T.C.
KIRKLARELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

GELENEKSEL BİLGİSAYAR AĞLARI VE YAZILIM TANIMLI AĞLARDA DOS ATAKLARININ TESPİTİ VE KARŞILAŞTIRILMASI

YÜKSEK LİSANS TEZİ

Oğuz Gökhan GAMSIZ

Tez Danışmanı: Dr. Öğr. Üyesi Bora ASLAN

Haziran – 2022

"**Geleneksel Bilgisayar Ağları ve Yazılım Tanımlı Ağlarda DoS Ataklarının Tespiti ve Karşılaştırılması**" adlı tez çalışması adlı tez çalışması **Oğuz Gökhan GAMSIZ** tarafından hazırlanmış olup aşağıdaki jüri tarafından **SEÇİM YAPIN** ile Kırklareli Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı'nda Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Danışmanı:

Dr. Öğr. Üyesi Bora ASLAN .....
Kırklareli Üniversitesi

Eş Danışmanı:

Unvanı Adı SOYADI **(Varsa)** .....
Üniversitesi

Jüri Üyeleri:

Unvanı Adı SOYADI .....
Üniversitesi

Unvanı Adı SOYADI .....
Üniversitesi

Unvanı Adı SOYADI **(Varsa)** .....
Üniversitesi

Unvanı Adı SOYADI **(Varsa)** .....
Üniversitesi

Tez Savunma Tarihi: ...../……/……

……
Unvanı Adı SOYADI
Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Kırklareli Üniversitesi Fen Bilimleri Enstitüsü Tez ve Proje Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında; tez içinde sunduğum bilgileri, verileri ve dokümanları, değişik sonuç verebilecek şekilde araştırma araç gereçleri kullanmadan, işlem veya kayıt sonuçlarını değiştirmeden akademik ve etik kurallar çerçevesinde elde ettiğimi, bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu, tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi, bu tezde sunduğum çalışmanın özgün olduğunu bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İmza
Oğuz Gökhan GAMSIZ
24/06/2022

ÖZET

GELENEKSEL BİLGİSAYAR AĞLARI VE YAZILIM TANIMLI AĞLARDA DOS ATAKLARININ TESPİTİ VE KARŞILAŞTIRILMASI

Oğuz Gökhan GAMSIZ

Yüksek Lisans Tezi

Kırklareli Üniversitesi

Fen Bilimleri Enstitüsü

Danışman: Dr. Öğr. Üyesi Bora ASLAN

Haziran 2022, 34 sayfa

Bu tezde yakın zamanda ortaya çıkan ve günümüzde kullanımının artarak devam ettiği yazılım tanımlı bilgisayar ağları ve geçmişten günümüze tüm ağ mimarilerinde kullanılan geleneksel bilgisayar ağlarının hizmet reddi (DoS) saldırılarını tespit etme yöntemleri detaylı bir şekilde karşılaştırılmıştır.

Yapılan bu çalışmada her iki ağında bu ataklara karşı farklı zafiyetleri olduğu açıkça görülmektedir. Fakat YTA'ların programlanabilir olmasının verdiği esneklik ile bu zafiyetlerin geleneksel bilgisayar ağlarına göre kapatılması ve ağın güvenliğinin sağlanması daha olasıdır. Bu da gelecekte YTA'ların daha güvenli bilgisayar ağları olacağı gerçeğini doğurmaktadır.

Anahtar Kelimeler: Geleneksel Bilgisayar Ağları, Yazılım Tanımlı Bilgisayar Ağları, Güvenlik, Servis Reddi Saldırısı

ABSTRACT

DETECTION AND COMPARISON OF DOS ATTACKS IN LEGECY COMPUTER NETWORK AND SOFTWARE-DEFINED NETWORK

Oğuz Gökhan GAMSIZ

MSc Thesis

Kirklareli University

Graduate School of Natural and Applied Sciences

Supervisor: Dr. Öğr. Üyesi Bora ASLAN

Co-supervisor: Title Name Surname

June 2022, 34 pages

In this thesis, the methods of detecting denial of services(DoS) attacks of software defined computer network(SDN), which have emerged recently are increasingly used today and legecy computer network, which have been used in all network architectures from past to present, are compared in detail.

In this study, it is clearly seen that both computer networks have diffrent vulnerabilities against these attacks. However, with the flexibility provided by the programability of SDNs, it is more likely to close these vulnerabilities and ensure the security of the network compared to legecy computer network. This gives rise to the fact that SDNs will be more secure computer network in the future.

Anahtar Kelimeler: Legecy Computer Network, Software Defined Computer Network, Security, Denial of Service Attack

TEŞEKKÜR

Yüksek lisans eğitimim boyunca bana desteklerini sunan ve tecrübesiyle tezimin her aşamasında yardımlarını esirgemeyen saygıdeğer danışmanım Dr. Öğr. Üyesi Bora Aslan'a teşekkürlerimi sunarım.

İÇİNDEKİLER

TABLOLARIN LİSTESİ

ŞEKİLLERİN LİSTESİ

KISALTMALAR

Bu çalışmada kullanılmış kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklamalar
DoS	Servis Reddi Saldırısı(Denial Of Service)
IP	İnternet Protokolü
ISO	Uluslararası Standardizasyon Örgütü
MAC	Medya Erişim Kontrol Adresi
OSI	Açık Sistem Ara bağlantısı(Open System Interconnection)
SNA	Sistem Ağ Mimarisi
TCP	Transmisyon Kontrol Protokolü
UDP	Kullanıcı Veri Bloğu Protokolü
YTA	Yazılım Tanımlı Ağlar

1. GİRİŞ

İnsanlık tarihi boyunca iletişim hep en önemli gereksinimlerden biri olmuştur. Bu gereksinimin yarattığı en büyük teknoloji ise internettir. İnternetin avantajları sayesinde insanlar kıtalararası iletişim kurabiliyor, veri depolayabiliyor, alışveriş yapabiliyor, bankacılık hizmetlerini gerçekleştirebiliyor hale gelmiştir. Fakat internet üzerinden yapılabilen bu işlemler beraberinde güvenlik sorunlarını da getirmiştir. Kötü amaçlı kişiler tarafından herhangi bir verinin çalınması veya herhangi bir ağı servis dışı bırakmak için gerçekleştirilen saldırılar, siber saldırı olarak tanımlanmıştır. En yaygın olarak kullanılan siber saldırı türü ise Denial Of Service (DoS) saldırısıdır. DoS Saldırılarının amacı genellikle saldırdıkları servisi hizmet dışı bırakmaktır. Bu nedenle DoS saldırılarının tespiti ağ üzerinde büyük önem arz etmektedir.

DoS ataklarının tespit edilmesi ve ağın en az zararı görmesi için hem geleneksel bilgisayar ağlarında hem de yazılım tanımlı bilgisayar ağlarında birçok çalışma yapılmıştır. Geleneksel bilgisayar ağları anahtarlar ve yönlendiricilerden oluşur ve bu cihazlar özel bir işletim sistemi ne ve yapılandırmasına sahiptirler. Bu kadar katı cihazlar ile merkezi bir yönetim yapısı oluşturulmasının çok yüksek maliyetleri vardır. Bu nedenle geleneksel bilgisayar ağlarının yerini alabilmesi için veri düzlemi ve kontrol düzleminin birbirinden ayrıldığı ve merkezi bir kontrolcünün olduğu “Yazılım Tanımlı Ağ” fikri ortaya çıktı. YTA mimarisinde veri düzlemi ve kontrol düzleminin ayrı olması onun DoS saldırılarını tamamen durdurmasa da daha esnek ve verimli bir şekilde kullanılabileceği düşünülmüştür.

Tezin ikinci bölümünde geleneksel bilgisayar ağları ve geleneksel bilgisayar ağlarında güvenlikten bahsedilmiştir, üçüncü bölümünde ise geleneksel bilgisayar ağlarında ve yazılım tanımlı ağlarda DoS saldırıları ve DoS saldırılarının tespitinden bahsedilmiştir.

Tezin son bölümünde ise geleneksel bilgisayar ağları ve yazılım tanımlı ağlarda DoS saldırılarının tespit yöntemleri karşılaştırılmalı olarak sunulmuştur.

2. GELENEKSEL BİLGİSAYAR AĞLARI

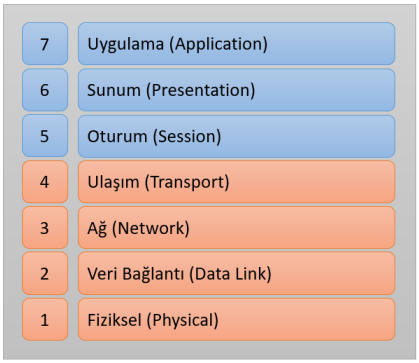
Geleneksel bilgisayar ağları, çok sayıda ağ cihazından oluşan ve bu ağ cihazlarının üzerinde çok sayıda protokolün çalıştığı yapılardan oluşmaktadır. Ağda gerçekleştirilmesi gereken değişikliklerde tüm cihazlara ayrı ayrı dokunmak gerekir. Bu da ağa uyum sağlamada ve politika değişikliklerinde esnekliği azaltmaktadır. Aynı zamanda Ağda yapılacak değişiklikler genellikle donanımsal değişiklikler gerektirdiğinden yüksek maliyetlere sebep olur. Bu nedenle geleneksel bilgisayar ağlarında ağ yönetimi oldukça zordur.

