilere kendi IP adresini ağ geçidi olarak anons edip, üzerinden geçen trafiği dinlemek için kullanılabilir.

   Böyle bir saldırının gerçekleşebilmesi için, saldırganın yönlendirici isteğini engelleyebilecek seviyede DoS yapabilmesi gerekmektedir. Ayrıca, Teredo sunucu adresinin değiştirilebilmesi için kimlik doğrulama mekanizmasının aşılması gereklidir. SSL gibi şifreli kanallar yardımıyla yapılan kimlik doğrulama işlemleri saldırganın işini daha da zorlaştıracaktır.

   IPsec kullanımı adres değiştirme ve trafik dinleme saldırılarını etkili bir şekilde engelleyebilmektedir ve ortadaki adam saldırılarına karşı önlem olarak önerilmektedir.

3. *Teredo servisine yönelik DoS:*

   Teredo servisine yönelik beş çeşit DoS saldırısı yapmak mümkündür.

   - Saldırgan ağın IPv6 kısmında bir Teredo yönlendiricisi gibi davranıp Terdeo IPv6 takısına yönelik yönlendirme yapabilir. Bu saldırı IPv6 yönlendirmesine yöneliktir.

   - Ortadaki adam saldırısında anlatıldığı gibi yönlendirme tavsiyesi mesajları değiştirilirse, istemci IPv4 paketlerini var olmayan bir adrese veya istenmeyen başka bir IPv4 adresine gönderebilir.

   - Teredo istemcileri iletişime geçtikleri en son düğümleri önbellekte (cache) tutar. Saldırgan istemciye çok sayıda Terdeo istemcisinden bağlantı geliyormuş gibi paket gönderdiğinde önbellek doldurulabilir. Bu durumda Teredo istemcileri arasındaki doğrudan bağlantı engellenebilir.

- Saldırgan, yerel eş keşfi işlemi (local discovery procedure) [37] balonunda değişiklik yaparak istemciye gönderebilirse bu işleme yönelik DoS saldırısı gerçekleşebilir.

- Teredo sunucu ve yönlendiricilerine çok sayıda paket göndererek sistem kaynakları tüketilebilir. Teredo sunucular hata korumalı (failover) çalışabildiğinden bir sunucudaki kaynaklar tükendiğinde servisi diğer bir Teredo sunucusu devralabilir. Teredo yönlendiriciler üzerinde istemci başına açılacak oturum sayısı kısıtlanarak önlem almak mümkündür.

4. *Teredo istemcisi olmayan düğümlere DoS:*

   Teredo yönteminde beklenmedik noktalara paket enjeksiyonu yaparak DoS saldırısı yapmak mümkün olabilir. Bu tip saldırılar üç grup altında toplanabilir:

   - Teredo sunucusu yansımla saldırısı için kullanılabilir.

   - IPv6 düğümlerine yönelik DoS saldırılarının Teredo sunucusu üzerinden taşınması.

   - IPv4 düğümlerine yönelik DoS saldırılarının Teredo yönlendiricisi üzerinden taşınması.

   Bu saldırılara karşı genel önlem, Teredo trafiğinin kendine özgü trafik paterninden yararlanmaktır. Teredo trafiği Teredo UDP portu ve IPv6 adres takısı gibi spesifik paternlere sahiptir. Trafiğe ait bu özellikler saldırı durumunda filtreler yazılmasını sağlayabilir.

4.2.6. DSTM yönteminde güvenlik

DSTM mekanizmasına ait tüm işlevsel parçalar, tanımlanacak güvenlik önlemlerinden faydalanabilir. DSTM düğümlerinin ihtiyaç duyduğu IPv4 adreslerini DoS saldırısı ile tüketmek güçtür. Çünkü DSTM intranet yapısında çalışmakta ve çok geniş bir havuzdan IP adresi dağıtabilmektedir.

IPv4 adres dağıtımında DHCPv6 mekanizmasından faydalanan DSTM alanları, DHCPv6 "kimlik doğrulama" mesajları yardımıyla daha güvenli hale getirilebilir. DSTM istemcisi IPv4 adresi aldıktan sonra IPsec servisleri kullanılarak uçtan-uca güvenlik sağlamak mümkündür.

DSTM tünel bitiş noktaları, yerel ağ içinde yer aldıkları için dışarıdan gelebilecek yansılama gibi saldırılara kapalıdırlar [38].

4.2.7. Cisco 6PE yönteminde güvenlik

6PE yöntemi IPv4 MPLS-VPN ile aynı mimariyi paylaştığı için, güvenlik tehditleri ve önlemleri de benzerdir. RFC 4381'de MPLS güvenliği irdelenmiştir. 6PE ağları için alınabilecek güvenlik tedbirleri özetle aşağıda açıklanmıştır:

- *Servis kalitesi yaklaşımı (QoS):* Taşırma (flooding) saldırılarını engellemek amacıyla VPN alanları için servis kalitesi uygulamaları kullanılmalıdır.

- *Güvenli PE:* MPLS ağında görülebilir tek kısım olan PE (provider edge) yönlendiricileri, erişim kontrol listeleri, güvenli kimlik doğrulama ve yetkilendirme yaklaşımlarıyla korunmalıdır.

4.2.8. VLAN ile IPv6/IPv4 birlikteliği yönteminde güvenlik

Bu yöntemin kullanılması ile meydana gelebilecek ilave bir güvenlik problemi yoktur. IPv4 ve IPv6 bağlantıları farklı kanallar üzerinden iletildiği için, iki protokolün aynı kanaldan iletilmesi ile oluşabilecek sorunlar burada yoktur [55].

4.2.9. 6over4 yönteminde güvenlik

6over4 yöntemi paket enjeksiyonu saldırılarına maruz kalabilir. Bu tarz saldırıların önüne geçmek için yönlendirici tarafından erişim kontrol listeleri ile IPv4 çoklu dağıtım adresleri filtrelenmelidir. Organizasyona ait olmayan IPv4 çoklu dağıtım adresleri bu listeler yardımıyla engellenebilir. Ayrıca bilinmeyen kaynaklardan gelen ve IPv4 başlığında proto 41 içeren paketler engellenerek 6over4 enjeksiyon saldırıları engellenebilir [56].

4.3. Çevirici Yöntemlerinde Güvenlik

Bu bölümde, çevirici yöntemlerine ilişkin güvenlik sorunları araştırılmıştır. Çevirici yöntemlerinin IPv6 geçiş çalışmalarının ilk zamanlarında geliştirildiği ve ikili yığın çalışamayan eski cihaz ve işletim sistemleri için tasarlandığı unutulmamalıdır. Daha önceki bölümlerde de belirtildiği gibi çevirici yöntemleri en son seçenek olarak değerlendirilmelidir.

4.3.1. SIIT yönteminde güvenlik

SIIT, bilinen IPv4 ve IPv6 protokolü kaynaklı güvenlik açıklarına bir yenisini getirmez. Ancak, kimlik doğrulama başlığını çevirimi yapılamadığı için bu başlıkla gelen güvenlik özellikleri kullanılamaz [43].

4.3.2. NAT-PT yönteminde güvenlik

NAT-PT yönteminde uçtan-uca IPsec kullanarak güvenliği sağlamak mümkün değildir. Çünkü gelen ve giden bütün trafiğin ALG tarafından işlenmesi, verinin açık olmasına bağlıdır.

IPv4 alanından gönderilen ve değiştirilmiş bir adres içeren paket, IPv6 alanından kurbanın IPv4 adresine doğru yansılama saldırısı oluşturabilir. Bu saldırının tersi de mümkündür.

NAT-PT cihazları ayrıca iki çeşit DoS saldırısına hedef olabilir:

- Saldırgan birçok değiştirilmiş IPv6 adresinden istek göndererek NAT-PT cihazı üzerinde bulunan adres havuzunu tüketebilir.

- ALG cihazları, protokolün kompleks oluşu nedeniyle donanım seviyesinde üretilmemiştir. Yani ALG cihazları, ağ işlemcileri yerine genel amaçlı işlemciler üzerinde çalışmaktadır. Saldırganın ALG cihazına çok sayıda paket göndermesi cihazın kolayca servis dışı kalmasına neden olabilir.

C. Aoun ve arkdaşları, NAT-PT kullanımı ile ortaya çıkabilecek sorunları kapsamlı olarak irdelemişlerdir. Bu çalışma sonucunda RFC 2766 ile açıklanan NAT-PT yöntemi terk edilmiştir [57].

4.3.3. Bump in the Stack yönteminde güvenlik

BIS yönteminde düğümler, IPv4 uygulamaları ile iletişim kurarken IPv4 yığınını, IPv6 uygulamaları ile iletişim kurarken IPv6 yığınını kullanırlar. Bu nedenle her iki protokol, tasarlanmış güvenlik tedbirlerinden yararlanabilir. Ancak, IPv6 düğümleri ile iletişimde IPv4 uygulamaları kullanılırken, ağ katmanı üzerinde güvenlik mekanizmaları çalıştırılamaz. Çünkü, protokol verileri taşınırken kullanılan IP adresleri kriptoludur ve IPv4 – IPv6 arasında çevirim yapılamaz. Bu durumda, IPv6 düğümleri ile konuşan uygulamanın IPv6 destekli hale getirilmesidir [44].

4.3.4. Bump in the API yönteminde güvenlik

BIA yönteminde güvenlik konusu, NAT-PT yönteminde belirtilen konulara benzemektedir. Aralarındaki temel fark, BIA yönteminde çevirme işleminin ağ katmanı yerine soket API katmanında yapılmasıdır. Bununla birlikte NAT-PT'den farklı olarak ağ katmanında IPsec mekanizmalarından faydalanmak mümkündür [45].

4.3.5. Transport Relay Translator yönteminde güvenlik

TRT çeviricileri, IPsec gibi uçtan-uca güvenlik servislerini desteklemez. Ayrıca kaynak adresine göre kimlik doğrulaması yapan uygulamaların (rsh/rlogin vs.) TRT servisi üzerinden geçirilmemesi gereklidir [46].

4.3.6. SOCKS yönteminde güvenlik

SOCKS temelli IPv6/IPv4 ağ geçitleri, SOCKSv5 protokolünü kullanmaktadır ve bu protokole ait tüm güvenlik etmenlerinden etkilenmektedir. SOCKSv5 protokolü ile ilgili güvenlik konuları RFC 1928'de tartışılmıştır.

Bu yöntemde uçtan-uca güvenlik tam olarak sağlanamamaktadır. Çünkü bağlantılar uygulama katmanında kesilerek ağ geçidi üzerinden aktarılmaktadır. Özetle, istemci ile ağ geçidi ve ağ geçidi ile varış düğümü arasındaki bir uçtan-uca güvenlikten bahsetmek mümkündür [58].

4.4. Bölüm Değerlendirmesi

Tünelleme yöntemleri genellikle paket enjeksiyonu saldırılarına hedef oluşturmaktadır. Ayrıca, yerleşik kimlik doğrulama ve yetkilendirme mekanizmalarının olmayışı önemlidir. Statik tünellerin güvenliği, yönlendiriciler üzerinde erişim kontrol listeleri yardımıyla sağlanabilir. Bu açıdan elle yapılandırılan Configured tunnel ve GRE tunnel yöntemleri daha sıkı güvenlik politikaları ile kontrol edilebilir. Kampus ağları gibi büyük ağların geçişinde otomatik tünel yöntemlerinden faydalanmak mümkündür. Otomatik yöntemlere, güvenlik açısından bakıldığında, kimlik doğrulama ve tünel oluşturma mekanizmalarını yürüten sunucuların korunması ön plana çıkmaktadır. Genel olarak tüm tünelleme yöntemlerinde, adres değiştirme saldırılarına karşı RPF mekanizmaları çalıştırılmalıdır.

Çevirici yöntemlerinin, hem güvenlik hem de yönetilebilirlik açısından en son çözüm olarak kullanılması önerilmektedir. Bu yöntemlerde kullanılan çeviriciler DoS saldırılarının hedefi olabilmektedir. Ayrıca, protokoller arasındaki adres çözümleme işlemlerinin DNS mekanizmasına dayanması, bu tür ağlarda DNS sunucuları da saldırı hedefi haline getirebilmektedir.

IPv6 geçiş stratejisi oluşturulurken, hiç şüphesiz güvenlik ve yönetilebilirlik konularına dikkat edilmelidir. Seçilecek geçiş yöntemi, ağın yapısına, topolojisine, ağ üzerinde çalışan servislere ve ağ cihazlarının IPv6 protokol desteğine bağlı olarak belirlenmelidir.

Son olarak, IPv6 kullanan bütün düğümler için, kişisel ateş duvarı ve saldırı tespit sistemi kullanılması önerilebilir.

5. IPv6 GEÇİŞ YÖNTEMLERİNDE PERFORMANS

Ağ üzerinden verilen birçok hizmet ve uygulama ağ performansına doğrudan bağlıdır. Bu bakımdan, hem akademik ağlarda hem de ticari ağlarda performans oldukça önemli bir konudur. IPv6 geçiş yöntemlerinin performans analizinin yapıldığı bu kısımda, tünelleme ve çevirici yöntemlerinin performansı yalın IPv6 ile karşılaştırılmıştır. Ayrıca TRT çevirici yönteminde, mevcut tek nokta hata durumunun giderilmesi için çözüm mekanizma önerilmiştir.

Tünelleme yöntemleri arasından Configured tunnel, 6to4, ISATAP ve TEREDO yöntemleri için performans testleri yapılmıştır. Automatic tunnel ve 6over4 gibi önerilmeyen yöntemler, Cisco 6PE gibi özel ağlar ve doğrudan omurga ağ performansını etkilemeyen diğer yöntemler için testler yapılmamıştır. DSTM yöntemine ait güncel bir uygulama bulunmadığı için bu yöntemin performans değerlendirmesi yapılmamıştır.

Çevirici yöntemlerinden sadece TRT yöntemine ilişkin performans testleri yapılmıştır. BIS ve BIA gibi yöntemler son kullanıcıya ait düğüm üzerinde çalıştığı için genel ağ performansını etkilemeyecektir ayrıca bu yöntemlere ilişkin güncel bir uygulama da bulunmamaktadır. NAT-PT kullanımı ise RFC 4966'da belirtilen güvenlik sorunları ve her bir uygulama için ayrı ALG gereksiniminden dolayı uygun bir geçiş yöntemi değildir. Bu nedenle performansı araştırılmamıştır.

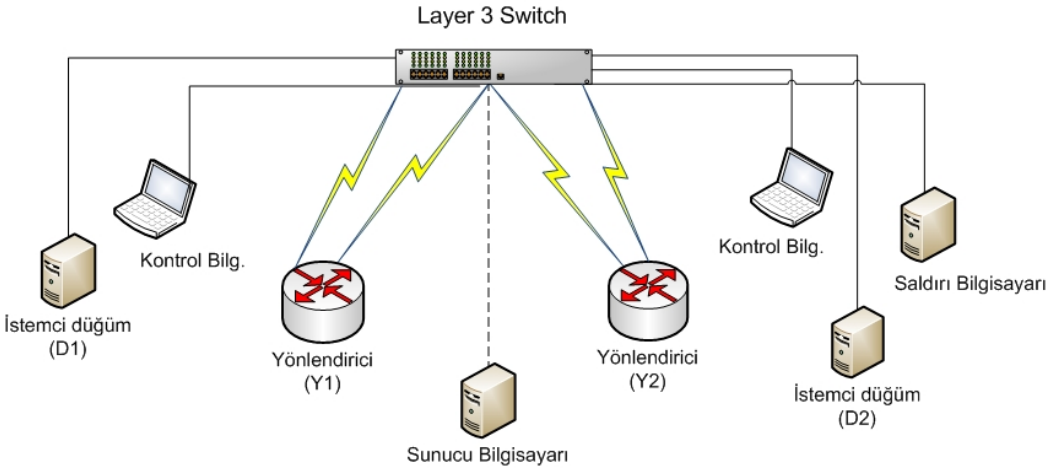
Farklı Cisco yönlendiricilerin, ikili yığın performans testlerine ilişkin detaylı rapor, Cisco tarafından yayınlanmıştır [59]. Bu nedenle, yapılan çalışmada ikili yığın testleri yer almamaktadır.

5.1. Deney Düzeneği ve Ölçüm Süreci

Hazırlanan deney düzeneği, IPv6 geçiş yöntemlerinin througput, işlemci kullanımı ve gecikme süresi değerlerini ölçmeye yöneliktir.