2.1 OSI Katman Modeli

OSI(Open System Interconnection) katman modelini ISO(International Organization for Standardization) tarafından geliştirilmiştir. Bu model ile ağ cihazlarında çalışan uygulamaların birbirleri ile nasıl iletişim kuracakları tanımlanmıştır. ISO son OSI standartını 1984 yılında yayınlamıştır.[1]

OSI katman modeli tanımlanmadan önce ticari ağların çoğunluğu belirli işletmeler tarafından oluşturulmuş standartlar ile oluşturulmuştur. Örnek vermek gerekir ise IBM'in SNA teknolojisi gösterilebilir. OSI modeli öncesi ağların ortak özellikleri, çoğunluk ile yalnızca o üreticinin donanımının kullanılmasına izin verecek biçimde tasarlanmalarıydı. OSI modeli ise bu teknolojilerden ayrı olarak çeşitli üreticilerinin ürünlerinin kullanılabileceği bir ağ yapısı olarak ortaya çıkmıştır. OSI modeli herhangi bir üretici donanımına, ağ mimarisine veya ağ protokollerine göre değişiklik göstermeksizin tüm bunların bir ağ bileşeni gibi kullanılmasını sağlamıştır.

OSI modelinde ağ üzerindeki iletişimi sağlamak için yedi adet katmana ayrılmıştır. Bu katmanlar sırası ile şekil 2.1'de gösterilmiştir.



Şekil 2.1. OSI Referans Modeli

Bir ağ içerisindeki veri iletimi için verinin mutlaka bu yedi katmanı da kullanması gerekmektedir. Verinin geçtiği her katmanda veriye yeni görevler yüklenmektedir. Paket veri iletimi esnasında sırası ile uygulama katmanından fiziksel katmanına doğru giderken veriye her katmanda yeni başlıklar eklenmektedir. Veri alt katmanlara iletilirken iletişim şekli sırası ile; Data halinde alınan bilgi taşıma katmanında segment adı verilen birimlere ayrılır, ağ katmanında segmentler halinde ulaşan verilere adres bilgileri eklenerek paket haline dönüştürülür, veri bağlantı katmanında ise paketlere cihaz MAC adresleri eklenerek frame adı verilen yapılar oluşturulur, son olarak ise frameler fiziksel katmanda bir bit veri dizinine çevrilerek iletim gerçekleştirilir. Veri alıcıya ulaştığında ise tam tersi şekilde fiziksel katmandan uygulama katmanına doğru açılarak hareket eder.

2.1.1 Fiziksel katman

Fiziksel katmanda veriler bit olarak alıcılara iletilmektedir. Bu katman bir ve sıfırlardan oluşan bitlerin nasıl elektrik, ışık veya radyo sinyallerine çevrileceğini ve iletileceğini tanımlamaktadır.[1] Alıcı tarafta ise fiziksel katmana ulaşan bu sinyalleri nasıl bir ve sıfırlara dönüştüreceği tanımlanmıştır. Veri iletiminin gerçekleşebilmesi için verici ve alıcı tarafın aynı kurallar çerçevesinde tanımlanmış olması gerekmektedir.

2.1.2 Veri bağlantı katmanı

Veri bağlantı katmanı fiziksel katmana erişmek için gerekli kuralları tanımlar. Bu katmanda Ethernet veya Token Ring protokolleri çalışmaktadır. Bu erişim yöntemleri verileri kendi standartlarına uygun olarak işlenir.

Alıcıya doğru yönlendirilen paketler bu katmanda framelere dönüştürülerek fiziksel katmana aktarılır. Bu işlemlerin çoğu NIC(Network Interface Card) kartında gerçekleşir. Ayrıca veri bağlantı katmanının ağa bağlı diğer cihazları tanımlama, kablonun o anda kim tarafından kullanıldığını tespiti ve alıcı tarafa fiziksel katmandan gelen verinin kontrolü görevlerini de yerine getirmektedir.

Veri bağlantı katmanı iki bölüme ayrılmaktadır;

- Media Access Control(MAC)
- Logical Link Control(LLC)

Veri bağlantı katmanı içerisinde çalışan MAC bölümü veriyi hata kodları(CRC), alıcı ve vericinin MAC adresleri ile beraber paketleyerek fiziksel katmana göndermektedir. Alıcı tarafta da MAC bölümü CRC kodlarını ve MAC adreslerini inceleyerek paketleri veri bağlantı katmanının ikinci bölüm olan LLC bölümüne aktarmaktadır. LLC bölümü ise bir üst OSI katmanı olan ağ katmanı geçiş görevlerini yürütmektedir. Protokollere özel mantıksal portlar(Service Access Points- SAPs) oluşturarak kaynak ve hedef cihazda aynı protokollerin(TCP/IP - UDP) iletişime geçmesini sağlamaktadır.[2]

Anahtar olarak adlandırılan ağ cihazları ikinci katmanda çalışmaktadırlar. Anahtarlar ikinci katmanda tanımlanan MAC adreslerini algılayabilir ve hedef cihazlar yönlendirebilirler.

2.1.3 Ağ katmanı

Ağ katmanı veri paketlerinin farklı bir ağa gönderilmesi gerektiğinde yönlendiriciler tarafından kullanılacak olan IP bilgilerin eklendiği katmandır. IP Ethernet ya da TCP/IP protokolünü kullanan ağların, ağ üzerinden birbirleri ile veri alışverişi yapmak için kullandıkları 32 bit boyutundaki mantıksal adreslerdir. Bu katmanda veriler paketler halinde taşınmaktadır. Ağ katmanı iki ağ arasında en kısa yoldan verilerin hedef cihaza ulaştırılmasını sağlamaktadır. Aynı zamanda hedef cihaza ulaşacak paketler için mantıksal adreslerin(IP), fiziksel adreslere(MAC) çevrildiği katmandır.

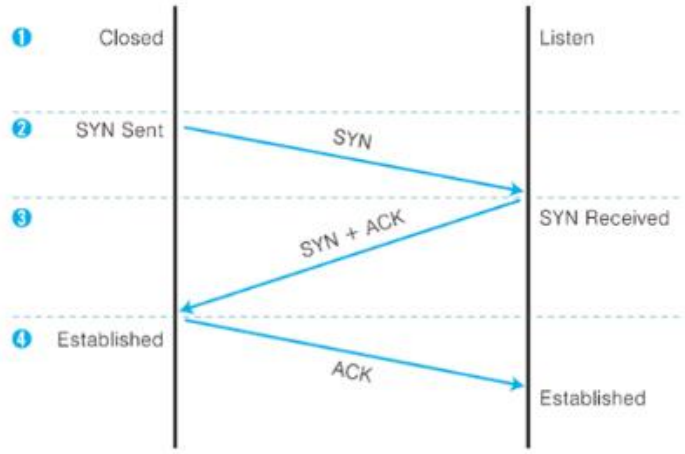
2.1.4 Taşıma katmanı

Taşıma katmanı üst katmanlardan ulaşan verileri ağ paketi boyutunda parçalara ayırarak verinin uçtan uca iletimini sağlamaktadır. TCP, UDP, SPX protokolleri gibi aktarım protokolleri bu katmanda çalışmaktadır.

Bu aktarım protokolleri bu katmanda hata kontrolü gibi görevleri de yerine getirmektedir. Aynı zamanda veri paketlerinin ara belleğe alınması gibi görevlerinde bu katmanda yapılmasından dolayı ağın servis kalitesini arttırmak için kullanılan QoS(Quality of Service) de bu katmanda çalışmaktadır.

TCP protokolü

Gelişmiş bilgisayar ağlarında kayıpsız veri gönderimini sağlayabilmek amacı ile TCP protokolü geliştirilmiştir. TCP protokolünün kayıpsız veri iletiminin yanında verilerin uygun uzunluklu parçalara bölünmesi, her bir parça için sıra numarası verilmesi, kayıp olan veya bozulmuş şekilde hedefe ulaşan verinin tekrar iletilmesi görevleri de vardır. TCP protokolünün çalışma esası üç fazdan oluşur. Öncelikle bir kaynak hedef ile bir bağlantı gerçekleştirir, bağlantı kurulduktan sonra veri iletimi yapılır. Veri iletimi tamamlandıktan sonra ise kaynak hedef ile kurduğu bağlantıyı sonlandırır. Birinci adım olarak bahsedilen kaynak ile hedefin arasındaki bağlantının kurulması ise Üç Zamanlı El Sıkışma(Three way handshake) adı verilen yöntem ile gerçekleştirilir. Bu yöntemde kaynak, hedefe bir "SYN" mesajı yollar. İsteği alan hedef ise isteğin ulaştığına dair bir "SYN-ACK" mesajı yollar. "SYN-ACK" mesajını alan kaynak "ACK" mesajı ile cevap vererek TCP oturumunu kurmuş olur. Üç zamanlı el sıkışma metodu şekil 2.2'de gösterilmiştir.



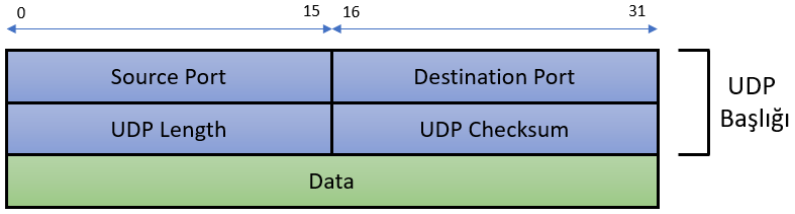
Şekil 2.2. TCP Üç Yönlü El Sıkışma Metodu

UDP protokolü

UDP Protokolü güvenli bir veri iletimine gerek duyulmayan durumlar için kullanılmakta ve veri iletimi için hedef ile kaynak arasında bir anlaşmaya gerek duymaz.

Hedefe ulaşan verilerin kontrolü için “ACK” almaz ve hedefe ulaşan paketler üzerinde herhangi bir sıra numarası uygulamaz. Bu gibi nedenlerden dolayı UDP protokolü TCP protokolüne göre hızlı olmasına rağmen veri iletimini garanti etmez. UDP taşıma katmanında kayıpsız veri iletim gerektirmeyen uygulamalar için geliştirilmiştir.

UDP hedef ile bir bağlantı oturumu, akış kontrolü ve kayıp veriler için tekrar iletim yapmadığı için Geniş alan ağlarında(WAN) ses ve görüntü gibi gerçek zamanlı veri akışının gerekli olduğu uygulamalarda kullanılır. UDP başlık yapısı şekil 2.3’te gösterilmiştir.



Şekil 2.3. UDP Paket Başlığı

2.1.5 Oturum katmanı

Oturum katmanı kaynak ve hedef cihazlar arasındaki bağlantıları kontrol etmektedir. Yereldeki ve uzaktaki bağlantıları kurabilir, yönetebilir ve sonlandırılabilir. Oturum katmanı mesajlaşma kurallarından(full-duplex, half-duplex), uygulamalar arasındaki mesajlaşma kontrolünden, farklı mesajlaşmalara kalınan noktadan devam edilmesinden sorumludur.

2.1.6 Sunum katmanı

Sunum katmanı, uygulama katmanındaki varlıkların arasındaki ortamı kurmakta ve kaynak verinin hedef cihaz tarafından anlaşılacak şekilde çevrilmesini sağlamaktadır. Bunun sonucunda farklı verilerin farklı uygulamalarda kullanılabilmesi sağlanmaktadır. Verilerin şifrelenmesi ve sıkıştırılması da bu katmanda gerçekleştirilir.

2.1.7 Uygulama katmanı

Uygulama katmanı son kullanıcıya en yakın olan OSI katmanıdır. Bu katman iletişim bileşenini yürüten uygulama ile etkileşime girer. Uygulama katmanı sayesinde iletişim kuran kişiler tanımlanarak kaynak kullanılabilirliğine karar verilerek senkronize iletişim sağlanır. Bu katman bilgisayar uygulaması ile ağ arasında bir arabirim sağlamaktadır.

OSI katmanları arasında yalnızca bu katman diğer OSI katmanlarına servis sağlamamaktadır. Bu katman ağ üzerinde kullanıcıların bazı gereksinimlerini karşılamaktadır. Gereksinimlere örnek vermek gerekir ise; SSH,FTP, HTTP, DNS vb.

2.2 TCP/IP Modeli

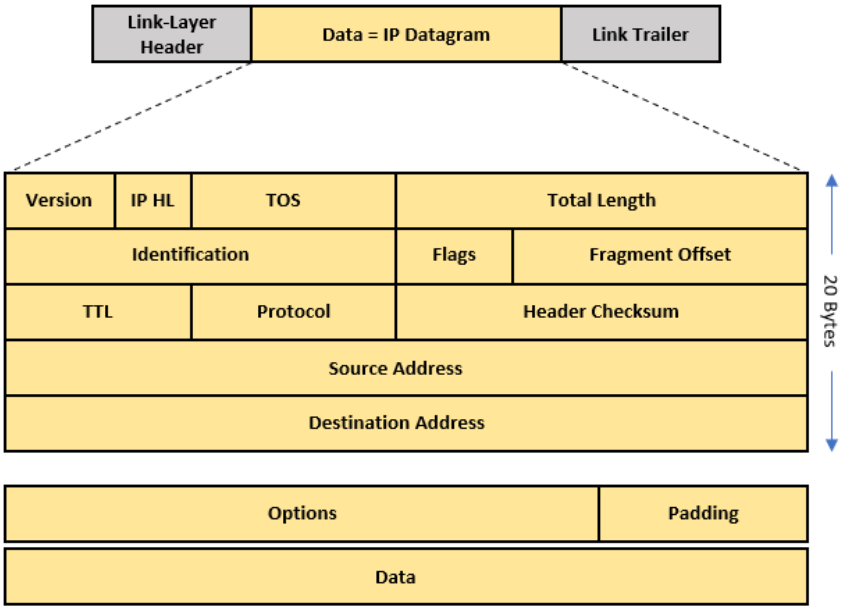
1980'li yıllarda Amerikan Savunma Bakanlığı(DoD) OSI Katman Modeline bir alternatif olması için TCP/IP modelini geliştirmiştir. TCP/IP TCP(Transmission Control Protocol) ve IP(Internet Protocol) protokollerinin bir arada kullanılması ile oluşturulmuş bir network iletişim metodudur. TCP/IP protokol yapısında OSI referans modelinden farklı olarak dört adet katman bulundurmaktadır. Bu katmanlar TCP/IP modelinde sırası ile uygulama katmanı, taşıma katmanı, internet katmanı ve ağ ara yüzü katmanı olarak tanımlanmıştır.[3]

TCP/IP ve OSI modeli arasındaki temel farklardan biri OSI modelinde katmanların katı kurallara sahip olmasından dolayı yeni ve gerekli protokollerin geliştirilmesindeki güçlüktür. TCP/IP modelinin katmanları arasında sert ve katı kuralları olmadığından gerekli durumlarda yeni protokoller mevcut katmanlar arasına daha rahat bir şekilde yerleştirilebilmektedir. TCP/IP ve OSI modeli arasındaki diğer bir fark ise OSI modelinin gerekmeyen katmanların kullanılmaması gibi bir uygulamaya izin vermemesidir. Örneğin, TCP/IP modelinde uygulama katmanında kullanılmamasına rağmen IP üzerinden kullanılabilen protokoller bulunmaktadır. Aşağıda TCP/IP modeline ait katmanlar ve OSI modeline ait katmanlar karşılaştırılmalı olarak verilmiştir.

2.2.1 İnternet protokolü

İnternet Protokolü(IP), ağ adreslemelerinde kullanılan bir yapıdır. IP, kaynak ve hedef arasında paketlerin yönlendirilmesini sağlayan bağlantısız bir protokoldür. İnternet protokolü birden fazla kullanıcının bulunduğu bir ağda bir kaynaktan bir hedefe giden datagramların yönlendirilmesinden sorumludur.[4] Her bir datagramın iki adet bileşeni vardır. Bunlardan biri başlık diğeri ise yüktür.

IP başlığından yönlendirilecek rota için gerekli olan kaynak IP adresi ve hedef IP adresi bulunmaktadır. Yük bileşeni ise veriyi ifade etmektedir. Data gram yapısı şekil 2.4'te verilmiştir.



Şekil 2.4. Datagram yapısı

2.3 Alan Ağları

2.3.1 Kişisel alan ağları (Personal area network)

Bu ağ kullanıcının çeşitli cihazlar ile arasında kablolu veya kablosuz olmak üzere kurulmuş olan bir ağ yapısıdır. Kişisel alan ağları kişisel aygıtların haberleşmesinin yanında kullanıcının internet veya başka bir ağa bağlantı gerçekleştirmesi amacı ile de kullanılabilir.

2.3.2 Yerel alan ağları (Local area network)

Yerel alan ağları genellikle 2 kilometre mesafeyi geçmeyecek şekilde bulunan kullanıcıların bir ağ üzerinden bir araya getirilmesi ve veri iletimi gerçekleştirebilmeleri için oluşturulan ağ yapılarıdır.

2.3.3 Metropol alan ağları (Metropolitan area network)

Metropol alan ağları bir şehir veya bir yerleşkede bulunan bilgisayarların birbirleri ile veri iletimi yapılabilmesi için kurulan ağlardır. Bütün bir şehrin bloklarına kadar yerel alan ağlarından daha geniş bir coğrafya için kullanılmaktadır.

2.3.4 Kampüs alan ağları (Campus area network)

Kampüs ağları sınırlı bir coğrafya içerisinde birden fazla yerel alan ağının birbirine bağlanmasına olanak sağlayan bir ağdır. Yoğun olarak üniversite ve okullarda kullanılmaktadır.

2.3.5 Geniş alan ağları (Wide area network)

Geniş alan ağları farklı coğrafyalarda yer alan yerel alan ağlarının karşılıklı olarak haberleşmesini sağlayan ağ yapılarıdır. Birden fazla yerel alan ağının birleşmesi ile oluşturulan geniş alan ağları, yerel alan ağlarına göre oldukça büyük ve oldukça kompleks bir yapıya sahiptirler.

2.4 Ağ Cihazları

Birden fazla bilgisayarın veri iletimi yapabilmesi için ilave donanımlara ihtiyaç duyulmaktadır. Bu cihazlar OSI referans modelinde bulunan katmanlarda hizmet vermektedirler.

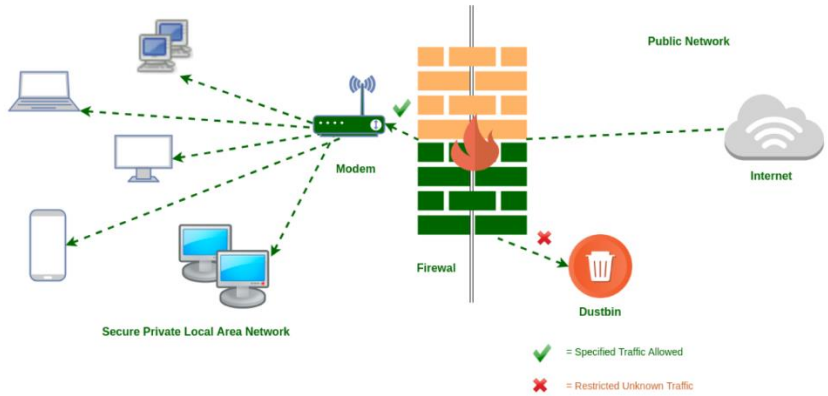
- *HUB:* Kendisine bağlı olan tüm cihazlar için bir iletim yolu sunar. Kaynak tarafından iletilen veriyi tüm portlarından yayar. Bu nedenle HUB tek bir çarpışma alanına sahip olduğundan hat üzerinden bir trafik varken ağa bağlı diğer cihazların hattın boşaltılmasını beklemesi gerekir. HUB cihazları OSI modelinin fiziksel katmanında çalışmaktadır. HUB cihazları aktif ve pasif olarak iki kısımda incelenmektedir. Aktif HUB cihazları sinyal seviyelerini güçlendirirken pasif HUB cihazları sinyal seviyelerini güçlendirmeden trafiği çoklu ortama yayar.