5.1.1. Test ortamı

IPv6 geçiş yöntemlerine ilişkin bütün performans testleri gerçek ağ ortamında yapılmıştır. Şekil 33'te gösterilen test ortamında iki adet Cisco 2600 serisi yönlendirici, iki adet eş özelliklere sahip istemci bilgisayarı, bir adet saldırı bilgisayarı, iki adet kontrol bilgisayarı, bir adet sunucu bilgisayarı ve Cisco 3560 serisi anahtarlama cihazı bulunmaktadır.



Şekil 33. IPv6 geçiş yöntemleri, test ortamı

İstemci, saldırı ve sunucu bilgisayarları eş donanımlara sahiptir. Her birinde Intel Pentium 4 2.66 GHz işlemci, 1024Mb ram bellek, 80GB sata sabit disk ve 1Gbit SkyKonnect ethernet kartı donanımları mevcuttur. Cisco yönlendiricilerde 2611XM (MPC860P) işlemci, 256Mb ram bellek ve iki adet 100Mbit ethernet arayüzü bulunmaktadır. Tüm cihazlar, Cisco 3560 Gigabit switch üzerine CAT5e kablolar ile bağlanmıştır.

Test senaryolarında kullanılan farklı ağlar, Cisco 3560 switch üzerinde oluşturulan VLAN'lar yardımıyla sağlanmıştır.

5.1.2. İşletim sistemleri

Sunucu bilgisayarında FreeBSD 7.2 i386, istemci bilgisayarlarında FreeBSD 8.0 i386, saldırı bilgisayarında ise 2.6.26-2-686 çekirdeğine sahip Debian (4.1.2-25) Linux işletim sistemleri kullanılmıştır. Sunucu ve istemci bilgisayarlar üzerinde yapılan ince ayarlar aşağıda verilmiştir:

```
kern.ipc.somaxconn=5000
kern.ipc.nmbclusters="32768"
```

5.1.3. Ölçüm araçları

IPv6 geçiş yöntemlerinin ağ performansını ölçmek için Netperf [60] uygulaması kullanılmıştır. Bu uygulama, istemci ve sunucu olarak iki farklı yapıda çalışmaktadır. D1 istemcisi üzerinde "netserver -6" parametresi ile sunucu olarak çalıştırılmış, D2 istemcisi üzerinde ise "netperf -P0 -fm -6 –H D1 -tTCP_STREAM -l120 -- -m [74-128]" parametreleri ile istemci modunda çalıştırılmıştır. Netperf çıktıları througput değerlerinin belirlenmesinde kullanılmıştır. Gecikme süresi değerleri ICMPv6 kullanılarak elde edilmiştir.

Yönlendirici cihazlar üzerinde işlemci kullanımı ölçmek için Net::Telnet:Cisco Perl modülü kullanılmıştır. Kontrol bilgisayarları tarafından çalıştırılan bu betikler EK-1'de verilmiştir.

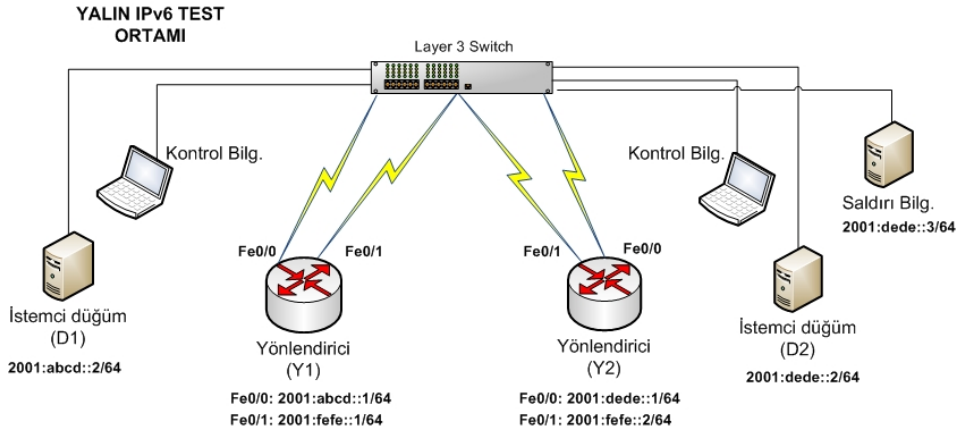
Yapılan testler, IPv6 performans metodolojine [61] uygun olarak 64, 128, 256, 512, 1024, 1280, 1518 ve 8192 byte büyüklüğündeki TCP ve UDP paketleri ile gerçekleştirilmiştir. Her bir test, 120 saniye süresince çalıştırılmıştır.

5.1.4. Ölçüm sonuçlarının anlamlı hale getirilmesi

Kontrol bilgisayarında toplanan ham veriler Perl betikleri yardımıyla işlenmiştir. Sonuçların anlamlı bir hale getirilmesinden sonra Gnuplot [62] uygulaması ile grafikler çizilmiştir.

5.1.5. Yalın IPv6 test düzeneği

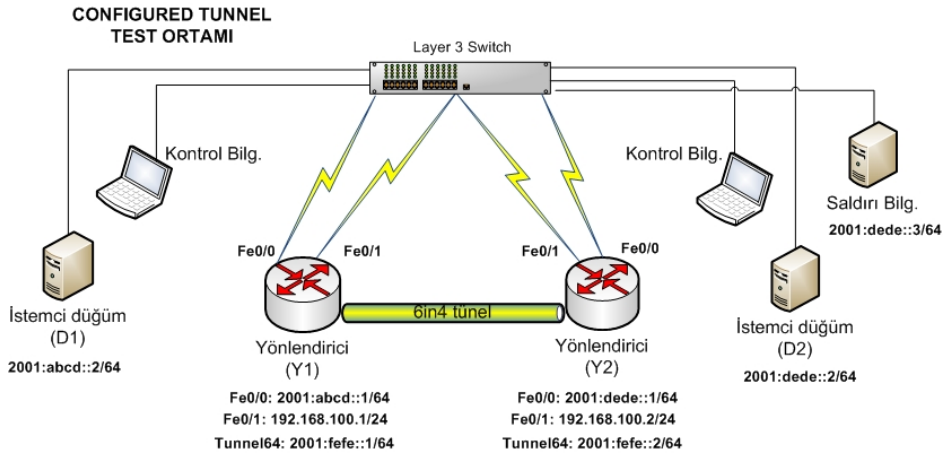
Yalın IPv6 performans testleri yapılırken, tüm düğümler sadece IPv6 ile konfigüre edilmiştir. D1 düğümü üzerinde Netperf uygulaması sunucu modunda, D2 düğümü üzerinde istemci modunda çalıştırılmıştır. Bu test düzeneği Şekil 34'te gösterilmiştir.



Şekil 34. Yalın IPv6 test düzeneği

5.1.6. Configured tunnel test düzeneği

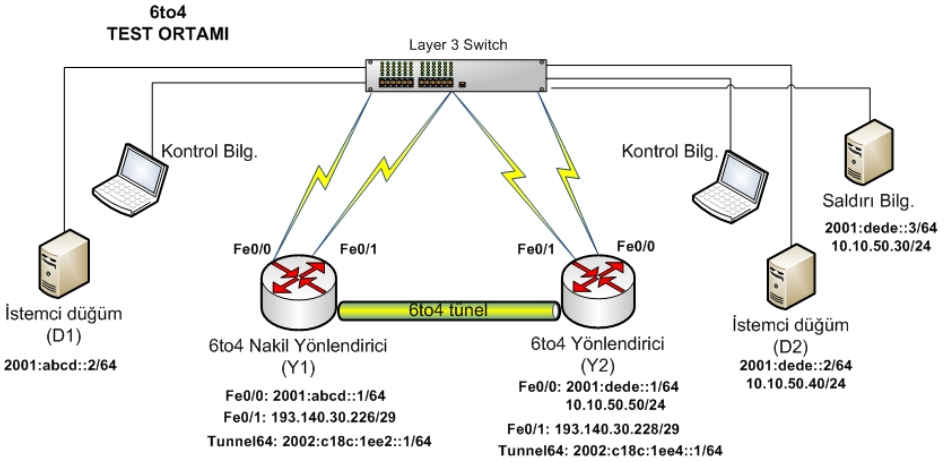
Bu test senaryosunda, farklı iki IPv6 ağının, IPv4 ağı üzerinden iletişimi için statik tünel kurulmuştur. D1 ve D2 düğümü sadece IPv6 adresine sahip, Y1 ve Y2 yönlendiricileri ise hem IPv6 hem de IPv4 adreslerine sahiptir. Kurulan test düzeneği Şekil 35'te gösterilmiştir. D1 düğümü üzerinde Netperf uygulaması sunucu modunda, D2 düğümü üzerinde istemci modunda çalıştırılmıştır.



Şekil 35. Configured tunnel test düzeneği

5.1.7. 6to4 test düzeneği

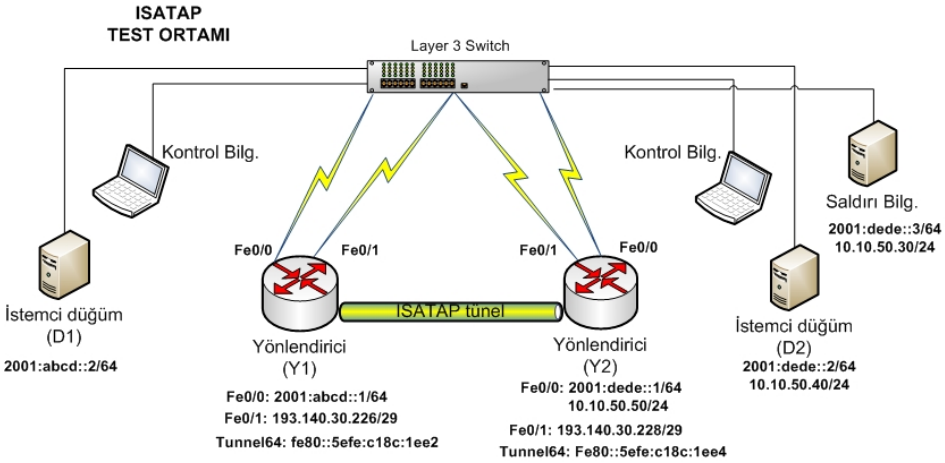
6to4 test senaryosunda, sadece IPv6 adresine sahip D1 ve D2 düğümlerinin, IPv4 ağı üzerinden 6to4 tünel yardımıyla iletişimi sağlanmıştır. Y1 ve Y2 yönlendiricileri hem IPv6 hem de IPv4 adreslerine sahiptir. Kurulan test düzeneği Şekil 36'da gösterilmiştir.



Şekil 36. 6to4 test düzeneği

5.1.8. ISATAP test düzeneği

ISATAP test düzeneğinde, IPv6 ağları arasındaki iletişim ISATAP tünel ile sağlanmıştır. Bu senaryoda Y1 ve Y2 yönlendiricileri arasındaki ağ IPv4 ağıdır. D1 düğümünde sadece IPv6 adresi, D2 düğümünde ise hem IPv4 hem de IPv6 adresi mevcuttur.

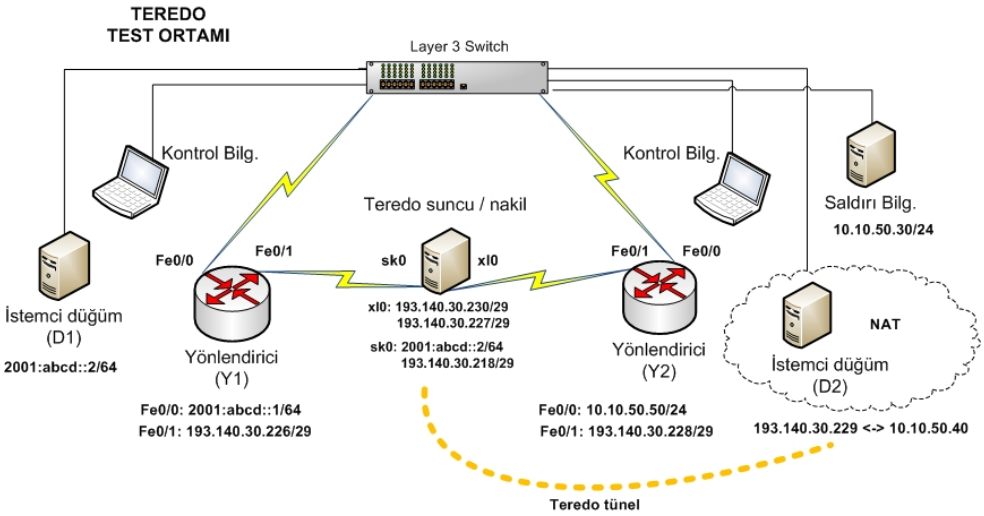


Şekil 37. ISATAP test düzeneği

D1 düğümü üzerinde Netperf uygulaması sunucu modunda, D2 düğümü üzerinde istemci modunda çalıştırılmıştır. ISATAP test düzeneği Şekil 37'de gösterilmiştir.

5.1.9. Teredo test düzeneği

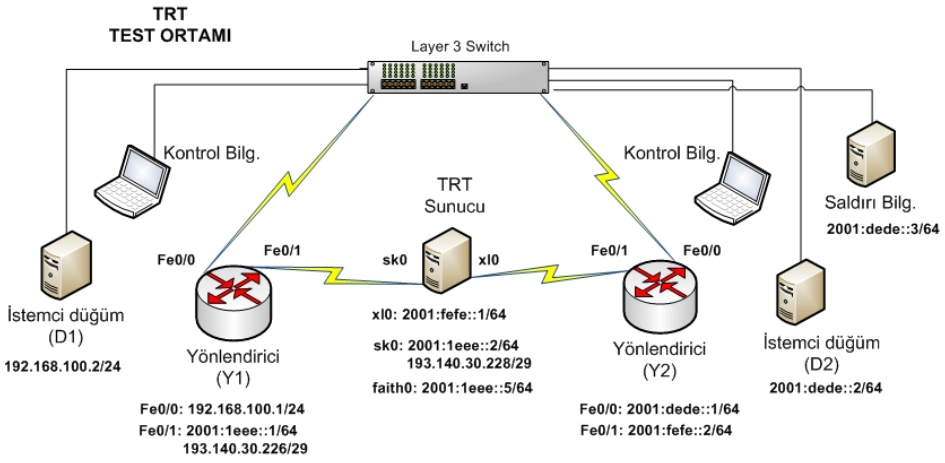
Bu senaryoda, IPv4 ağında ve NAT arkasında bulunan D2 düğümü, sadece IPv6 adresine sahip D1 düğümü ile iletişim kurumuştur. NAT işlemi Y2 yönlendiricisi tarafından gerçekleştirilmiştir. Teredo sunucusu aynı zamanda Teredo nakil yönlendiricisi olarak çalıştırılmıştır. Teredo sunucu / nakil FreeBSD 7 işletim sistemine sahip ve Miredo [63] yazılımı çalıştıran bir bilgisayardır. Miredo yazılımı aynı zamanda D2 istemcisi tarafından Teredo istemci uygulaması olarak çalıştırılmıştır. Şekil 38'de Teredo test ortamı ayrıntılı olarak gösterilmiştir.



Şekil 38. Teredo test düzeneği

5.1.10. TRT test düzeneği

TRT senaryosunda sadece IPv6 adresine sahip D2 düğümü, IPv4 ağındaki D1 düğümü ile iletişimini TRT sunucusu üzerinden gerçekleştirmiştir. Şekil 39'da görüldüğü gibi D2 düğümü IPv6 ağında, D1 düğümü IPv4 ağında ve TRT sunucusu ikili yığın yapıda çalışmaktadır. TRT sunucusu, FreeBSD 7 işletim sistemine ve iki adet ethernet arayüzüne sahiptir. İletim katmanında protokoller arasında çevirme işlemi "faithd" [64] uygulaması tarafından gerçekleştirilmiştir. Diğer yöntemlerde kullanılan Netperf uygulaması faith yazılımı tarafından desteklenmediği için bu senaryoda kullanılmamıştır. Netperf yerine D1 istemcisinde FTP sunucusu çalıştırılmış ve D2 istemcisi tarafından 200Mb büyüklüğündeki bir dosya transferi gerçekleştirilmiştir.



Şekil 39. TRT test düzeneği

5.2. Test Sonuçları

IPv6 geçiş yöntemlerine ait performans testleri, normal durumda ve saldırı altında olmak üzere iki aşamada gerçekleştirilmiştir. Saldırı altında test yapılması, sistemlerin stres altında davranışlarını ortaya koymaya yöneliktir.