- *Anahtar:* Anahtarlar, HUB'lar gibi kendisine bağlı olan bilgisayarlara bir veri iletim yolu sunar. Anahtarlar HUB'lardan farklı olarak anahtarlamalı veri iletimi yaparlar. Ağa bağlı iki cihaz arasında yapılacak olan veri iletimini tüm portlarına bağlı olan bilgisayarlara değil yalnızca trafiğin iletilmek istendiği hedefe doğru gönderir. Kaynaktan gelen verileri MAC adreslerine göre yönlendirdiklerinden dolayı MAC adreslerine bağlı çarpışma alanları yaratırlar ve iki bilgisayar arasında veri iletimi mevcutken bile diğer kullanıcılar da aralarında veri iletimi yapabilirler. OSI modelinin veri bağlantı katmanında hizmet verirler.

- *Köprü:* İki veya daha fazla aynı protokolü kullanan bilgisayar ağlarını birbirine bağlamak için kullanılan ağ cihazıdır. Köprü içerisinde erişim yapılabilecek adres tablosunu barındırarak verinin hangi ağdan hangi ağa gittiğini belirler ve adres tablosu ile eşleşiyorsa verinin iletilmesine izin verir. Köprü ağda filtre görevi adreslenemeyen ağların geçmesine izin vermez. OSI modelinin veri bağlantı katmanında hizmet verir.

- *Yönlendirici:* Bir yönlendirici iki veya daha fazla ağ arasında iletimi sağlayan ağ cihazlarıdır. Aralarında iletim sağlayacağı her ağ için üzerinde bir ara yüz taşır. Her iki ağında ara yüzüne sahip olan yönlendiriciler paketlerin bu ağlar arasında iletilirken yollarını bulmalarını sağlar. Bu cihazlar rota seçimleri için üzerlerinde yönlendirme protokolleri çalıştırarak hedefe gidecek en iyi yol seçimlerini yaparlar. OSI modelinin ağ katmanında hizmet verirler.

- *Güvenlik Duvarı:* Temel amacı ağdaki gelen trafik ile giden trafiği kontrol etmek ve belirlenmiş olan ağ kurallarına uyarak trafiği yönlendirmektir. Bunun yanı sıra aynı zamanda daha gelişmiş kontrol metotları da kullanmaktadır. Bu kontrol yöntemlerinden biri ise «Proxy Service» adı verilen yöntemdir. Bu yönteme göre ise gelen trafikler kaydedilerek detaylı bir incelemeden geçirilir ve diğer meşru veriler ile karşılaştırılır. Eğer veriler içerisinde zararlı bir yazılım kodu olarak güvenlik duvarı üzerinde tanımlanmış bir kod ile eşleşirse, paket zararlı olarak işaretlenerek ağa girişine izin verilmez. Şekil 2.5'te bir güvenlik duvarının geleneksel bir ağ mimarisinde nasıl konumlandırıldığı gösterilmiştir.



Şekil 2.5. Güvenlik duvarının geleneksel ağ mimarisinde konumlandırılması

2.5 Geleneksel Bilgisayar Ağlarında Güvenlik ve Tehditler

Bilgisayar ağlarında güvenlik, verileri, uygulamaları, donanımları ve ağları saldırı veya yetkisiz erişimlerden korumak olarak tanımlanabilir. Diğer bir deyişle verileri çalmak, yok etmek, donanımları etkisiz hale getirmek için yapılan saldırılara siber saldırı denir. Bilgisayar ağlarında alınan güvenlik önlemleri bu saldırılardan kaynaklanacak zararları en aza indirgemeyi amaçlar. Saldırganın amaçları ve kullandığı teknikler doğrultusunda farklı siber saldırı türleri bulunmaktadır. Bunlardan bazıları; ortadaki adam(man -in-the-middle), servis reddi(DoS) ve zararlı yazılım(malware) saldırılarıdır.[5]

2.5.1 Ortadaki adam(Man-in-the-middle)

Ortadaki adam saldırısında saldırgan kendini ağda gerçekleşen iki bağlantı arasındaki verilerin dinlenmesi ve verilerin ele geçirilmesini amaçlamaktadır. Aynı zamanda saldırgan bu verileri dinlemek ile birlikte verileri manipüle de edebilmektedir. Kablosuz ağlarda paketlerin tamamen broadcast olarak yayınlanmasını kullanarak hiçbir işleme gerek kalmadan paketler saldırgan tarafından ele geçirilebilir.

2.5.2 Servis reddi(DoS)

Servis reddi saldırıları hedef sistemde çalışan servislerin devre dışı bırakılmasını amaçlar. Bu saldırılar iki şekilde meydana gelebilmektedir. Hedefin bant genişliği, işlemci veya hafıza kaynaklarını tüketerek ya da serviste bulunan bir zafiyetten yararlanarak bu saldırılar yapılabilir.

DoS saldırılarının çeşitleri;

- UDP Flood saldırısı
- TCP-SYN Flood Saldırısı
- Smurf Saldırısı
- Ping of Death saldırısı
- Buffer Overflow saldırısı
- IP Fragmentation saldırısı
- Land taşma saldırısı

2.5.3 Zararlı yazılım(Malware)

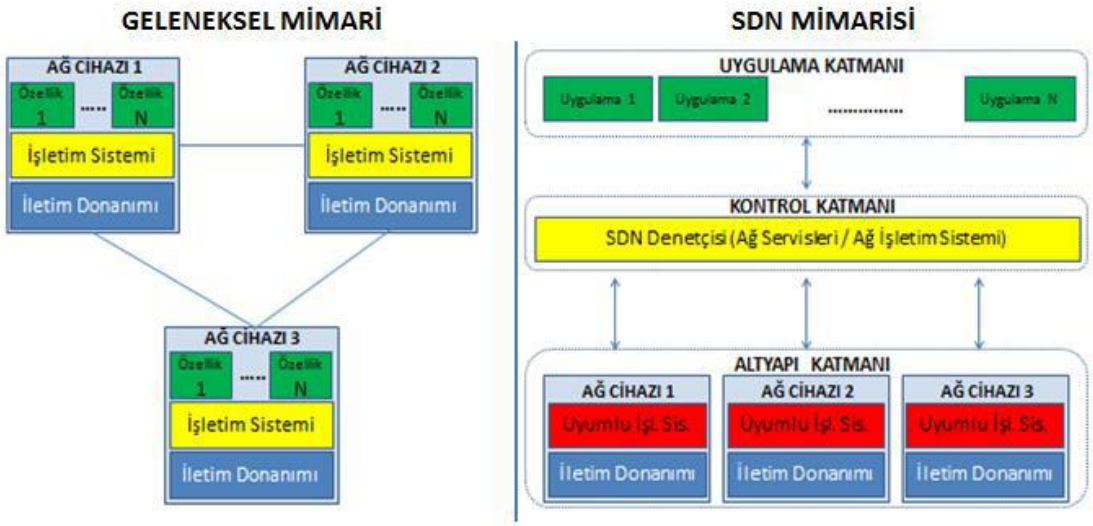
Zararlı yazılım saldırısında saldırganlar hedef donanımları devre dışı bırakmak, veri hırsızlığı ve özel sistemlere erişim amaçları ile kullanmaktadır. Kötü amaçlı yazılımlar uzun süreler boyunca sistem zafiyetlerinden yararlanarak bilgi toplama ve casusluk işlemleri yaparlar.

3. YAZILIM TANIMLI AĞLAR

Yazılım tanımlı ağ, fiziksel olarak ayrılmış fakat mantıksal olarak merkezi olarak kontrol edilebilen bir ağ kontrol yazılımıdır. Geleneksel ağ cihazlarını kullanarak kısa sürede düşük maliyetler ile sanal bir ağ kurulmasını mümkün kılar. YTA ile kurulmuş bir ağda değişiklik yapmak çok kolay ve maliyetsizken geleneksel bilgisayar ağlarında bu donanım bağımlılığı nedeni ile oldukça zor ve maliyetlidir. Bu teknoloji daha önce 2006 yılında SANE ile tanıtılmasına rağmen, son zamanlarda popülerlik kazanmıştır. SANE, kullanıcının kimliğini doğrulamak ve ağdaki ilkeleri uygulamak için merkezi bir kontrolcü gibi kullanıldı. SANE başlığının geliştirilebilmesi için internet protokol kümesinin değiştirilmesi gerektiğinden daha sonra reddedildi. Bu gereksinimin ortadan kaldırılabilmesi için 2007 yılında ETHANE geliştirildi. ETHANE mevcut IP kümesini bir değişiklik gerektirmeden kullanabiliyordu. SANE ve ETHANE'den ilhan alan YTA 2008 yılında Stanford üniversitesinde Nick Mckeown ve bir gurup araştırmacı tarafından OpenFlow'un icadı ile başladı. SDN, kontrol düzlemi ve veri düzlemini birbirinden ayırarak bunların arasındaki iletişimi OpenFlow yardımı ile sürdürmektedir. Kontrol düzlemi veri düzleminden ayırabilmek, kontrol düzlemini merkezileştirilebilir ve programlanabilir kılmaktadır. Bu da YTA'ya geleneksel ağlarda olmayan esnek ve ölçeklenebilirlik özelliğini kazandırmaktadır. Merkezileştirilmiş kontrolör, ağ davranışlarını kontrol ederek ağı tek bir yerden yönetebilmektedir.

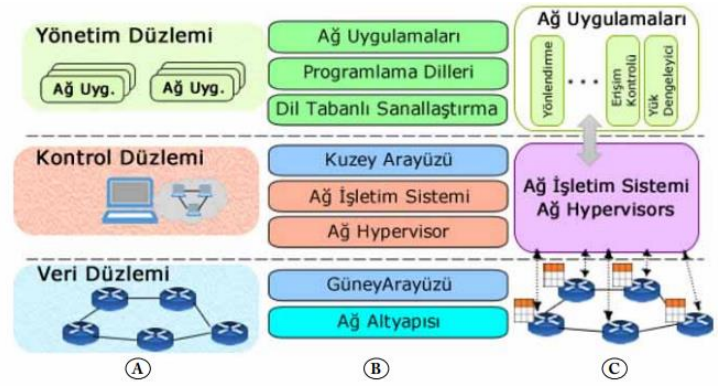
3.1 Yazılım Tanımlı Ağ Mimarisi

ONF, YTA mimarisinin üç katmandan ile temsil edildiğini ve bu katmanların; veri katmanı, kontrol katmanı ve yönetim katmanı olduğunu ifade ede etmiştir. ONF'nin ortaya koyduğu YTA mimarisinin geleneksel bir ağ mimarisi ile arasındaki farklılıklar şekil 3.1'de görülebilmektedir.



Şekil 3.1. ONF YTA mimarisi ve geleneksel ağ mimarisinin karşılaştırılması

Kreutz ve arkadaşları ise katman fonksiyonlarını da ayrıntılı bir şekilde açıklayarak sekiz katmandan oluştuğunu ifade etmiştir. Kreutz'a göre bu katmanlar; ağ altyapısı, güney ara yüzü, ağ hypervisor, ağ işletim sistemi, kuzey ara yüzü, dil tabanlı sanallaştırma, programlama dilleri ve ağ uygulamalarıdır. Kreutz'un bu mimarisinde güney ara yüzü, ağ işletim sistemi, kuzey ara yüzü ve ağ uygulamaları yalnızca YTA mimarisine özgüdür. Yazılım tanımlı ağlar için düzlem yapısı, katman yapısı ve sistem mimarisi şekil 3.2'de verilmiştir.[8]



Şekil 3.2. a)Düzlem Yapısı b)Katman Yapısı c)Sistem tasarım mimarisi

3.1.1 1.Katman: ağ altyapısı

Anahtar ve yönlendirici gibi ağ cihazlarından oluşur. YTA modelinin bu katmanı geleneksel ağ altyapısı ile benzerlik gösterir. En büyük farkları ise geleneksel ağ cihazları kendi yazılımları ile otonom kararlar alırlarken YTA mimarisinde bu cihazlar yalnızca basit yönlendirme işlemleri yapmaktadır. YTA'da bu karar mekanizması ağ cihazlarından alınıp merkezi kontrolöre devredilmiştir. [9-10]

3.1.2 2.Katman: southbound ara yüzü

Soutbound ara yüzü, ağ cihazları ile kontrol katmanı arasında bulunur. Kontrol ve veri katmanının ayrılmasını sağlayan katmandır. Southbound ara yüzünün en yaygın olanları Openflow ve NetConf(Network configuration protocol)dur.

3.1.3 3.Katman: ağ hypervisors

Günümüzde sanallaştırma teknolojileri etkin bir şekilde kullanılmaya başlanmıştır. Sanal sunucu sayılarının fiziksel sunucu sayılarını aşması bunun bir göstergesi olarak görülmektedir. Hypervisors, farklı sanal makinelerin benzer donanım kaynaklarını paylaşmasına izin vermektedir . Bu mimaride de hypervisors fiziksel ağdan bağımsız olarak sanal ağlar kurulmasına olanak verir. Bir ağı oluşturan fiziksel ağ cihazları(anahtar, yönlendirici, güvenlik duvarı) sanallaştırılarak bu kaynakların yönetimi hypervisors üzerinden yapılır.[10]

3.1.4 4.Katman: ağ işletim sistemi

YTA mimarisinde ağ yöneticisi tarafından belirlenen ilkelere dayalı ağ yapılandırmasını oluşturmak için kontrol mantığının en önemli öğesi olan ağ işletim sistemi (NOS) önemli bir unsurdur.

Ağ işletim sistemi, uygulama tasarımı ve performansı, ağ ölçeklenebilirliği, ağ durumu ve ağın servis kalitesi(QoS) gibi görevleri yürütmektedir. Ağ işletim sistemleri bir sunucu üzerinde çalışan ve ağın merkezi bir şekilde yönetilmesini sağlayan bir yazılımdır. Fakat yalnızca bunun ile sınırlı kalmayarak YTA ağ sorunlarını çözümleyerek ağın sürekli çalışabilirliğini temin etmektedir. NOX, ONIX, Maestro, NOSIX gibi birçok farklı ağ işletim sistemi bulunmaktadır.[9]

3.1.5 5.Katman: northbound ara yüzü

Northbound ara yüzü kontrol düzlemi ile yönetim düzlemi arasında yer alır. Yönetim düzleminde hizmet veren uygulamalar ile ağ işletim sistemi arasında bir iletişim kurulmasını sağlar.

3.1.6 8.Katman: ağ uygulamaları

Ağ uygulamaları katmanı YTA mimarisinin en önemli katmanı olarak görülmektedir. Ağ cihazlarının davranışlarını belirleyerek ağda bulunan tüm cihazlara yönlendirme kurallarının sırasıyla yüklenmesini sağlar. Bu yöntem ile mevcut ağlarda kullanılan yük dengeleme, ağ güvenliği, yönlendirme vb. gibi ağ uygulamalarının yanında güç tüketimi gibi yeni özelliklerin de kazandırılmasını sağlar.

3.2 OpenFlow Protokolü

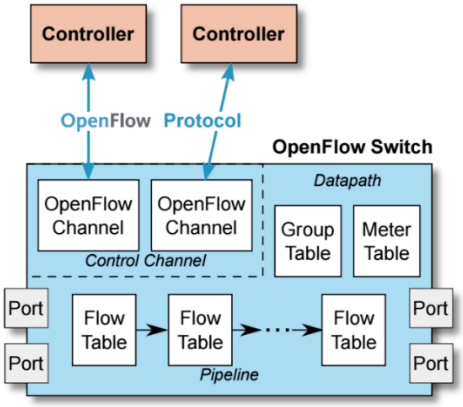
OpenFlow protokolü, 2008 yılında Stanford üniversitesinde Nick Mckeown ve bir gurup araştırmacı tarafından ortaya atılmıştır. SDN, kontrol düzlemi ve veri düzlemini birbirinden ayırarak bunların arasındaki iletişimi OpenFlow ile gerçekleştirmektedir. OpenFlow protokolünün geliştirilmesi amacı ile 2011 yılında Open Networking Foundation(ONF) kurulmuştur.

Uygulama katmanında oluşturulan yönlendirme kurallarının denetleyici tarafından ağ cihazlarına gönderilebilmesi için OpenFlow protokolü kullanılmaktadır. Ağ işletim sistemi OpenFlow protokolü ile southbound ara yüzünden ağ cihazlarına bu yönlendirme kurallarını yüklemekte ve böylece ağ cihazları aktif rota hesapları değil de yalnızca basit yönlendirme işlemleri yapmaktadırlar. [10-11-12]

3.2.1 OpenFlow anahtar

OpenFlow anahtarları, güvenli bir iletişim kanalı ile OpenFlow protokolü kullanılarak kontrolcü ile iletişim kuran etkin veri anahtarlarıdır. Bu anahtarlar bir veya daha fazla akış tablosuna ve bir grup tablosuna sahiptirler. OpenFlow anahtarlar sahip oldukları bu akış tabloları ve grup tablosuna göre paket araması ve yönlendirme yaparlar. Kontrolcü OpenFlow protokolünü kullanarak bu akış tablolarında değişiklikler yapabilir ya da bu akış tablolarına yenilerini ekleyebilir. Eğer bir OpenFlow anahtarında birden fazla akış tablosu mevcut ise anahtar paket eşleştirmelerine ilk akış tablosundan başlar ve son akış tablosuna kadar devam ederek paketin eşleşeceği akış tablosunu arar.

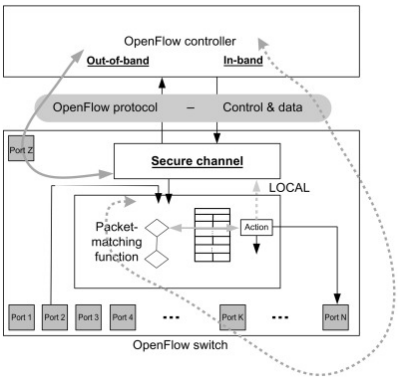
Gelen paketler ilk akış tablosu ile eşleşmek zorundadır, diğer akış tabloları ilk akış tablosunun çıktılarına göre oluşturulmaktadır. Aşağıda OpenFlow anahtar mimarisi şekil 3.3'te gösterilmiştir.