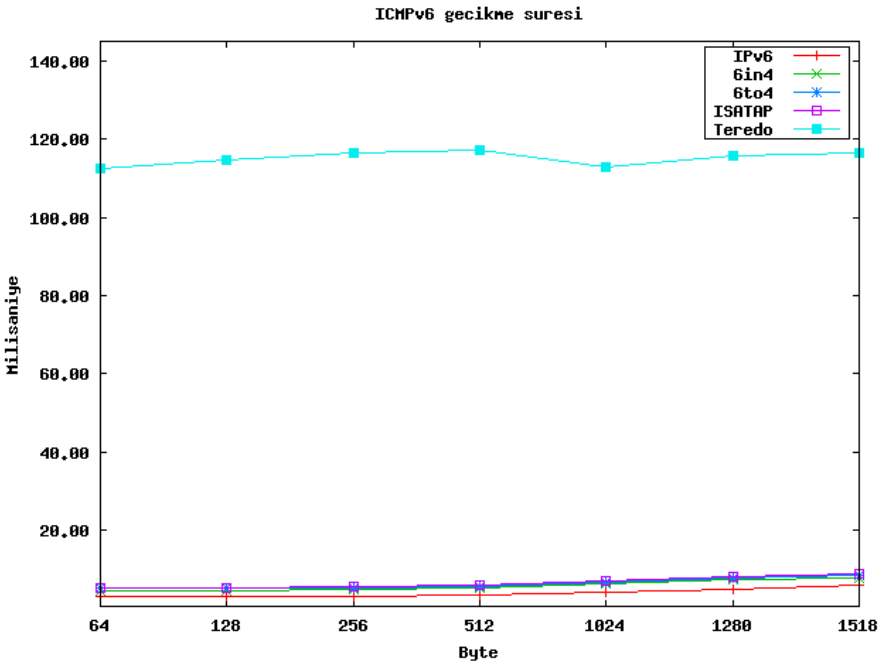
Test sonuçlarını gösteren grafiklerde, yalın IPv6 “IPv6”, Configured tunnel “6in4”, 6to4 yöntemi “6to4”, ISATAP yöntemi “ISATAP” ve Teredo yöntemi “Teredo” olarak adlandırılmıştır.

5.2.1. Normal durumda test sonuçları

Normal durumda, gecikme süresi, througput ve işlemci kullanımı testleri gerçekleştirilmiştir. Her bir metrik için ayrıntılı sonuçlar aşağıda tartışılmıştır.

5.2.1.1. Gecikme süresi

ICMPv6 ile yapılan gecikme süresi testi, sadece normal durum altında yapılmıştır. Boyutları 64 ile 8192 byte arasında değişen paketler kullanılarak yapılan test sonuçları Şekil 40’ta gösterilmiştir.



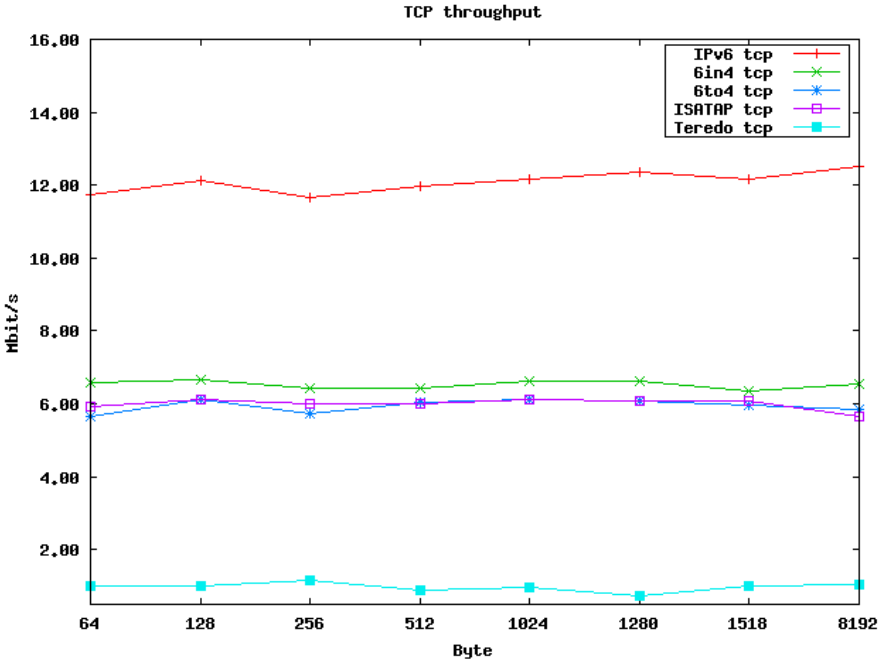
Şekil 40. ICMPv6 gecikme süresi

Teredo yönteminde gecikme süresinin diğer yöntemlerden çok daha fazla olduğu görülmektedir. Genel olarak 2ms ile 8ms arasında olan gecikme süresi değerleri Teredo yönteminde 116ms değerine kadar artmıştır. Teredo yönteminde yaşanan bu kaybın nedeni, kullanıcı seviyesinde çalışan Miredo yazılımı olabilir.

5.2.1.2. Througput

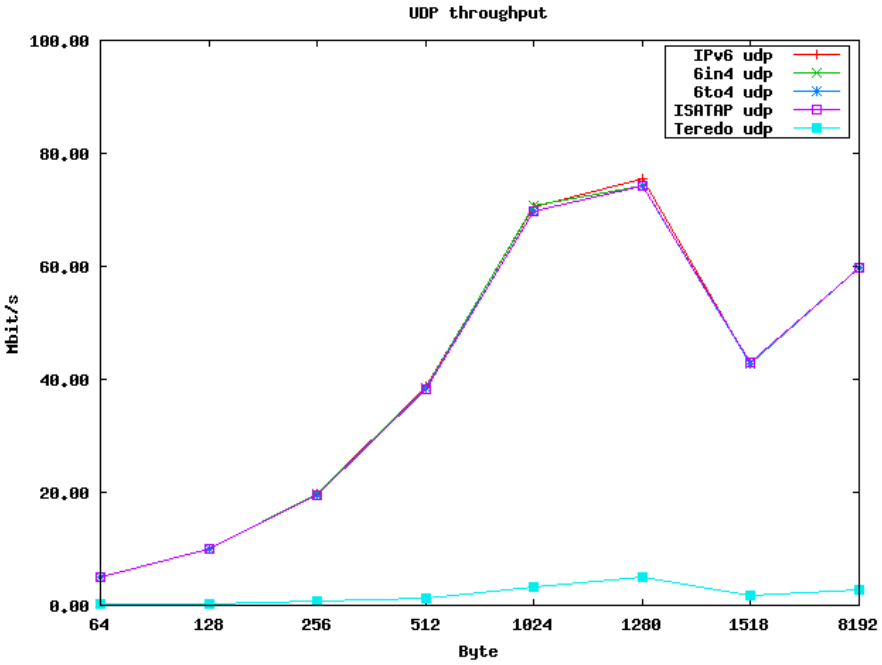
Througput testleri, TCP ve UDP protokolleri kullanılarak farklı paket boyutları için tekrarlanmıştır.

TCP protokolü kullanıldığında, en yüksek performans yalın IPv6 ile sağlanmıştır (12.65Mbit/s). En düşük TCP througput performansı ise Teredo yöntemine aittir. ISATAP, Configured tunnel ve 6to4 yöntemleri benzer performans değerleri göstermiştir. Fakat bu tünelleme yöntemleri, yalın IPv6 performansının ancak yarı değerini sağlayabilmiştir. TRT yönteminde TCP throughput testi FTP uygulaması ile yapılmıştır. Bunun nedeni, TRT sunucusu üzerinde kullanılan FAITH yazılımının Netperf uygulamasını desteklememesidir. FTP ile yapılan dosya transferinde, TRT TCP througput değeri, ortalama 12.64 Mbit/s olarak tespit edilmiştir. Şekil 41'de TRT yöntemi hariç, TCP performans değerleri gösterilmiştir.



Şekil 41. TCP througput

UDP througput performansı, Şekil 42'de görüldüğü gibi Teredo yöntemi hariç benzer değerlerdedir.



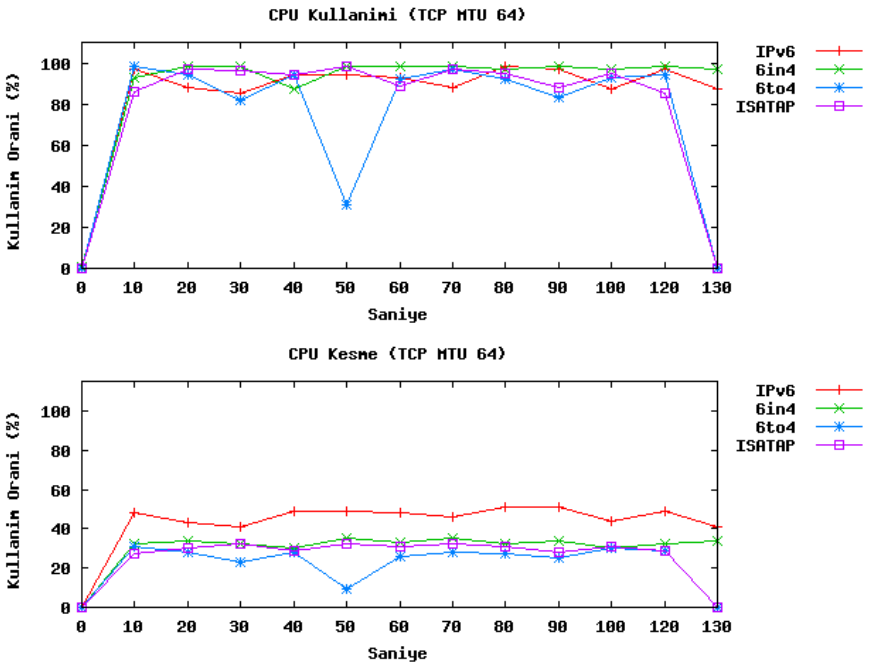
Şekil 42. UDP throughput

5.2.1.3. İşlemci kullanımı

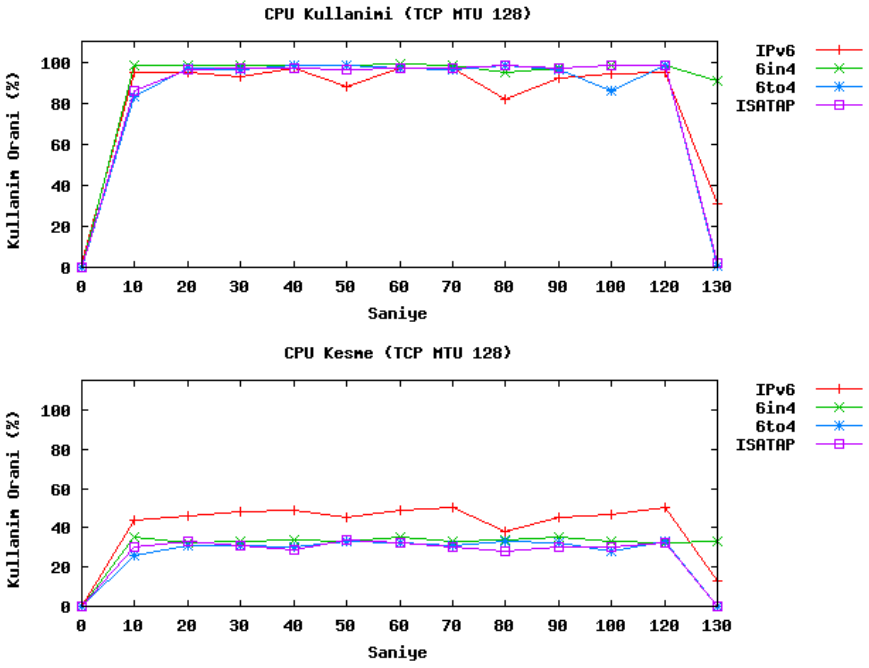
Performans testleri gerçekleştirilirken, yönlendiriciler üzerinden alınan işlemci kullanımı bilgileri aşağıda verilmiştir. Görselleştirilen bu veriler arasında TRT ve Teredo yöntemlerine ait bilgiler bulunmamaktadır. Çünkü bu yöntemlerde, yönlendiriciler üzerindeki işlemci kullanımından çok TRT ve Teredo sunucu / nakil üzerindeki yük önemlidir.

TRT ve Teredo sunucusu üzerinde oluşan işlemci yükü, önemsenmeyecek derecede az olarak tespit edilmiştir. Bu yöntemlerde çevirme ve tünelleme işlemleri FreeBSD işletim sistemi ve PC donanımı üzerinde yapıldığı için, farklı mimarideki Cisco yönlendiriciler ile gerçekleştirilen testlerle birlikte görselleştirilmemiştir.

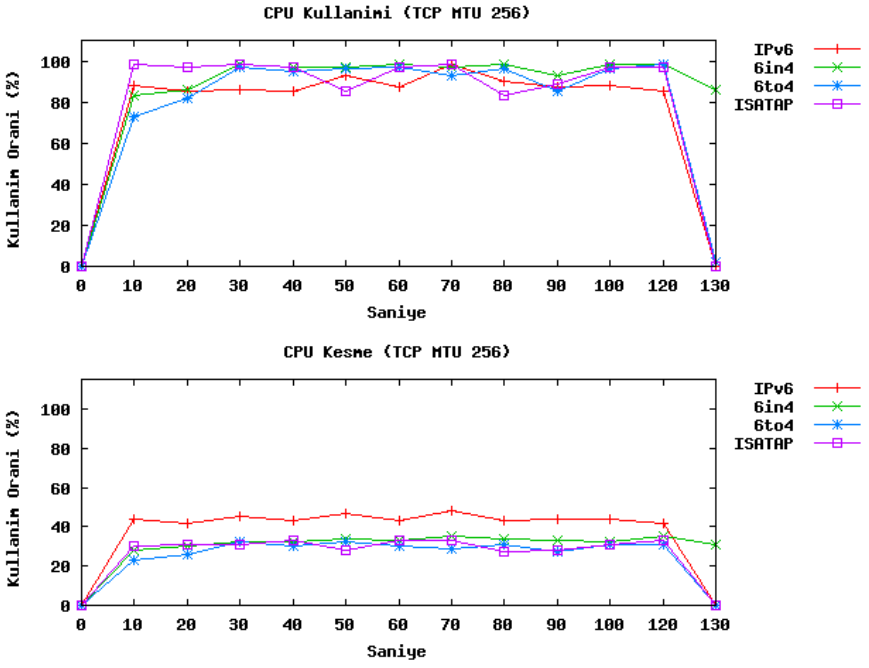
- *Y1 yönlendiricisi işlemci kullanımı:*



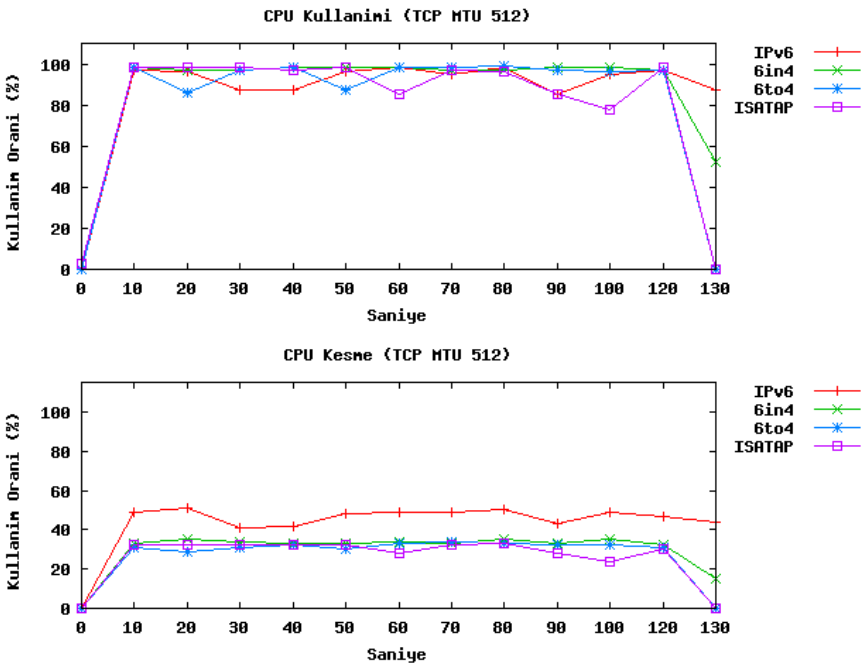
Şekil 43. TCP MTU 64 işlemci kullanımı (Y1)



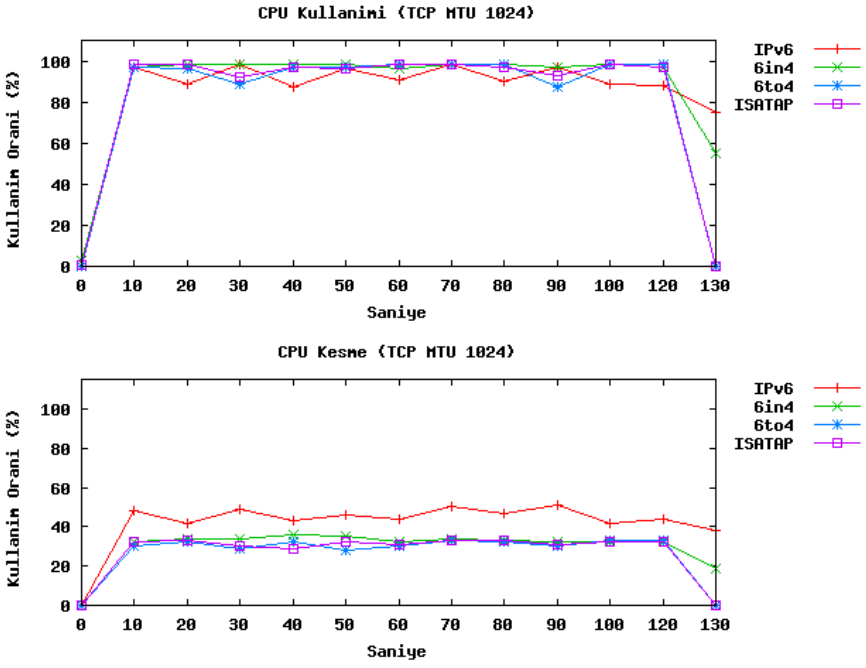
Şekil 44. TCP MTU 128 işlemci kullanımı (Y1)



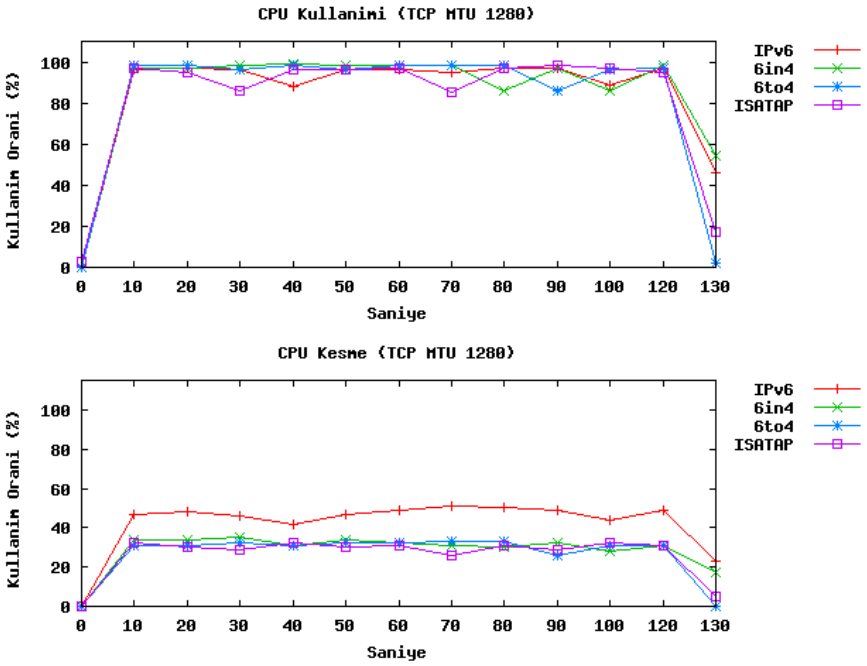
Şekil 45. TCP MTU 256 işlemci kullanımı (Y1)



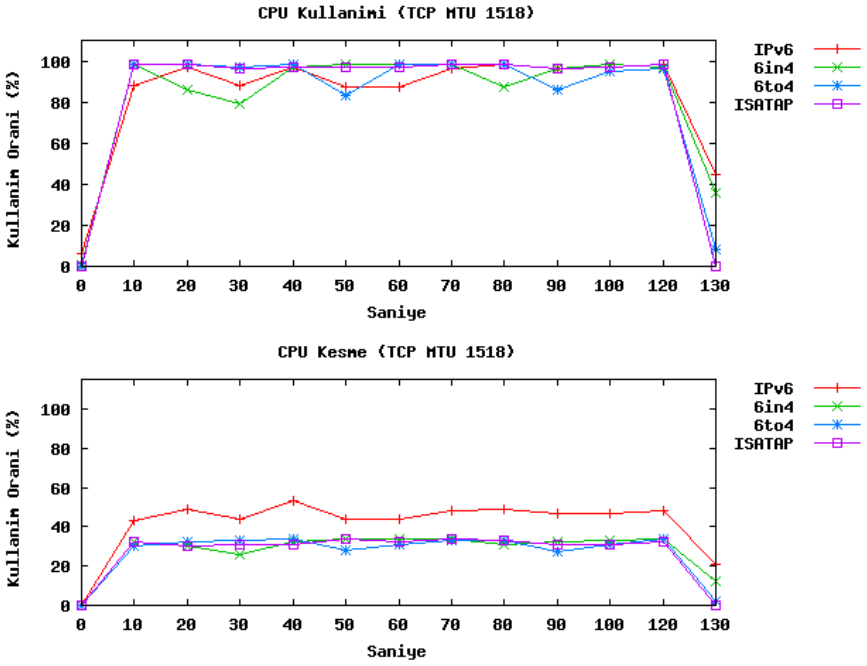
Şekil 46. TCP MTU 512 işlemci kullanımı (Y1)



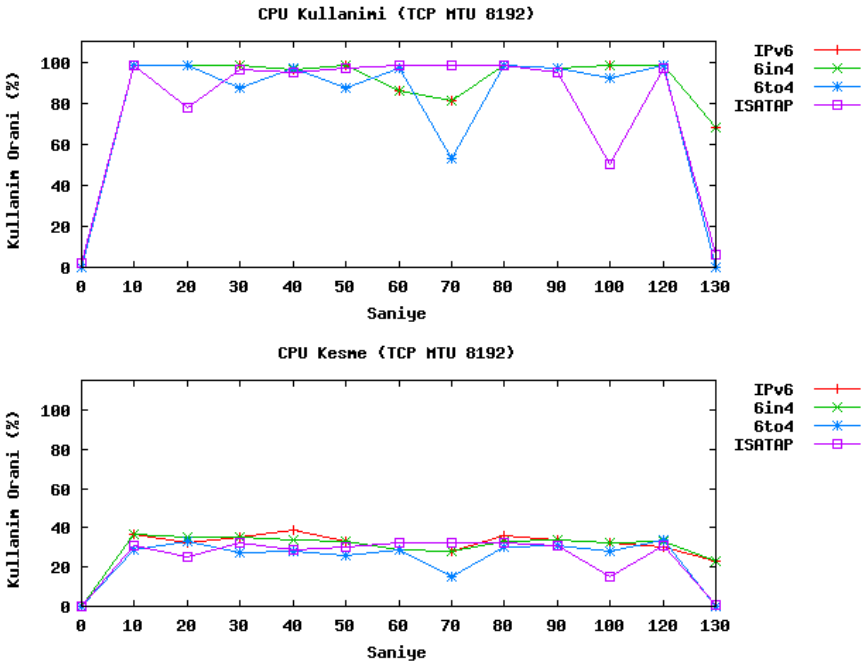
Şekil 47. TCP MTU 1024 işlemci kullanımı (Y1)



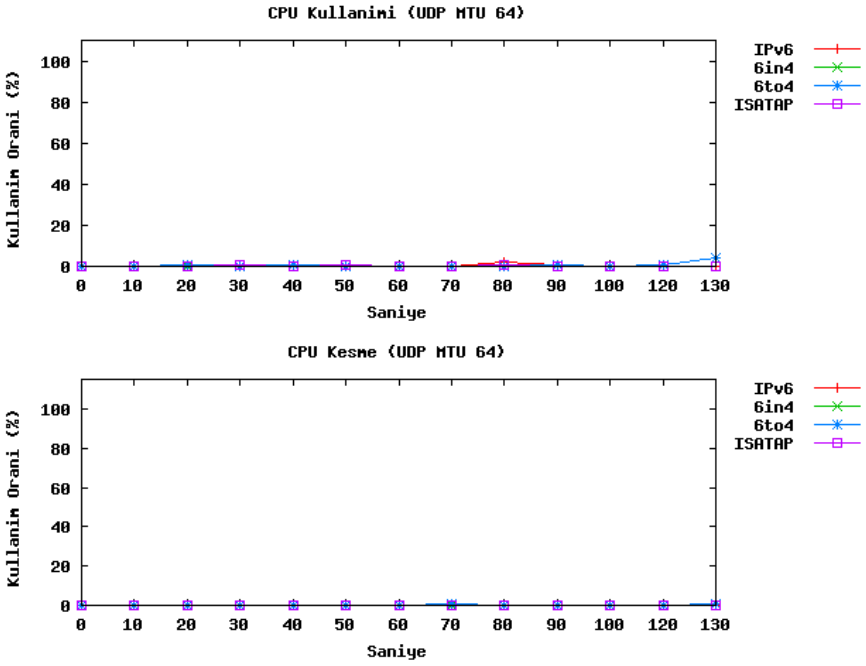
Şekil 48. TCP MTU 1280 işlemci kullanımı (Y1)



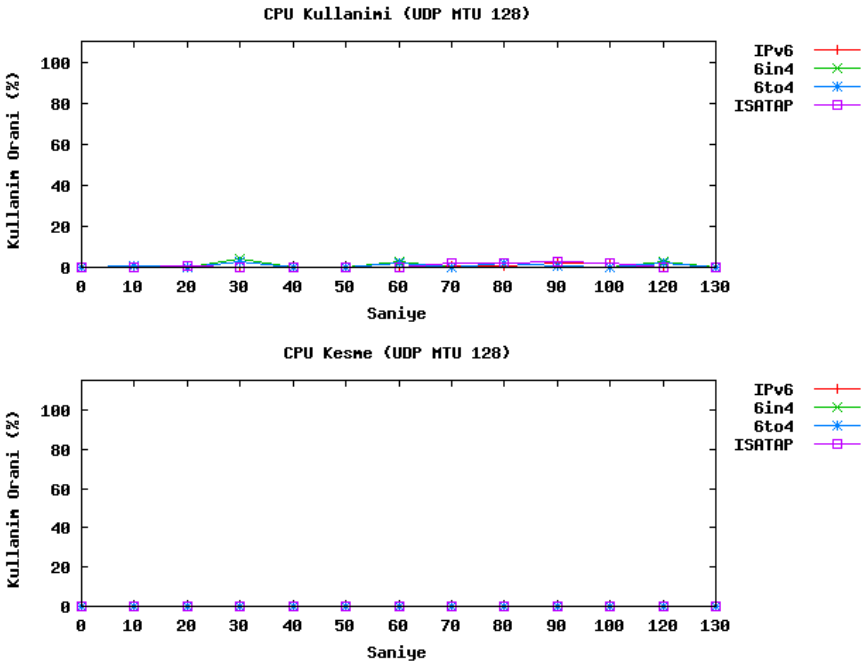
Şekil 49. TCP MTU 1518 işlemci kullanımı (Y1)



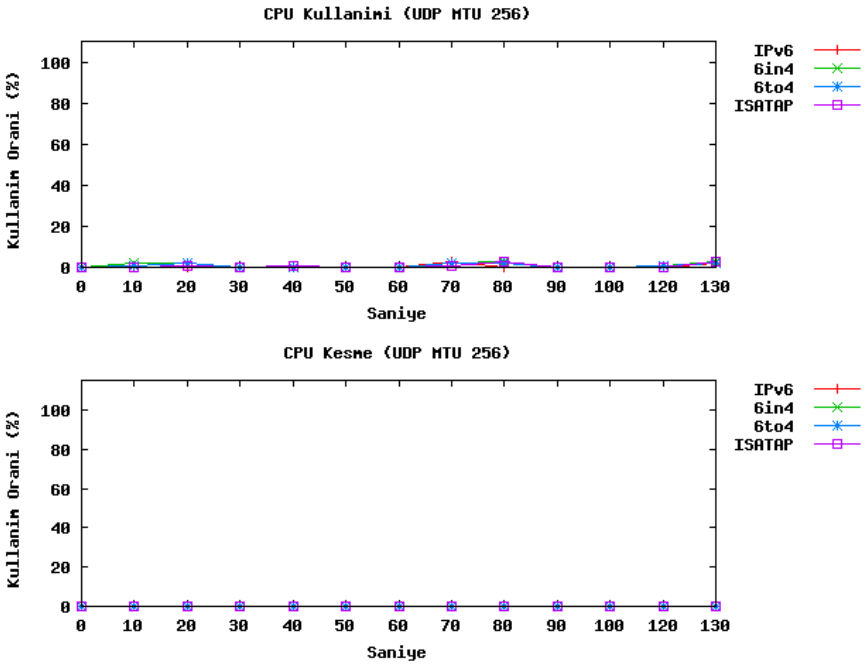
Şekil 50. TCP MTU 8192 işlemci kullanımı (Y1)



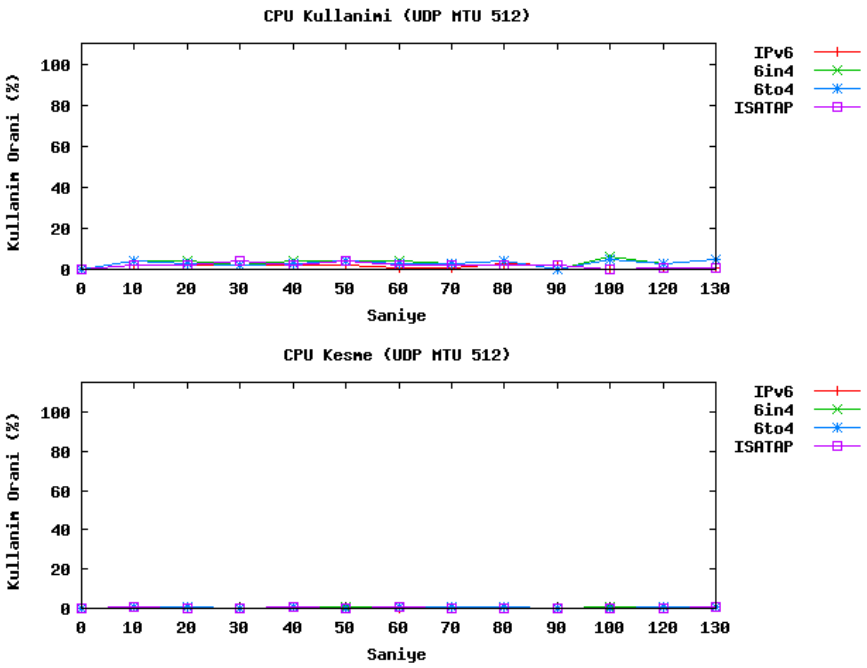
Şekil 51. UDP MTU 64 işlemci kullanımı (Y1)



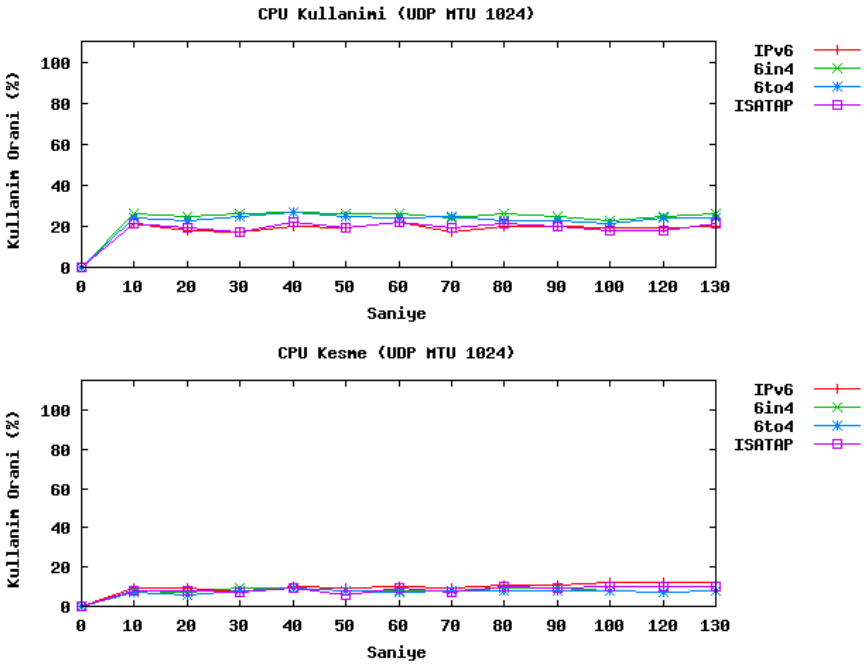
Şekil 52. UDP MTU 128 işlemci kullanımı (Y1)