Şekil 3.3. OpenFlow anahtar mimarisi

3.2.2 OpenFlow kontrolcü

OpenFlow Kontrolcüsünün, ağ kurallarının takibi, kuralların güncellenmesi, kuralların OpenFlow anahtarlarına dağıtılması gibi görevleri vardır. Kontrolcü OpenFlow anahtarları üzerindeki akış tablolarında güncelleme yaparken TLS/SSL gibi güvenli iletişim kanalları kullanmaktadır. Birçok farklı programlama dili ile yazılmış OpenFlow kontrolcüleri bulunsa da bunların en yaygınları Python programlama dili ile yazılmış POX ve C++ programlama dili ile yazılmış NOX kontrolcülerdir. POX kontrolcü, NOX kontrolcüsü temele alınarak geliştirilmiş ve Python programlama dilinin kattığı esneklikler dolayısı ile geliştirilmesi ve yeni kabiliyetler kazandırılması NOX'a göre daha basit bir hal almıştır. Aynı zamanda POX kontrolcüsü Linux, Unix, Windows gibi neredeyse bütün işletim sistemlerinde çalışabilmektedir. OpenFlow kontrolcü mimarisi şekil 3.4'te gösterilmiştir.



Şekil 3.4. OpenFlow kontrolcü mimarisi

3.3 Yazılım Tanımlı Ağlarda Güvenlik ve Tehditler

Yazılım tanımlı ağlar geleneksel ağlara esneklik ve programlanabilirlik katarak güçlendirir, merkezi yönetim sağlar ve maliyetleri düşürür. Güvenlik bakış açısı ile bakıldığında ise cihazların kimliğini doğrulamaya ve merkezi olarak analiz etmeye yardımcı olan bir görünüm sağlayarak saldırıları tespit etmeyi ve azaltmayı amaçlar. Fakat kontrol ve veri düzleminin ayrılması ile birlikte de en kötü senaryoda merkezi kontrolörün darboğaz haline gelerek yeni güvenlik açıkları ortaya çıkaracağı görülmektedir.[13]

Yazılım tanımlı ağlarda saldırı ve güvenlik açıklarını yetkisiz erişim, veri sızıntısı, veri değişikliği, kötü amaçlı/tehlike uygulamalar, servis reddi, yapılandırma sorunları ve yazılım tanımlı ağların sistem seviyesi güvenliği olduğunu belirtmişlerdir. Aynı zamanda YTA'dan yeni nesil iletişim ağı olma potansiyeline sahip erken bir aşamada yeni bir ağ paradigması olduğundan, kullanılabilirlik ve güvenilirlik tarafında çok şey bekleniyor. Yazılım tanımlı ağ mimarisinde 7 adet tehdit vektörü ortaya koyulmuştur. Bu tehdit vektörlerinin dördü hem geleneksel bilgisayar ağlarında hem de yazılım tanımlı ağlarda bulunurken geri kalan üçü ise yalnızca yazılım tanımlı ağlarda mevcuttur. Tablo 3.1'de tehdit vektörlerinin YTA'ya özgü olup olmadığı ve YTA'daki sonuçları kısaca gösterilmiştir.

Tehdit Vektörleri	YTA'ya Özel Mi?	YTA Üzerindeki Etkileri
Vektör 1	Hayır	DoS saldırıları için zafiyet yaratır.
Vektör 2	Hayır	Potansiyel atak zafiyeti yaratır
Vektör 3	Evet	Merkezi kontrolcünün hizmet dışı bırakılmasını amaçlar
Vektör 4	Evet	Hatalı bir kontrolcü tüm ağı tehlikeye atabilir.
Vektör5	Evet	Merkezi kontrolcü üzerinden kötü amaçlı yazılımlar dağıtılabilir.
Vektör 6	Hayır	Potansiyel atak zafiyeti yaratır.
Vektör 7	Hayır	Hata ayıklaması sırasında negatif etkiler yaratabilir.

Tablo 3.1. YTA'lara özgü tehdit vektörleri

- *Vektör 1;* Switch ve yönlendirici gibi ağ cihazlarına saldırmak için ağ içerisinde oluşturulan sahte trafikler ile yaratılan tehdittir.[13] Yazılım tanımlı ağlarda fiziksel ve sanal ağ cihazlarına DoS saldırı başlatmak için OpenFlow protokolü kullanılır.

- *Vektör 2;* Ağ içerisinde bulunan anahtar cihazlarının güvenlik zafiyetlerinden yararlanılarak oluşturulan bir tehdittir. Ağda bulunan trafiği uzatmak, sahte trafik oluşturmak, trafikleri düşürmek veya paketleri yavaşlatmak için kullanılabilir.

- *Vektör 3;* Yazılım tanımlı ağlarda kullanılan TLS(Transport Layer Security) / SSL(Secure Socket Layer) protokolleri kontrolcü ve anahtarlar arasında güvenli iletişimi sağlamak için yeterli değildir. Kötü niyetli biri ağın kontrol düzlemine eriştiğinde ağda trafik sızıntılarına ve DoS saldırılarına sebebiyet verebilir.[13]

- *Vektör 4;* Hatalı bir kontrolör tarafından oluşan tehdittir. Bu tehdit sonucunda kontrolcünün davranışları değişebilir ve kontrolcü tüm ağı tehdit edebilir.

- *Vektör 5;* Yazılım tanımlı ağlarda kontrolörün kötü amaçlı bir yazılıma maruz kalması sonucu oluşan tehdittir. Böyle bir tehdit ağı servis dışı bırakabilirken diğer kötü amaçlı yazılımların kullanılmasına da olanak sağlayabilir.

- *Vektör 6;* Hem geleneksel bilgisayar ağlarında hem de yazılım tanımlı ağlarda görülen bu tehdit potansiyel güvenlik açıklarından yararlanarak ağa zarar vermeyi amaçlar.

- Vektör 7; Hem geleneksel ağlarında hem de yazılım tanımlı ağlarda oluşan bu tehdit, problem ve sorun esnasında kötü amaçlı kişiler tarafından yaratılan negatif efektler ile sorunun ve problemin çözümüne engel olmaktadır.[13]

  Bu tehdit vektörlerinin yanında YTA'nın mimarisinde bulunan veri katmanında, kontrol katmanında ve yönetim katmanında güvenlik konuları ayrıntılı olarak incelenmiştir.

3.3.1 Veri katmanında güvenlik

Hizmet reddi saldırılarının en çok yaşandığı katmandır. Bir saldırgan bir ağ cihazının kontrolünü ele geçirebilir ve cihazı kaynakları tüketecek şekilde yüksek trafikler oluşturulmasını sağlayabilir. Bu da merkezi kontrolcünün isteklere cevap vermesini durduracaktır.

DoS saldırılarının bu denli tehlikeli olması sebebi ile SOM adı verilen yapay zeka geliştirilmiştir. SOM, belirli bir trafiğin sahte veya gerçek olup olmadığını tespit etmek için trafik başına ortalama paket ve trafik başına ortalama bayt gibi mekanizmaları kullanan bir yapay zeka ürünüdür. Ancak SOM makine öğrenmesi yapan bir yapay zeka olduğu için olası ağ trafiğine aşina olabilmesi için trafiği uzun süreler dinlemesi gerekmektedir. [14]

Bu katmanda karşılaşılan diğer bir siber saldırı tipi ise ortadaki adam saldırısıdır. Ortadaki adam saldırısı ile saldırgan bu katmanda trafikleri gizlice dinleyebilmekte ve veri hırsızlığı yapabilmektedir. Fakat Benton ve arkadaşları bu tehdidin, kontrolör ve ağ cihazları arasında TLS gibi güvenli bir protokol kullanarak azaltılabileceğini iddia etmektedirler.

3.3.2 Kontrol katmanında güvenlik

Bu katmandaki en tehlikeli saldırılardan biri tüm ağı devre dışı bırakabilecek bir hizmet reddi saldırısı ile kontrolcüde yıkıcı etkiler yaratmaktır. Bu zafiyeti azaltmak için Shin ve arkadaşları "TCP-SYN" taşması ile kontrol düzlemi doygunluğunu durduran bir yazılım olan AVANT-GUARD'ı önerdiler. Fakat AVANT-GUARD UDP ve ICMP gibi diğer protokollerin saldırılarına maruz kalabiliyordu. Ancak kontrolörün hata toleransının düşürülmesi ve aynı zamanda güvenliğinin sağlanabilmesi amacı ile Kreutz ve arkadaşları kontrolörün kopyalanmasını önerdi. Bu fikir ile birlikte kontrolör üzerinde görülen saldırılarda dahi hızlı bir aksiyon ile alınan zararlar minimuma indirilecektir.