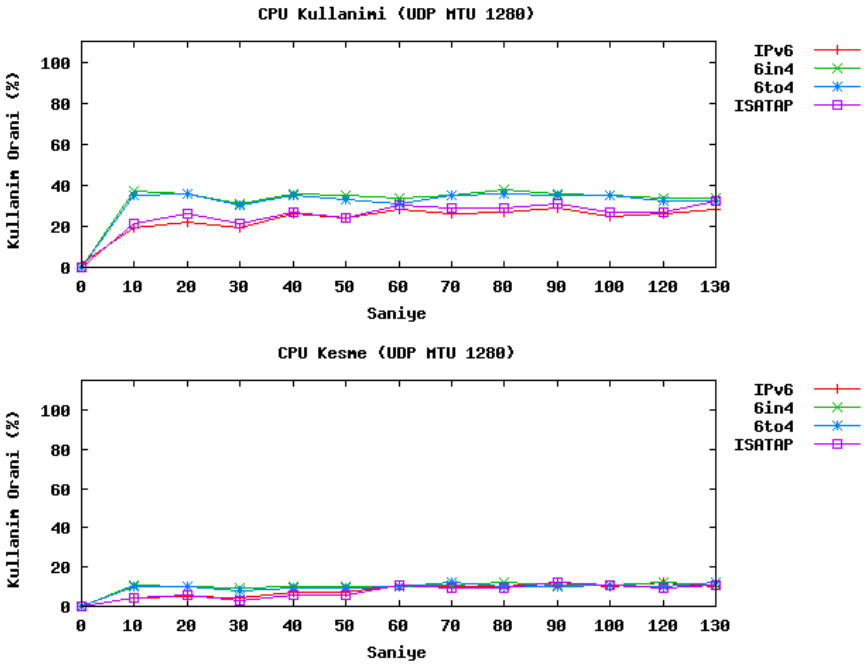
Şekil 53. UDP MTU 256 işlemci kullanımı (Y1)



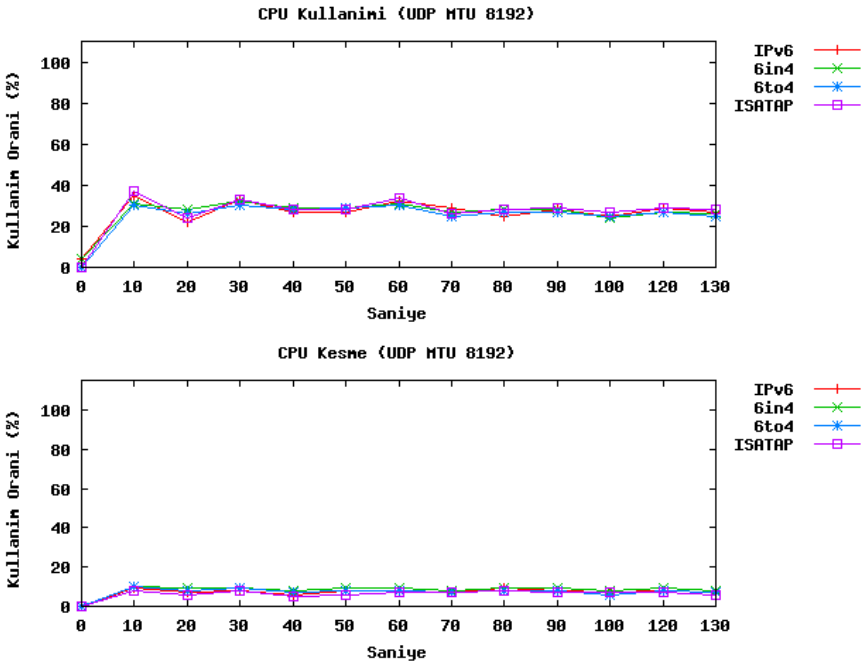
Şekil 54. UDP MTU 512 işlemci kullanımı (Y1)



Şekil 55. UDP MTU 1024 işlemci kullanımı (Y1)

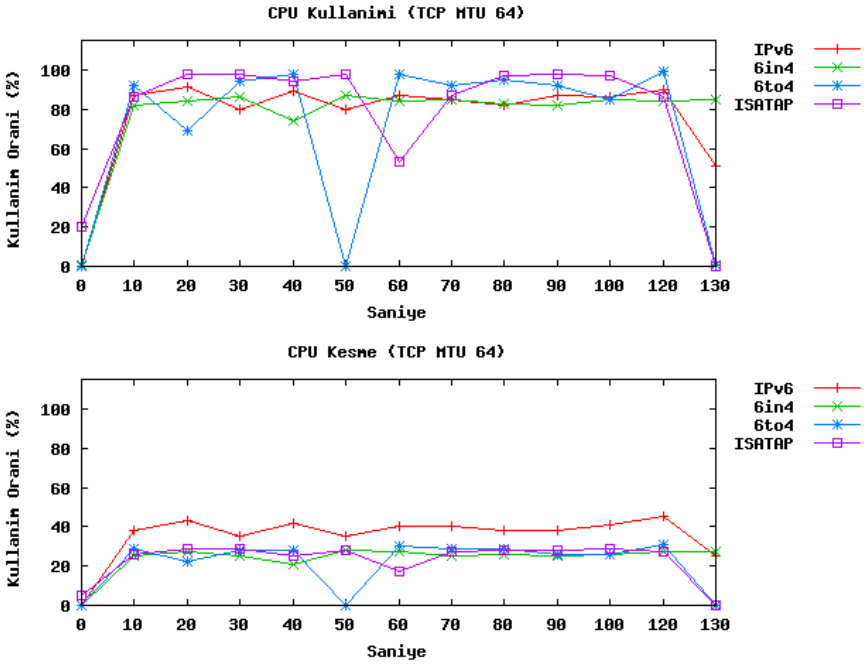


Şekil 56. UDP MTU 1280 işlemci kullanımı (Y1)

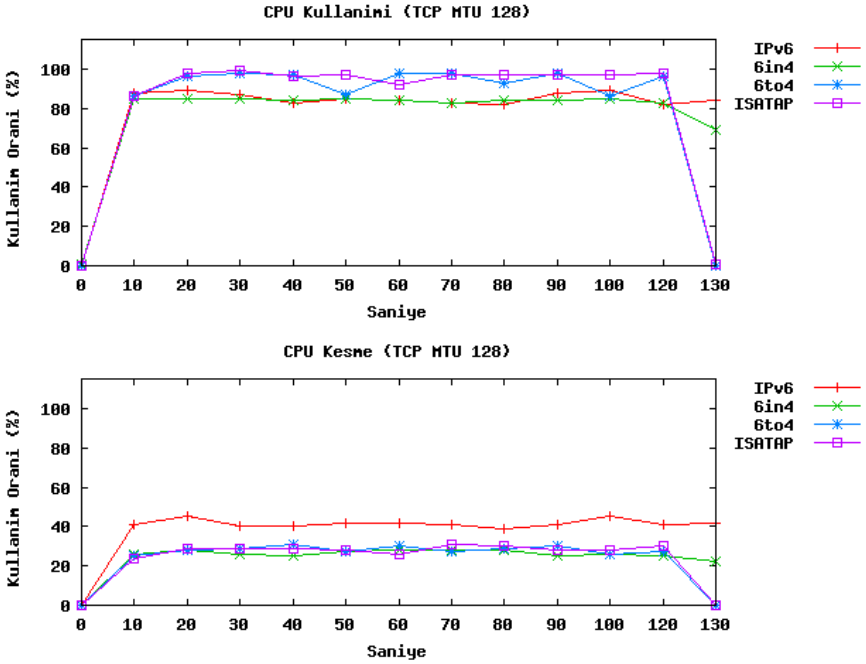


Şekil 57. UDP MTU 8192 işlemci kullanımı (Y1)

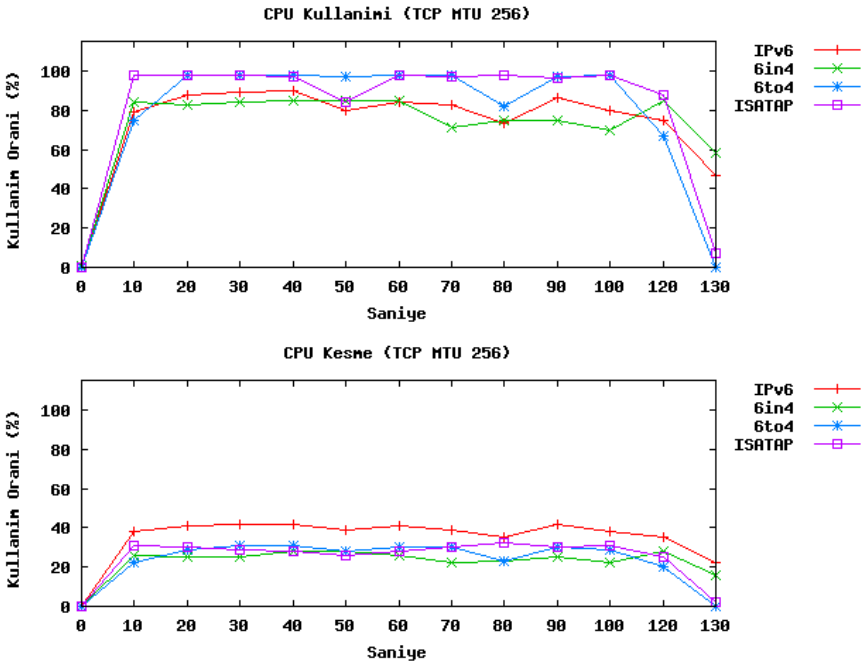
- *Y2 yönlendiricisi işlemci kullanımı:*



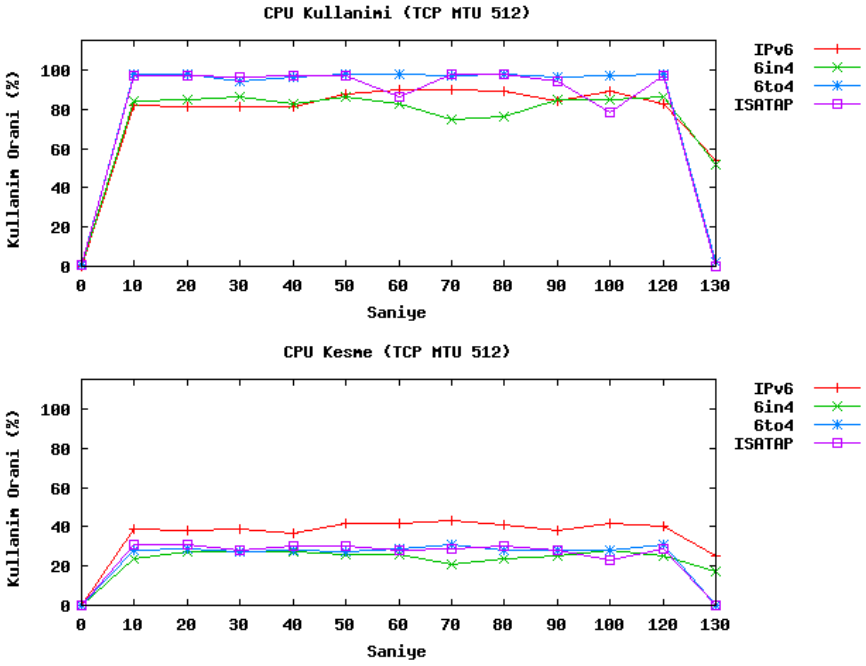
Şekil 58. TCP MTU 64 işlemci kullanımı (Y2)



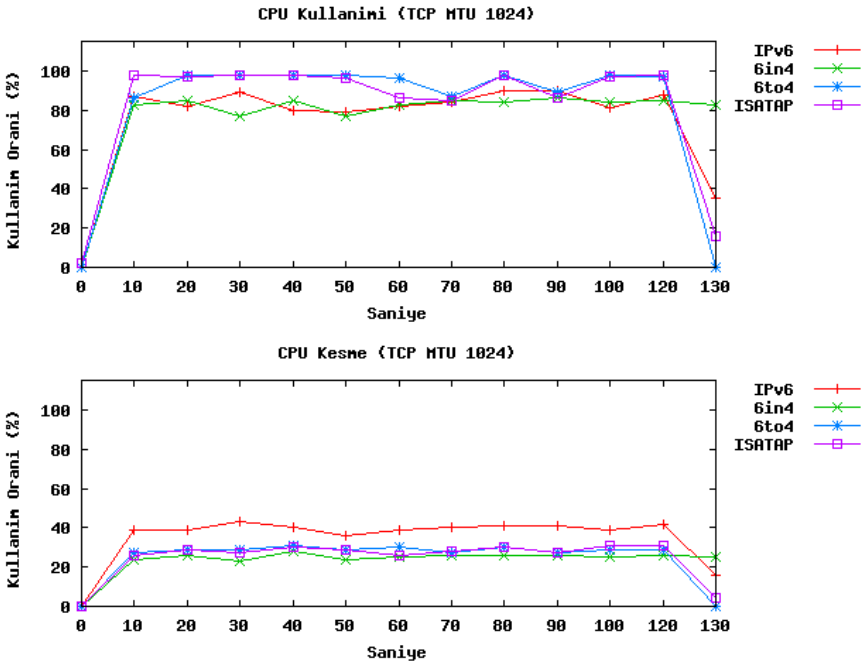
Şekil 59. TCP MTU 128 işlemci kullanımı (Y2)



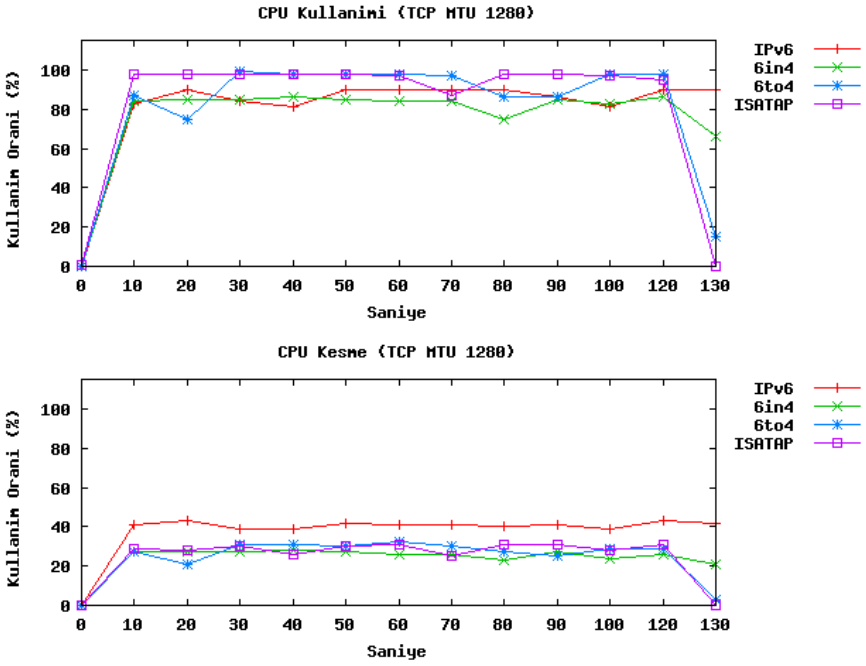
Şekil 60. TCP MTU 256 işlemci kullanımı (Y2)



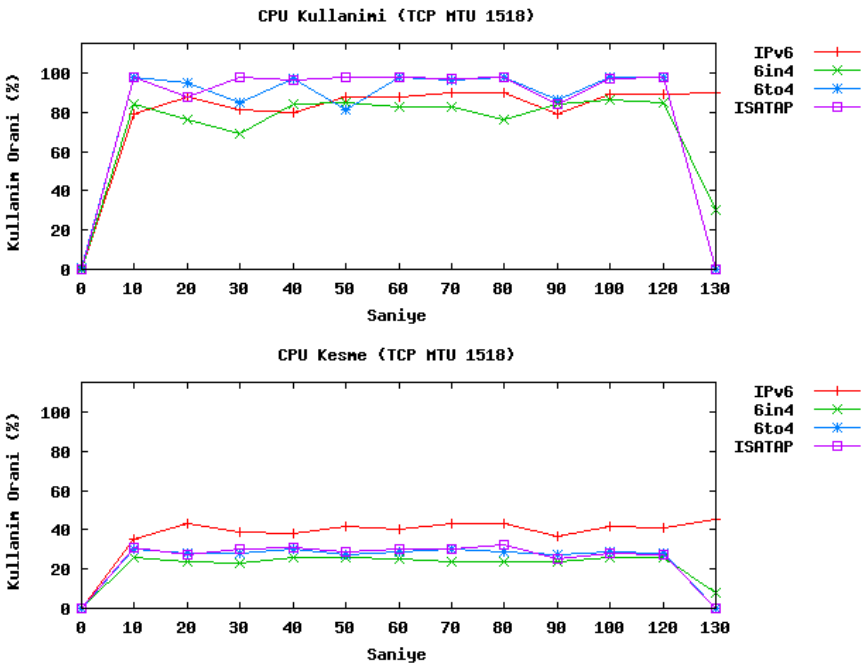
Şekil 61. TCP MTU 512 işlemci kullanımı (Y2)



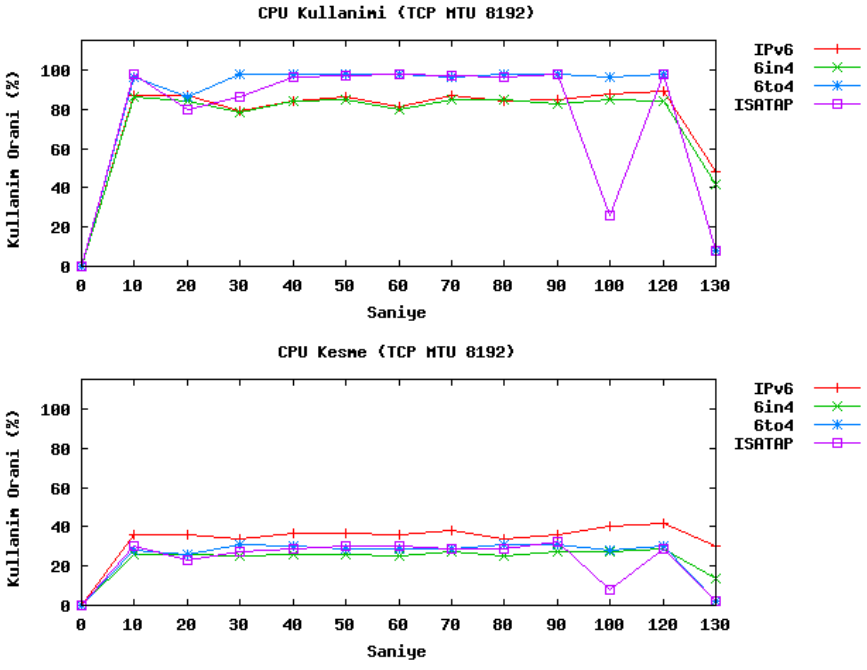
Şekil 62. TCP MTU 1024 işlemci kullanımı (Y2)



Şekil 63. TCP MTU 1280 işlemci kullanımı (Y2)



Şekil 64. TCP MTU 1518 işlemci kullanımı (Y2)



Şekil 65. TCP MTU 8192 işlemci kullanımı (Y2)

UDP performans testleri yapılırken, kontrol bilgisayarı Y2 yönlendiriciden işlemci kullanım istatistiklerini elde edememiştir. Çok sayıda UDP trafiğinden dolayı, kontrol bilgisayarı gerekli istatistikleri almak için TCP bağlantısını

gerçekleştirememiştir. Seri port (COM1) üzerinden bağlanan konsol yardımıyla Y2 yönlendiricisinde işlemci kullanımı tespit edilmeye çalışılmış ve %98-99 gibi bir işlemci yükü tespit edilmiştir. Kontrol bilgisayarı tarafından gerekli veriler periyodik olarak toplanamadığı için UDP grafikleri çizilememiştir.

5.2.2. Saldırı altında test sonuçları

IPv6 geçiş yöntemlerinin stres altında gerçekleştirilen performans testlerine ilişkin sonuçlar aşağıda verilmiştir. Bu test sonuçları, IPv6 geçiş yöntemlerinin servis dışı bırakma saldırılarından nasıl etkileneceği konusunda bir öngörü oluşturmaktadır.

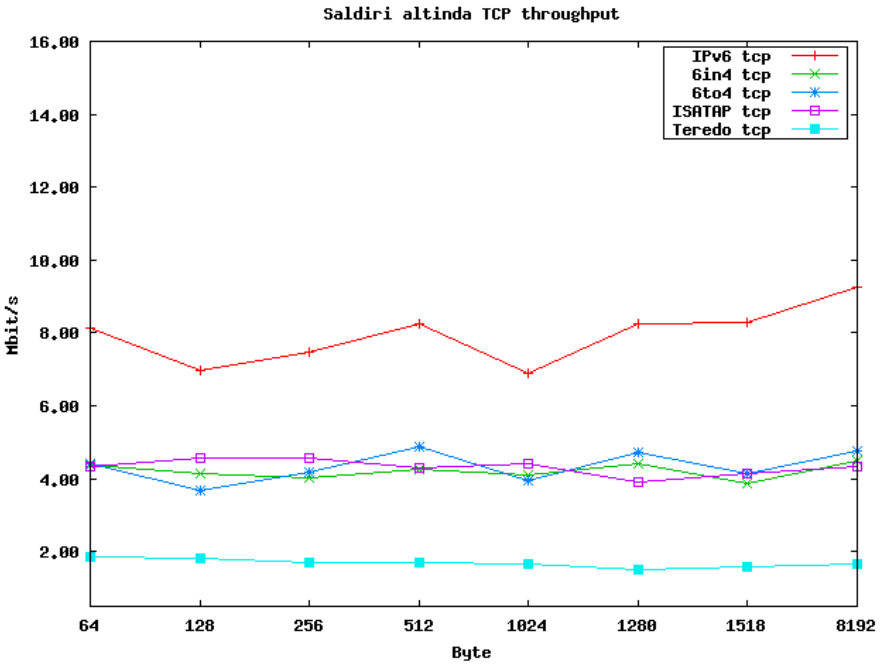
Cisco yönlendiricilere 100Mbit bağlantı ile saldırıldığında, yönlendiricilerin kısa sürede servis dışı kaldığı gözlemlenmiştir. Bu durum, donanımlarının eski ve yetersiz olması ile açıklanabilir. Performans testlerinin gerçekleştirilebilmesi için saldırı bilgisayarının bağlı olduğu switch portu 10Mbit çalışacak şekilde konfigüre edilmiştir.

Saldırı aracı olarak Hyenae [65] paket üreticisi kullanılmıştır. Yalın IPv6, Configured tunnel, 6to4 ve ISATAP yöntemlerinde D2 istemcisi ile aynı ağda bulunan Y2 yönlendiricisine “icmp-echo” saldırısı yapılmıştır. Teredo ve TRT yöntemlerinde ise, Teredo ve TRT sunucuya “icmp-echo” saldırısı yapılmıştır. Teredo ve TRT sunucuları üzerinde VMSTAT [66] uygulaması ile işlemci üzerinde oluşan yükler tespit edilmiştir. Bu iki yöntemin diğer geçiş yöntemleri ile karşılaştırması mimari farklılığından dolayı bir anlam ifade etmeyecektir. Bu nedenle grafikleri çizilmemiştir.

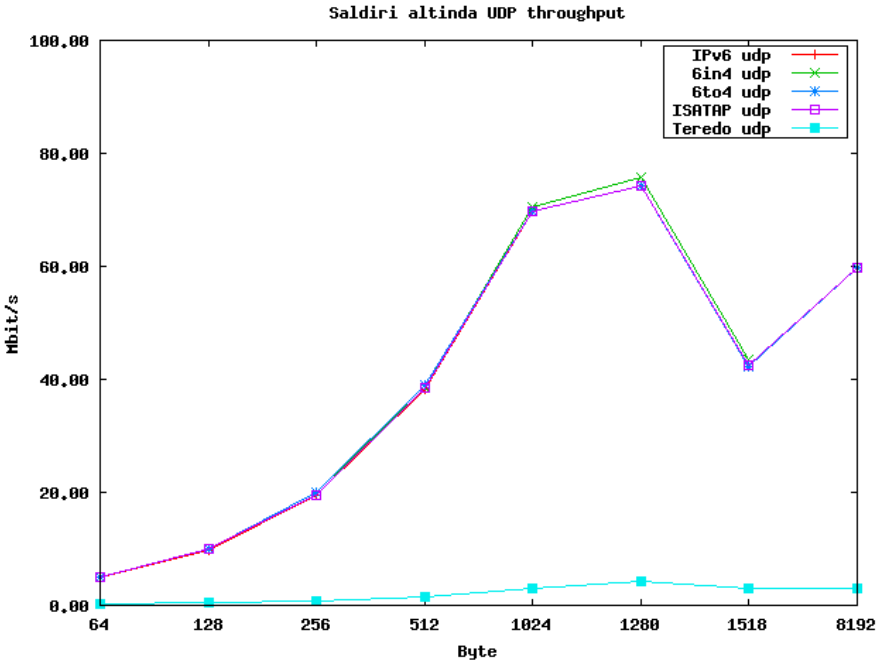
5.2.2.1. Througput

Saldırı altında TCP througput performansı yaklaşık %50 oranında düşmüştür. Bu durum Şekil 66’da açıkça görülmektedir. Normal durumda olduğu gibi saldırı altında da en yüksek performans yalın IPv6, en düşük performans ise Teredo yöntemine aittir. ISATAP, Configured tunnel ve 6to4 yöntemleri ise benzer performans değerleri göstermiştir.

Saldırı altında UDP througput testlerine ait sonuçlar Şekil 67’de gösterilmiştir. Bu sonuçlara göre normal durumda ve saldırı altında UDP througput performansının değişmediği görülmektedir.



Şekil 66. Saldırı altında TCP throughput

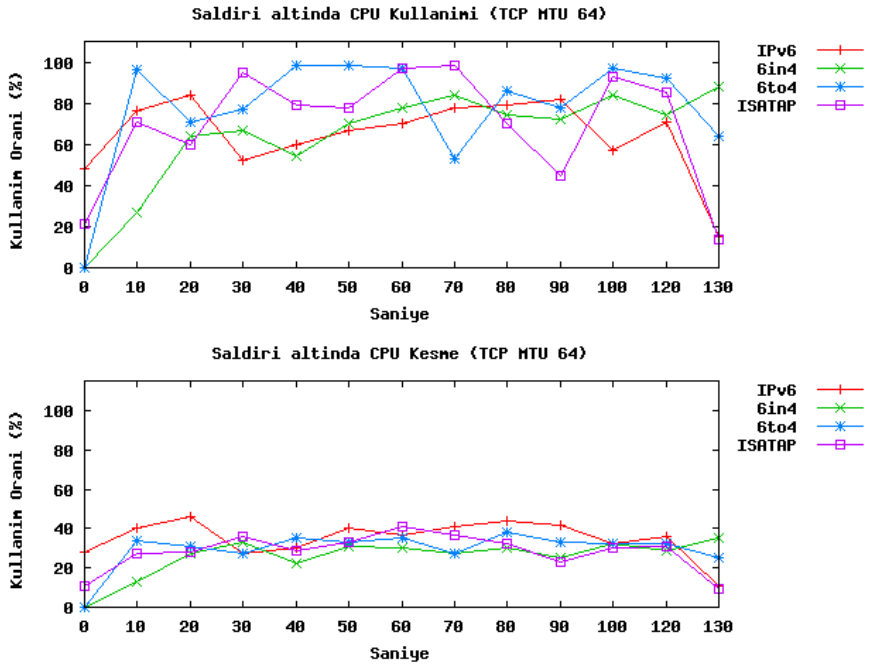


Şekil 67. Saldırı altında UDP throughput

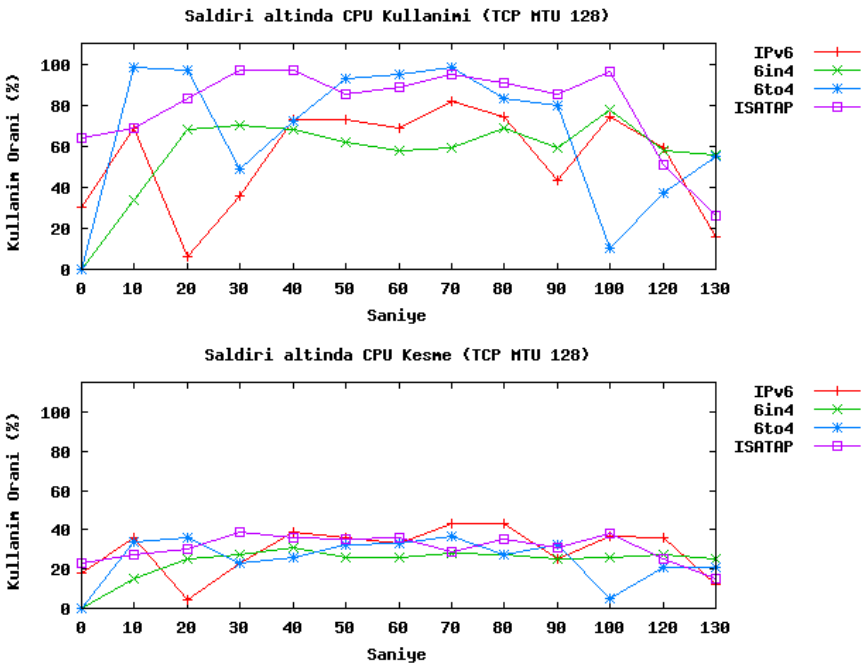
5.2.2.2. İşlemci kullanımı

Teredo ve TRT geçiş yöntemleri saldırı altında oldukça stabil kalmıştır. Bu durumu, sunucu üzerindeki gigabit arayüzün, donanımın ve FreeBSD 7 ağ yığınının performansı ile açıklamak mümkündür. VMSTAT çıktılarında FreeBSD 7 üzerindeki işlemci yükünün önemsenmeyecek derecede olduğu tespit edilmiştir.

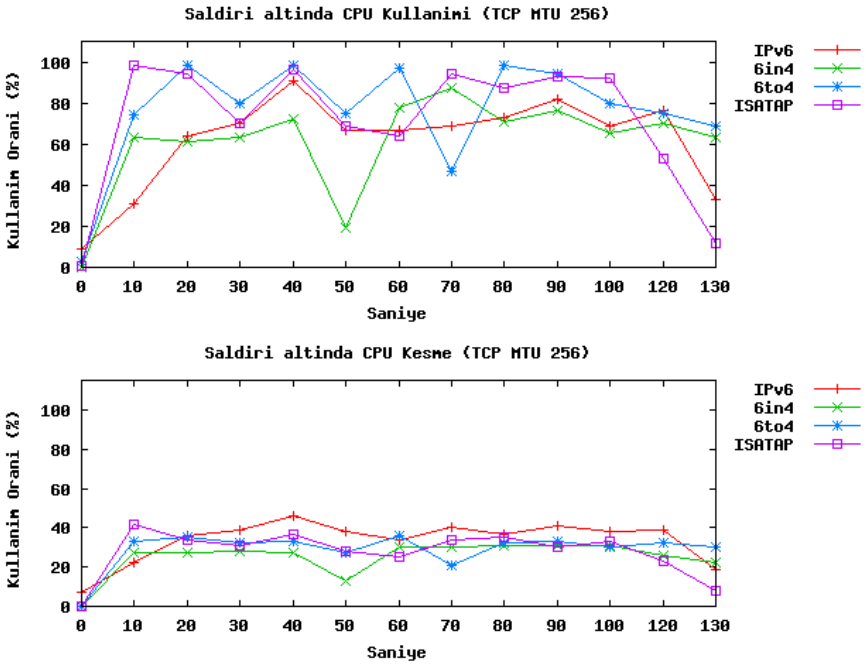
Saldırı senaryosunda, Y1 yönlendiricisi doğrudan etkilenmediği için bu yönlendiricinin işlemci kullanımı önemsenmemiştir. Aşağıda verilen grafikler, TCP througput testleri yapılırken R2 yönlendiricisi üzerinde oluşan yükleri göstermektedir. UDP testleri sırasında yönlendirici yükü nerdeyse %100'e ulaştığı için grafikleri oluşturmada kullanılan veri kümesi toplanamamıştır.