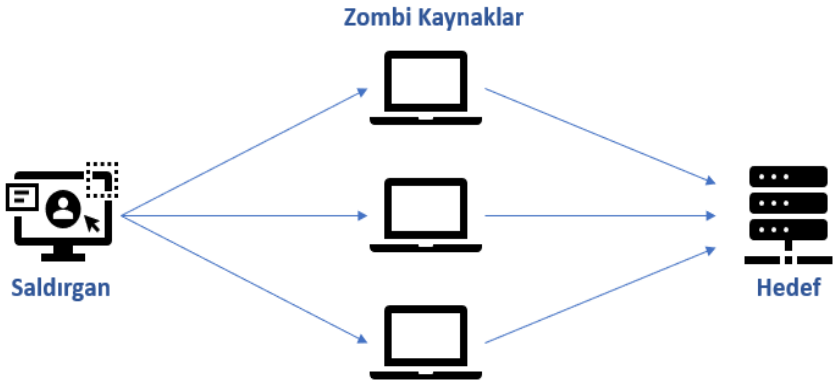
3.3.3 Yönetim katmanında güvenlik

Jarra ve arkadaşlarının tanımlarına göre, bu katmandaki güvenlik açığı kontrolör ve uygulama arasındaki güvendir. SDN, uygulamaların ağdaki değişiklikleri uygulamasına izin verirken herhangi bir güvenlik politikası yürütmez ve doğrulama teknikleri kullanmaz. Kötü amaçlı yazılımlar ile bu zafiyetten yararlanılarak ağın hizmet dışı kalması sağlanabilir. Sonkoly ve arkadaşları ve Sherwood ve arkadaşları, ağ sanallaştırma izolasyonunu uygulama katmanındaki bir dilim olarak, tehditleri tek bir dilimde içeren ve güvenlik ihlallerinin diğer dilimlere yayılmasını önleyen bir mekanizma olarak tanımladılar. Fakat bu izolasyon yalnızca tehdidin yayılmasını önleyerek kontrol altına alınmasına olanak sağlar, tehdidi azaltmayı garanti etmez.[14]

4. DENIAL OF SERVICE (DoS) SALDIRILARI

Bilgisayar ağlarında güvenlik, verileri, uygulamaları, donanımları ve ağları saldırı veya yetkisiz erişimlerden korumak olarak tanımlanabilir. Diğer bir deyişle verileri çalmak, yok etmek, donanımları etkisiz hale getirmek için yapılan saldırılara siber saldırı denir. Bilgisayar ağlarında alınan güvenlik önlemleri bu saldırılardan kaynaklanacak zararları en aza indirgemeyi amaçlar. Saldırganın amaçları ve kullandığı teknikler doğrultusunda farklı siber saldırı türleri bulunmaktadır. Bu türler arasında en çok tercih edileni ise DoS saldırılarıdır.

Servis reddi saldırıları hedef sistemde çalışan servislerin devre dışı bırakılmasını amaçlar. Bu saldırılar iki şekilde meydana gelebilmektedir. Hedefin bant genişliği, işlemci veya hafıza kaynaklarını tüketerek ya da serviste bulunan bir zafiyetten yararlanarak bu saldırılar yapılabilir. Saldırganlar genel olarak zararlı yazılımlar ile ele geçirdikleri ve kendilerine ait olmayan donanımları da kullanarak yüksek miktarlarda trafik yaratabilirler. Aynı zamanda saldırganların DoS saldırılarını yaparken kullandıkları sahte IP adresleri saldırganların tespit edilmesini imkansız kılmaktadır.



Şekil 4.1. Dağıtılmış DoS saldırısı

4.1 Denial Of Service (DoS) Saldırı Türleri

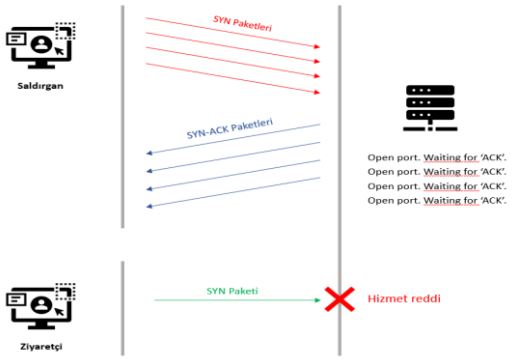
UDP Flood Saldırısı: UDP flood saldırısı, bir istemcinin bağlantı noktalarından birine gönderilen UDP paketine yanıt verdiğinde attığı adımlardan yararlanarak çalışır. Normal koşullar altında bir istemci bağlantı noktasından bir UDP paketi aldığında yanıtı 2 adımda verir.

- İstemci bağlantı noktasında istekleri dinleyen herhangi bir uygulamanın çalışıp çalışmadığının kontrolünü sağlar.
- Bu bağlantı noktasında paketleri dinleyen herhangi bir uygulama çalışmıyorsa, istemci hedefe ulaşamadığını bir ICMP paketi ile geri bildirir.

Bu işlemler sırasında istemci gelen paketi dinleyen bir uygulama olup olmadığının kontrolünü sağlarken ve tekrar istemciye ICMP paketleri ile geri dönüş yaparken art arda alınan UDP istekleri ile istemci boğulur. Alınan her UDP paketinin kontrolü ve ardından yanıtı için kaynak harcayan istemci yüksek miktarda alınan UDP paketleri sonucunda kaynaklarını tüketir ve normal trafikler için hizmet reddine sebebiyet verir.

TCP-SYN Flood Saldırısı: TCP-SYN flood atakları istemci kaynaklarını tüketmek için TCP üç yönlü el sıkışmasının bir kısmını kullanan bir hizmet reddi saldırısıdır. Normal bir istemci ve sunucu TCP üç yönlü el sıkışması 3 adımda gerçekleşir.

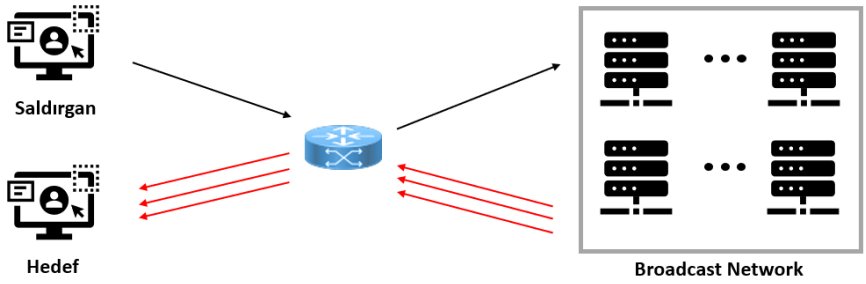
- İstemci, sunucuya bir SYN mesajı gönderir ve bir talepte bulunur.
- Sunucu, istemciye bir SYN-ACK mesajı gönderir ve istemcinin gönderdiği SYN mesajını onaylar.
- İstemci bir ACK mesajı ile sunucuya onay yanıtı verir ve TCP bağlantısı kurulur.



Şekil 4.2. TCP-SYN saldırısı

Bir TCP-SYN flood saldırısında ise saldırgan sunucuya tekrarlanan SYN paketleri gönderir. Sunucu meşru olarak görünen tüm SYN paketlerine karşılık SYN-ACK paketi ile TCP denemesine yanıt verir. Saldırgan ya aldığı SYN-ACK paketlerine ACK paketleri ile karşılık vermez ya da sahte bir IP adresi kullanarak SYN-ACK paketlerini hiç almaz. Sunucu bu her iki durumda da bir süre boyunca SYN-ACK paketlerinin kötü niyetli saldırgan tarafından onaylanmasını bekleyecektir. Bu süre boyunca sunucu saldırgana TCP-RST paketi göndererek bağlantı isteklerini kapatamaz ve istekler açık kalır. Bu bağlantı istekleri kapanmadan saldırgan tarafından birçok bağlantı isteği daha gelmeye devam edecek ve çok sayıda TCP oturumu sunucu üzerinde açık kalacak. Saldırının sonunda bağlantı tabloları doldukça sunucu meşru istemcilere olan bağlantıları için de hizmet reddine sebebiyet verecek.

Smurf Saldırısı: Smurf saldırısı adını kötü amaçlı yazılımdan alan bir DoS saldırısıdır. Smurf saldırıları da ICMP istek paketi göndererek gerçekleştirilmektedir ancak smurf yayın ağlarının özelliklerinden yararlanarak ağa verilen hasarın potansiyelini arttırmaktadır. Standart bir smurf saldırısında saldırgan bir istemciye ICMP paketi göndererek bir yanıtı tetikler. Bir yayın ağında her istemciye gönderilen ICMP istekleri için alıcı her istemciden yanıt bekler. Saldırganlar yayın ağlarında saldırı trafiğini arttırmak için bu yöntemi kullanırlar.



Şekil 4.3. Smurf saldırısı

Ping Of Death Saldırısı: Ping of Death(PoD) saldırısı, saldırganın maksimum boyuta sahip bir ICMP paketinden daha yüksek boyutlu bir paketin gönderilerek sunucunun hizmet dışı bırakılmasını sağlayan bir saldırı türüdür. IPv4 ping paketleri maksimum 65.535 byte boyutunda olabilirler. Saldırgan tarafından gönderilen paketlerin her biri maksimum 65.535 byte olacak şekilde segmentlere bölünse de sunucuda bir araya getirilmeye çalışılan paketler arabellek taşmasına neden olacak ve sunucunun hizmet reddine sebebiyet verecektir.

Buffer Overflow Saldırısı: Buffer denilen birim, verilerin bir yerden başka bir yere taşınırken geçici olarak tutuldukları bölgelerdir. Eğer verinin hacmi bu ara bellek hacmini aşarsa ara bellek taşma olayı gerçekleşir. Saldırgan bu saldırıları tipik olarak hatalı biçimlendirilmiş girdiler ile yapar ve öngörülemeyen ara bellek taşmaları sonucunda sunucu üzerinde hizmet retlerine sebebiyet verir.

IP Fragmentation Saldırısı: Her ağın işleyebileceği datagramların benzersiz bir sınırı vardı ve veri iletimi için bu datagramların parçalanmaları gerekir. Bu sınır maksimum iletim(MTU) birimi olarak adlandırılır. Eğer hedef alıcının MTU değerinden daha büyük datagramlar göndermeye çalışıyorsa, bu datagramların parçalanarak iletilmesi gerekir.

Bu saldırı tipi hem UDP hem de TCP paketleri ile yapılabilmektedir.