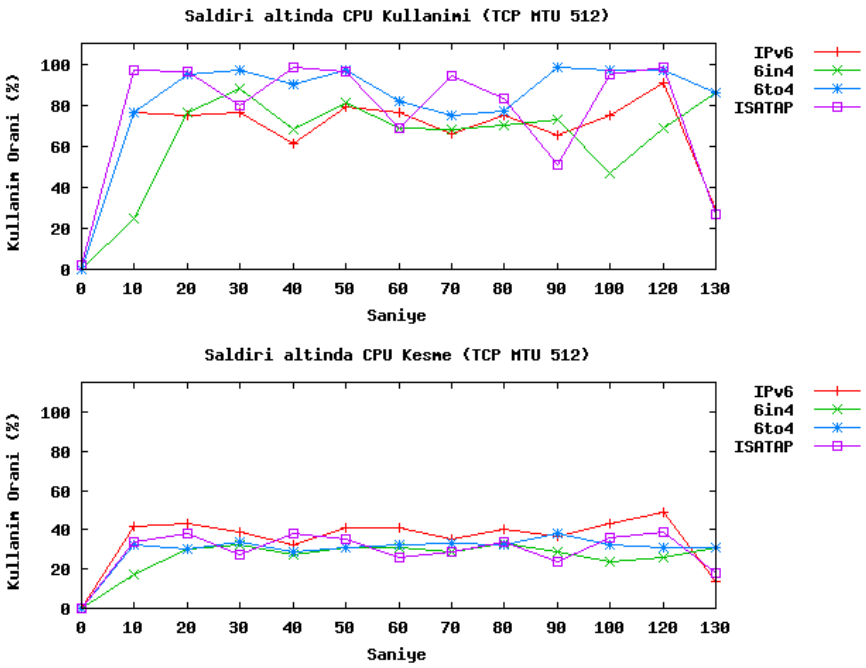
Şekil 68. Saldırı altında TCP MTU 64 işlemci kullanımı (Y2)



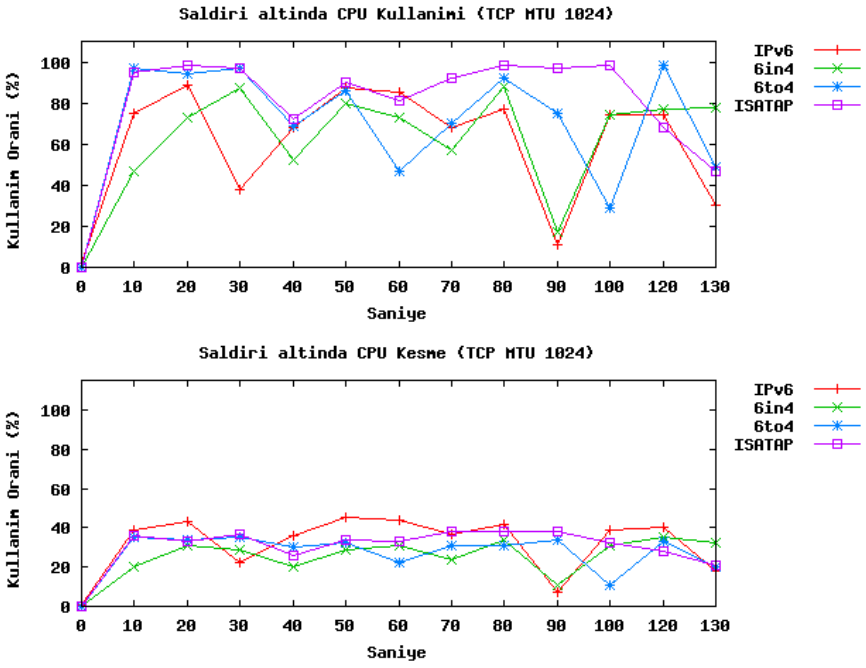
Şekil 69. Saldırı altında TCP MTU 128 işlemci kullanımı (Y2)



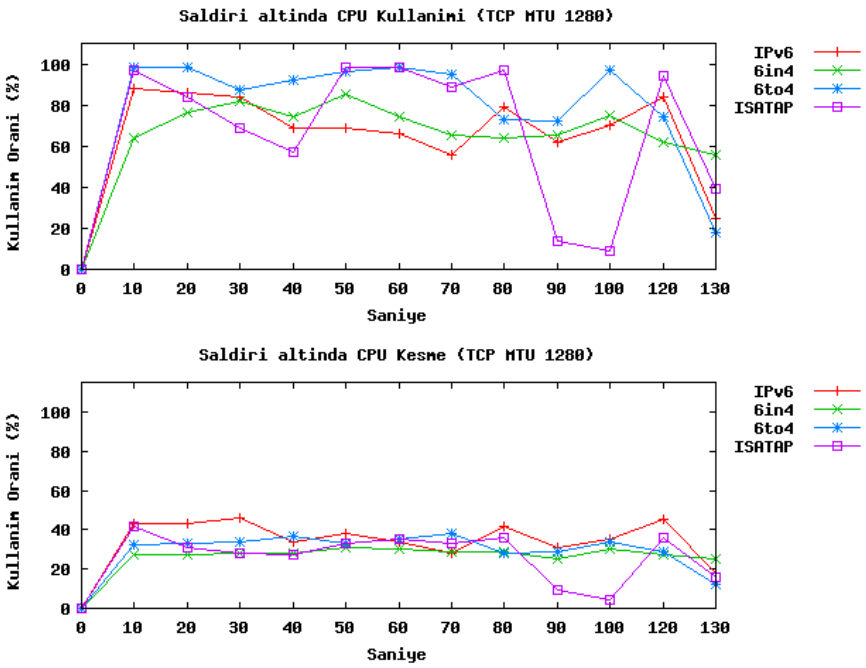
Şekil 70. Saldırı altında TCP MTU 256 işlemci kullanımı (Y2)



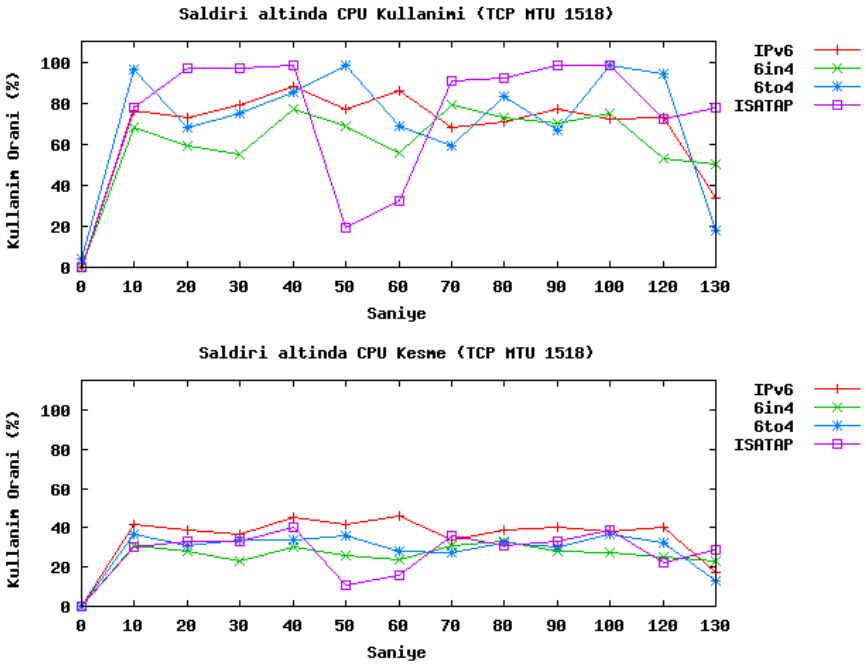
Şekil 71. Saldırı altında TCP MTU 512 işlemci kullanımı (Y2)



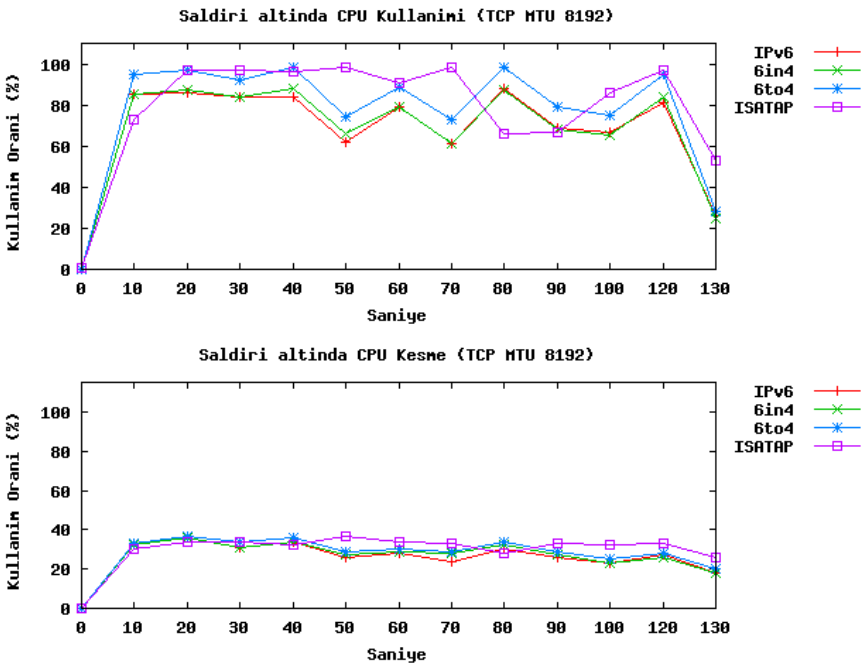
Şekil 72. Saldırı altında TCP MTU 1024 işlemci kullanımı (Y2)



Şekil 73. Saldırı altında TCP MTU 1280 işlemci kullanımı (Y2)



Şekil 74. Saldırı altında TCP MTU 1518 işlemci kullanımı (Y2)



Şekil 75. Saldırı altında TCP MTU 8192 işlemci kullanımı (Y2)

5.3. Bölüm Değerlendirmesi

Yapılan testlerden elde edilen veriler ışığında, yalın IPv6 performansının IPv6 geçiş yöntemlerinden daha performanslı olduğu görülmektedir. Özellikle TCP kullanıldığında %50'ye varan farklar gözlemlenmiştir. En düşük performansa sahip geçiş yöntemi olan Terdeo, yalın IPv6 TCP performansına göre %110 daha azdır. Diğer yandan, TRT çevirici yönteminde ftp kullanılarak elde edilen TCP througput değerinin, yalın IPv6 througput değerine yakın olduğu tespit edilmiştir. Bu durum, TRT yönteminde kullanılan donanımın Cisco 2600 serisi yönlendiricilerden daha iyi olması ve FreeBSD ağ yığının performansı ile açıklanabilir.

UDP protokolü bağlantı kurmadan (connectionless) paket iletişimi gerçekleştirdiği için TCP protokolüne göre daha performanslı sonuçlar elde edilmiştir. Genel olarak, IPv6 geçiş yöntemlerinin UDP througput performansı yalın IPv6 performansı ile benzer bulunmuştur. Bu bulgunun istisnası, TCP protokolünde de olduğu gibi Teredo yöntemidir.

Yapılan performans testleri sonucunda, işlemci kullanım değerlerinin benzer olduğu tespit edilmiştir. Yalın IPv6 kullanıldığında, işlemci kesme (interrupt) değerinin %10 - %15 daha fazla olduğu görülmüştür. Bu durum, testlerde TCP protokolü kullanıldığında gözlemlenmiştir. UDP testleri gerçekleştirilirken, Y2 yönlendiricisi üzerindeki işlemci yükü %100'e yakın değerlere ulaşmıştır. Y1 yönlendiricisinde ise paket boyutlarındaki artışa paralel olarak işlemci yükü de artmıştır. TRT ve Teredo yöntemlerinde, sunucu üzerindeki işlemci yükü önemsiz derecede az bulunmuştur.

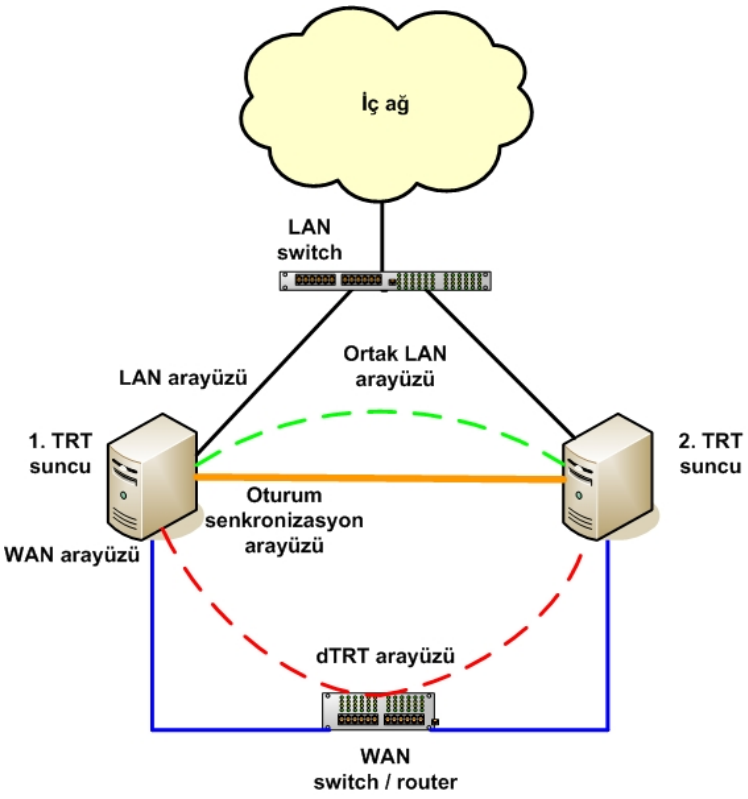
Daha önce de belirtildiği gibi, mimari ve donanım farkı bunda etkili olmuştur ve diğer yöntemlerle kıyaslanması objektif bir değerlendirme sağlamayacaktır.

Saldırı altında yapılan testlerde ise, stres altında TCP performansının %40'a varan değerlerde azaldığı gözlemlenmiştir. Beklendiği gibi işlemci kullanımı da artmıştır. UDP testlerinde ise önemli bir fark tespit edilememiştir.

5.4. TRT Çevirici Yöntemi İçin Yedekli Mimari

Birçok kaynakta TRT yönteminin tek nokta hatalarına açık olduğu işaret edilmiştir [67]. TRT sunucusunun herhangi bir sebeple devre dışı kalması durumunda, IPv6 düğümlerinin IPv4 düğümleri ile iletişimi kesilebilir. Bu soruna çözüm olarak yedekli TRT mimarisi önerilmiş ve dTRT (dual TRT) olarak adlandırılmıştır.

TRT yönteminde IPv6 – IPv4 çevirimi iletim katmanında yapılmaktadır. Veri iletişimi sırasında ağ katmanında herhangi bir çevirme işleminin yapılmaması dTRT yönteminin çıkış noktasıdır. Temel olarak, TRT sunucusunun ağ katmanındaki oturumlar bir başka sunucuya aktarıldığında, aktif – pasif çalışan yedekli bir mimari kurulabilir.



Şekil 76. dTRT mimarisi

Önerilen dTRT mimarisine ait bileşenler Şekil 76'da gösterilmiştir. dTRT mekanizmasında her bir sunucu için beş farklı arayüz tanımlamak gereklidir;

- Mavi renk ile gösterilen bağlantılar dış ağ arayüzleridir (WAN).
- Turuncu ile gösterilen bağlantı oturum senkronizasyon arayüzleridir.
- Yeşil ile gösterilen bağlantı ortak LAN arayüzüdür.
- Siyah ile gösterilen bağlantılar yerel ağ arayüzleridir (LAN).
- Kırmızı ile gösterilen bağlantı dTRT ağ arayüzüdür.

WAN arayüzüne global IP adresleri atanır, LAN arayüzüne ağ yapısına bağlı olarak özel veya global IP adresleri atanır, ortak LAN arayüzüne LAN arayüzü ile aynı ağdan bir IP adresi atanır, oturum senkronizasyon arayüzlerine özel IP adresleri atanır ve son olarak dTRT arayüzüne TRT sunucusunun IP adresi atanır. Bahsedilen arayüzlerden dTRT arayüzü ve ortak LAN arayüzü sanal (pseudo) arayüzdür, diğerleri ise gerçektir. Sanal arayüzlere, her iki TRT sunucusunda da aynı IP adresleri verilmelidir.

dTRT mekanizmasının işleyişi aşağıda anlatılmıştır:

1. TRT sunucularına ait gerçek arayüzlere IP adresleri atanır.
2. Sunucuların oturum senkronizasyonu için kullanacakları arayüzler çapraz (cross) kablo veya bir switch ile bağlanır.
3. dTRT arayüzüne her iki sunucuda da aynı IP adresi atanır. Bu sanal arayüz üzerinde belirlenecek bir metrik yardımıyla aktif ve pasif olan taraf belirlenir. Aktif sunucu, dTRT arayüzünde tanımlı olan IP adresini çalıştığı sürece kendi üzerine alır.
4. Ortak LAN arayüzüne her iki sunucuda da aynı IP adresi atanır. Bu sanal arayüz üzerinde belirlenecek bir metrik yardımıyla aktif ve pasif olan taraf belirlenir. Aktif sunucu, ortak LAN arayüzünde tanımlı olan IP adresini çalıştığı sürece kendi üzerine alır.
5. Aktif TRT sunucusu, ağ katmanındaki oturumları, senkronizasyon arayüzünden pasif sunucuya iletir.
6. Herhangi bir nedenle aktif sunucu devre dışı kaldığında, pasif sunucu, dTRT arayüzünde ve ortak LAN arayüzünde tanımlı olan IP adreslerini kendi üzerine alır.
7. Aktif sunucu yeniden çalışır duruma geldiğinde, pasif sunucu üzerinde bulunan IP adreslerini kendi üzerine alır.

dTRT mekanizması, OpenBSD tarafından geliştirilen Packet Filter (pf), pfsync ve CARP yazılımları ile kolayca uygulanabilir.

6. SONUÇ

Bu çalışmada, IPv6 geçiş yöntemlerinin güvenlik ve performans analizi yapılmıştır. Çalışmada yapılan performans testleri gerçek ağ üzerinde ve yaygın olarak kullanılan ağ donanımları ile gerçekleştirilmiştir. Elde edilen sonuçlar ışığında, ağ büyüklüğü ve ağ topolojisine bağlı olarak optimum IPv6 geçiş yönteminin seçiminde bir öngörüde bulunmak mümkündür.

KAYNAKLAR

1. İnternet : Internet World Stats " Internet Usage Statistics" http://www.internetworldstats.com/stats.htm, (2009).

2. İnternet : IETF "Next Generation Transition" http://www.ietf.org/wg/concluded/ngtrans.html, (2009).

3. İnternet : 6bone "6bone" http://go6.net/ipv6-6bone, (2009).

4. İnternet : Tübitak ULAKBİM "Ulusal IPv6 Protokol Altyapısı Tasarımı ve Geçişi Projesi" http://www.ipv6.net.tr/, (2009).

5. Raicu, I., "An Empirical Analysis of Inetrnet Protocol Version 6", Yüksek lisans tezi, ***Wayne State Univrsity Computer Science***, Detroit (2002)

6. Chang, Y., Wang, R., Chao, H., Chen, J., "Performance Investigation of IPv4/IPv6 Transition Mechanisms", ***Journal of Internet Technology***, 5 (2): (2004).

7. AlJa'afreh, R., Mellor, J., Awan, I., "A Comparison between the Tunneling process and Mapping schemes for IPv4/IPv6 Transition", ***International Conference on Advanced Information Networking and Applications Workshop***, Bradford UK, 601-606 (2009).

8. Park, E., Choe, B., "An IPv4-to-IPv6 Dual Stack Transition Mechanism Supporting Transparent Connections between IPv6 Hosts and IPv4 Hosts in Integrated IPv6/IPv4 Network", ***Communications, 2004 IEEE International Conference***, 2**,** 1024 - 1027 (2004).

9. Hong, Y. G., Shin, M. K., Kim, H. J., "Application Translation for IPv6 at NAT-PT", ***Communications, 2003 APCC,*** 1**,** 21-24 (2003).

10. Srisuresh, P., Holdrege, M., "IP Network Address Translator (NAT) Terminology and Considerations", ***Internet Engineering Task Force***, (1999).

11. Huang, S., Wu, Q., Lin, Y., "Tunneling IPv6 through NAT with Teredo Mechanism", ***19th International Conference on Advanced Information Networking and Applications***, 2, Taiwan, 813-818 (2005).

12. Govil, J., Govil, J., Kaur, N., Kaur, H., "An Examination of IPv4 and IPv6 Networks : Constraints and Various Transition Mechanisms", ***SECON 2008***, California USA, 178-185 (2008).

13. İnternet : Netfilter "Netfilter Hooks for Tunnel Writers" http://www.netfilter.org/documentation/HOWTO/netfilter-hacking-HOWTO-6.html, (2009).

14. Raste, T. M., Kulkarni, D. B., "Design and implementation scheme for deploying IPv4 over IPv6 tunnel***", Journal of Network and Computer Applications 31***, 66-72 (2008).

15. Johnson, D., Perkins, C., Arkko, J., "Mobility Support in IPv6", ***Internet Engineering Task Force*** , (2004).

16. Lee, S., Park, J., Kahng, H., Chong, I., "Performance Analysis of an Efficient Network Transition Mechanism Supporting Mobile IPv6***", ICOIN 2006***, Sendai Japan, 339-348 (2006).

17. Babiarz, J., Chan, K., Baker, F., "Configuration Guidelines for DiffServ Service Classes", ***Internet Engineering Task Force*** , (2006).

18. Bouras, C., Primpas, D., Stamos, K., "IPv6 QoS Testing on Dual Stack Network", ***AAA-Idea'06***, Pisa Italy, (2006).

19. Dutta, A., Alberi, J., Horgan, B., McAuley, T., Chee, D., Lyles, B., "IPv6 Transition Techniques For Legacy Application", ***MILCOM 2006***, Washington USA, 1-9 (2006).

20. İnternet : 6NET "IPv6 Deployment Guide" http://www.6net.org/book/deployment-guide.pdf, (2009).

21. Zheng, Q., Liu, T., Guan, X., Qu, Y., Wang, N., "A New Worm Exploiting IPv4-IPv6 Dual-stack Networks", ***WORM'07***, Virginia USA, (2007).

22. Çalışkan, B., Bektaş, O., "IPv6 İkili Yığın Geçiş Yönteminde Uygulamaların Saldırı Altında Performans Analizi", ***ISC 2008***, Ankara Türkiye, 145-150 (2008).

23. Borman, D., Deering, S., Hinden, R., "IPv6 Jumbograms", ***Internet Engineering Task Force*** , (1999).

24. McCann, J., Deering, S., Mogul, J., "Path MTU Discovery for IP version 6", ***Internet Engineering Task Force*** , (1996).

25. Hinden, R., Deering, S., "IP Version 6 Addressing Architecture", ***Internet Engineering Task Force*** , (2006).

26. Hinden, R., Deering, S., Nordmark, E., "IPv6 Global Unicast Address Format", ***Internet Engineering Task Force*** , (2003).

27. Hinden, R., Haberman, B., "Unique Local IPv6 Unicast Addresses", ***Internet Engineering Task Force*** , (2005).

28. Huitema, C., Carpenter, B., "Deprecating Site Local Addresses", ***Internet Engineering Task Force*** , (2004).

29. Nordmark, E., Gilligan, R., "Basic Transition Mechanisms for IPv6 Hosts and Routers", ***Internet Engineering Task Force*** , (2005).

30. Nordmark, E., Gilligan, R., "Basic Transition Mechanisms for IPv6 Hosts and Routers", ***Internet Engineering Task Force*** , (2000).

31. Carpenter, B., Moore, K., "Connection of IPv6 Domains via IPv4 Clouds", ***Internet Engineering Task Force*** , (2001).

32. Hinden, R., O'Dell, M., Deering, S., "An IPv6 Aggregatable Global Unicast Address Format", ***Internet Engineering Task Force*** , (1998).

33. Rekhter, Y., Moskowitz, B., Karrenberg, D., Groot, G. J., Lear, E., "Address Allocation for Private Internets", ***Internet Engineering Task Force*** , (1996).

34. Hinden, R., Deering, S., "Internet Protocol Version 6 (IPv6) Addressing Architecture", ***Internet Engineering Task Force*** , (2003).

35. Narten, T., Nordmark, E., Simpson, W., Soliman, H., "Neighbor Discovery for IP Version 6 (IPv6)", ***Internet Engineering Task Force*** , (2007).

36. Durand, A., Fasano, P., Guardini, I., Lento, D., "IPv6 Tunnel Broker", ***Internet Engineering Task Force*** , (2001).

37. Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", ***Internet Engineering Task Force*** , (2006).

38. Bound, J., Toutain, L., Medina, O., Dupont, F., Afifi, H., Durand, A., "Dual Stack Transition Mechanism (DSTM)", ***Internet Engineering Task Force*** , (2002).

39. Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., Carney, M., "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", ***Internet Engineering Task Force*** , (2003).

40. İnernet : Cisco System 6PE "IPv6 over MPLS"
http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/iosip_an.pdf, (2009).

41. McPherson, D., Dykes, B., "VLAN Aggregation for Efficient IP Address Allocation", ***Internet Engineering Task Force*** , (2001).

42. Palet, J., Olvera, C., Fernandez, D., “Forwarding Protocol 41 in NAT Boxes”, ***Internet Engineering Task Force*** , (2003).

43. Nordmark, E., “Stateless IP/ICMP Translation Algorithm (SIIT)”, ***Internet Engineering Task Force*** , (2000).

44. Tsuchiya, K., Higuchi, H., Atarashi, Y., “Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)”, ***Internet Engineering Task Force*** , (2000).

45. Lee, S., Shin, M-K., Kim, Y-J., Nordmark, E., Durand, A., “Dual Stack Hosts Using "Bump-in-the-API" (BIA)”, ***Internet Engineering Task Force*** , (2002).

46. Hagino, J., Yamamoto, K., “An IPv6-to-IPv4 Transport Relay Translator”, ***Internet Engineering Task Force*** , (2001).

47. Leech, M., Ganis, M., Lee, Y., Kuris, R., Koblas, D., Jones, L., “SOCKS Protocol Version 5”, ***Internet Engineering Task Force*** , (1996).

48. Thomson, S., Narten, T., Jinmei, T., “IPv6 Stateless Address Autoconfiguration”, ***Internet Engineering Task Force*** , (2007).

49. Moore, N., “Optimistic Duplicate Address Detection (DAD) for IPv6”, ***Internet Engineering Task Force*** , (2006).

50. Hogg, S., Vyncke, E., “IPv6 Security ”, ***Cisco Press***, Indianapolis, 427-476, (2009).

51. Baker, F., Savola, P., “Ingress Filtering for Multihomed Networks”, ***Internet Engineering Task Force*** , (2004).

52. Kent, S., Atkinson, R., “Security Architecture for the Internet Protocol”, ***Internet Engineering Task Force*** , (1998).

53. Dommety, G., “Key and Sequence Number Extensions to GRE”, ***Internet Engineering Task Force*** , (2000).

54. Templin, F., Gleeson, T., Thaler, D., “Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) “,***Internet Engineering Task Force*** , (2008).

55. Chown, T., “Use of VLANs for IPv4-IPv6 Coexistence in Enterprise Networks”, ***Internet Engineering Task Force*** , (2006).

56. Carpenter, B., Jung, C., “Transmission of IPv6 over IPv4 Domains without Explicit Tunnels”, ***Internet Engineering Task Force*** , (1999).

57. Aoun, C., Davies, E., "Reasons to Move the Network Address Translator - Protocol Translator (NAT-PT) to Historic Status", ***Internet Engineering Task Force*** , (2007).

58. Kitamura, H., "A SOCKS-based IPv6/IPv4 Gateway Mechanism", ***Internet Engineering Task Force*** , (2001).

59. İnternet : Cisco Systems "Performance-Comparison Testing of IPv4 and IPv6 Throughput and Latency on Key Cisco Router Platforms", http://www.cisco.com/web/strategy/docs/gov/IPv6perf_wp1f.pdf, (2009).

60. İnternet : Netperf "Netperf" http://www.netperf.org/netperf/, (2009).

61. Popoviciu, C., Hamza, A., Velde, G., Dugatkin, D., "IPv6 Benchmarking Methodology for Network Interconnect Devices", ***Internet Engineering Task Force*** , (2008).

62. İnternet : Gnuplot "Gnuplot" http://www.gnuplot.info, (2009).

63. İnternet : Miredo "Miredo" http://www.remlab.net/miredo/, (2009).

64. İnternet : FAITH "faithd" http://www.freebsd.org/cgi/man.cgi?query=faithd&sektion=8, (2009).

65. İnternet : Hyenae "Hyenae" http://sourceforge.net/projects/hyenae/, (2009).

66. İnternet : FreeBSD VMSTAT "vmstat" http://www.freebsd.org/cgi/man.cgi?query=vmstat&apropos=0&sektion=0&manpath=FreeBSD+7.2-RELEASE&format=html, (2009).

67. İnternet : Cisco Systems "IPv6 Deployment Strategies" http://www.cisco.com/en/US/docs/ios/solutions_docs/ipv6/IPv6dswp.html#wp1075253, (2009).

EKLER

EK-1

Cisco yönlendiricilerden, işlemci kullanım oranlarını elde eden betik.

```
#!/usr/bin/perl
#
# Copyright (c) 2009 Beyhan CALISKAN beyhan@ulakbim.gov.tr.
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met:
# 1. Redistributions of source code must retain the above copyright
#    notice, this list of conditions and the following disclaimer.
# 2. Redistributions in binary form must reproduce the above copyright
#    notice, this list of conditions and the following disclaimer in the
#    documentation and/or other materials provided with the distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
# ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
# IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
# ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
# FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
# DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
# OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
# LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
# OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
# SUCH DAMAGE.

use warnings;
use Net::Telnet::Cisco;

my $session=Net::Telnet::Cisco ->new(Host=> '10.10.50.50', Input_log =>
"input.log",);
      $session->login('','secret_passwd') ;

         $session->enable("passwd") || die 'cant enable' ;
         $session->always_waitfor_prompt;

sub GetCpu {

         $run = "sh process cpu | in five seconds" ;
         $session->cmd("$run");
}

$i = 0;
while ($i <= 12) {

          &GetCpu()
          sleep 10;
```

```
        ++$i ;
}

$session->close ;
```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ÇALIŞKAN, Beyhan

Uyruğu : T.C.

Doğum tarihi ve yeri : 1981, Bulgaristan

e-mail : byhkaan [at] gmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Gazi Üniversitesi /Yönetim Bil. Sis.	2010
Lisans	Gazi Üniversitesi/ Fizik Eğitimi	2006
Lise	Aksu And. Öğr. Lisesi	2000

İş Deneyimi

Yıl	Yer	Görev
2002-2007	Gazi Üniversitesi	Sistem ve Ağ yöneticisi
2007-2008	Meteksan NET	İnternet Sistem Müh.
2008-	Tübitak-ULAKBİM	Sistem yöneticisi

Yabancı Dil

İngilizce

Yayınlar

1. Seral, D., Çalışkan, B., "Pasif Ağ Verileri Üzerinden Düzensizlik Tespiti", 3. Uluslararası Katılımlı Bilgi Güvenliği ve Kriptoloji Konferansı, 152-156, 2008

2. Çalışkan, B., Bektaş, O.**, "**IPv6 İkili Yığın Geçiş Yönteminde Uygulamaların Saldırı Altında Performans Analizi**",** 3. Uluslararası Katılımlı Bilgi Güvenliği ve Kriptoloji Konferansı, 145-150, 2008

3. Gajewski, J., Przybylski, M., Soysal, M., Mitsos, Y., Çalışkan, B., "Caucasian Optical Internet - e-Initiatives Enabler in Southern Caucasus", eChallenges e-2009 Conference, ISBN: 978-1-905824-13-7

4. Çalışkan, B., Güneş, B., "Developing Next Generation E-Learning Infrastructure With IPv6", 3rd International Computer & Instructional Technologies Symposium, 2009