- UDP paketleri ile yapılan saldırıda saldırgan sunucunun MTU değerinden yüksek sahte ICMP paketleri gönderir. Bu paketler sahte olduğundan sunucu tarafından yeniden birleştirilemez ve gelen istek sayısının artması ile sunucu kaynakları tüketilerek hizmet reddine neden olur.
- TCP paketleri ile yapılan IP fragmentation saldırıları ise “TearDrop” olarak bilinir. Bu saldırılar TCP paketlerinin yeniden birleştirme mekanizmalarını hedef alarak, paketlerin sunucu üzerinde tekrar birleştirilememesini sağlar. Veri paketlerinin çakışması ile sunucu kaynakları tüketilir ve sunucu üzerinde hizmet reddine neden olur.

Land Saldırısı: Land saldırısı, saldırganın bir TCP-SYN paketinin kaynak ve hedef bilgilerini aynı olacak şekilde değiştirmesi yapılan DoS saldırısıdır. Bir sunucuya gönderilen bu TCP-SYN paketi bir sonsuz döngüye girerek TCP yığınları oluşturur ve bunun sunucunda sunucu kaynaklarının tükenmesi hizmet reddine neden olur.

4.2 Geleneksel Bilgisayar Ağlarında DoS Saldırı Tespit Yöntemleri

Ağ güvenliği hem geleneksel bilgisayar ağlarının hem de yazılım tanımlı ağların vazgeçilmez bir parçasıdır. Geleneksel bilgisayar ağlarında ağ güvenliğinin sağlanabilmesi için birçok cihaz, protokol ve teknik kullanılmaktadır.

Bu bölümde DoS ataklarının tespitinde kullanılan üç ayrı yöntem incelenmiştir.

- İstatistiksel Yöntem
- Esnek Hesaplama Yöntemi
- Veri Madenciliği ve Makine Öğrenimi

İstatistiksel yöntem: İstatistiksel yöntemler, normal kalıpları ayırt etmek üzerine oluşturulurlar. Tipik olarak bir istatistiksel model yaratılır ve akan trafik için bir desen oluşturulur. Saldırı esnasında gelen paketler bu desen ile karşılaştırılarak trafiğin bir saldırı olup olmadığı tespit edilmeye çalışılır.

DoS saldırını, yasal trafikten ayırt etmek için korelasyon analizi olarak kullanılan çeşitli istatistiksel yöntemler mevcuttur. Hoque ve arkadaşları, korelasyon değerlerini toplamak için üçgen alan matrisi kullanmışlardır. Ağ trafiklerini birden çok ayrı zaman diliminde yakalayarak bir ön işleme tabi tuttular ve entropi için üç ayrı eşik değeri oluşturdular. Bu eşik değerleri aşıldığında ise bir saldırı tespiti tanımlamışlardır. Bu yöntem, CAIDA ve TUIDS veri setlerinde iyi sonuçlar vermiştir.[15]

Özçelik ve arkadaşları, kümülatif toplam entropisi adında yeni bir DoS algılama yaklaşımı önerdiler. Bu yöntemde, algılama verimliliğini arttırmak için paket başlığının entropisine ekstra bir kodlama tekniği uygulanmıştır. Bu çalışmada uzun vadeli değişkenleri filtrelemek ve yanlış alarmların sayısını azaltabilmek için bir dalgacık transformu kullanılmıştır. Çalışma üniversite trafik ağına yapılan DoS saldırıları ile test edilmiştir. Çalışmada yalnızca paket başlık alanının entropisini kullanan çalışmaya göre algılama oranının arttığı ve yanlış pozitif alarmların azaldığı gözlemlenmiştir.[16]

Al-Mamory ve arkadaşları, DoS saldırılarının tespiti için entropi kavramının kullanıldığı bir başka yöntem geliştirdiler. Bu yöntemde gelen verileri, normal ve saldırı durumunda sınıflandırabilmek için DBSCAN algoritmasını kullandılar. Bu yöntemde kaynak ve hedef IP, port numarası, paket sayısı ve paket oranının yanı sıra ICMP, UDP ve TCP-SYN protokolleri için de paket hızı gibi değerler eşik değeri olarak alınmıştır. Bu değerlerin oluşturduğu entropi kullanılarak saldırı tanımı yapılmaktadır. Çalışma, MIT Lincoln Laboratuvarında DoS veri seti olarak kullanılan LLDoS10 üzerinde, gerçek bir ağ ortamında gerçekleştirilmiştir. [17]

Esnek Hesaplama Yöntem: DoS saldırılarının tespitinde radyal tabanlı fonksiyonlar, genetik algoritmalar ve sinir ağları gibi öğrenme modelleri de kullanılmaya başlanmıştır. Hsieh ve Chan, 2000 DARPA veri setini kullanarak paketlerin özelliklerini etkin bir şekilde tanımlayan sinir ağlarına dayalı bir DoS algılama yöntemi önermişlerdir. Bu öneri dayanıklılık paket sayısı, paket boyutu ortalaması, zaman aralığı varyansı, paket boyutu varyansı, byte sayısı, paket hızı gibi yedi özellik kullanır. Yapılan çalışmada %94'lük bir doğruluk elde edilmiştir.[18]

Phyu ve arkadaşları, K-En Yakın Komşu algoritmasına dayalı bir yöntem önermişlerdir. İlk olarak DoS saldırılarının tespitinde kullanılacak nitelikler, değerlerine bağlı olarak paketlerden çıkarılmaktadır. Paket dışarısına alınan bu değerleri karşılaştırmak için K-En Yakın Komşu algoritması kullanılır. Ayıklanan bu değerler yasal trafik verilerinden farklılık göstererek DoS saldırılarının yapısını gösterir. Bu çalışmanın DoS saldırılarının tespitinde paket içerikleri olarak zaman aralığı varyansı, paket boyutu varyansı, paket oranı, bit oranı, SYN paketi sayısı, ACK paketi sayısı, FIN sayısı değerlerini kullanır. Yapılan çalışma %95'in üzerinde doğruluk oranına sahiptir.[19]

Veri Madenciliği ve Makine Öğrenimi: Veri madenciliği, büyük veri setlerinden yararlanmak için tasarlanmıştır. İstatistiksel yöntemler işlenmiş ve yapılandırılmış verilerden çıkarım yaparken, veri madenciliği işlenmemiş ve yapılandırılmamış veriler kullanarak bir hipotez yaratmayı amaçlar.
Yadav ve arkadaşları, Stacked Auto Encoder tarafından uygulama katmanı DoS saldırısı trafiği işleme özelliğini sınıflandırarak bir model önermişlerdir. Öncelikle bu araç ile paketleri öğrenmeye başladırlar, ardından bir regresyon analiz yöntemi kullanarak sınıflandırma yapmışlardır. Bu sınıflandırma için kaynak IP, zaman damgası, zaman dilimi, URL, yanıt kodu, gönderilen byte sayısı, referans ve kullanıcı aracısı gibi özellikler öğrenilmiştir. Çalışmada DoS saldırı tespit oranı %98,99 ve ortalama yanlış pozitif alarm sayısı ise %1,27 olarak ölçülmüştür.[20]

Kato ve arkadaşları, DoS saldırılarına karşı sistemi eğitmek ve saldırıların tespit analizleri için radyal temel işlev çekirdeği ile birlikte bir makine öğrenme tekniği olan destek vektör makinesi tekniklerini kullanarak bir yöntem önermişlerdir. Çalışmada DoS saldırılarının tespiti için Kaynak ve hedef IP, paketler arasındaki saniye cinsinden zaman aralığı ve byte cinsinden paket boyutu gibi özellikler kullanılmıştır.

Farklı veri setlerine yapılan birçok çalışmada %85'in üzerinde doğruluk oranı ölçülmüş, en yüksek doğruluk oranı ise %98,7 olarak ölçülmüştür.[21]

4.3 YTA'da DoS Saldırılarının Tespit Yöntemleri

Chen ve arkadaşları, ürüttükleri bir çalışmada YTA'da kullanılacak olan saldırı tespit yöntemlerinin kontrolcü üzerinde bulundurulması gerektiği ancak bu yöntemin kontrolcü üzerinde kaynak kullanımı arttırdığı ve bir tek nokta sorunu oluşturacağından hesaplamaların dağınık hale getirilmesinin doğru olacağını önce sürmüşler. Yaptıkları çalışmada her bir YTA anahtarını sinir sisteminde bulunan bir nörona benzetmişlerdir. Yapılan SYN flood, UDP flood, ECMP flood saldırılarının tespiti için ise sırası ile SYN_ACK paketlerinin diğer paketlere oranı, UDP paketlerinin diğer paketlere oranı ve ICMP paketlerinin diğer paketlere oranı yöntemlerini izlemişlerdir. Chen ve arkadaşları, çalışmaları sonucunda SYN flood saldırısı için %94.8, UDP flood saldırısı için ise %93.1, ICMP flood saldırısı için %96.3'lik başarı oranları gözlemlenmiştir.[22]

Y. Qian ve arkadaşları, çalışmalarında bir kara liste yöntemi izlemişlerdir. Geliştirdikleri yöntemde kontrolcü kendisine gelen her paket için "Packet_In" mesajlarını inceleyerek kaynağın MAC ve IP adresine göre saniyede gönderdiği paket sayısı ve saniyedeki byte miktarını inceliyordu. İncelediği bu değerler kontrolcü tarafında belirlenen eşik değerlerini geçiyor ise kontrolcü bu kaynaktan gelen paketlerin alınmaması için her bir YTA anahtar üzerine bu kaynak için blok kuralları yazılıyor. Fakat bu çalışma ile DoS saldırılarının tepe noktası 900'den 100'e düşürülse de kullanıcılar için bir bant genişliği kazanımı görülmez.[23]

Cui ve arkadaşları, tarafından yürütülen çalışmada ise saldırı tespiti iki aşamadan oluşmaktaydı. Kontrolcüye gelen her bir "Packet_In" mesajının sayısı tutuluyor ve bu sayılarda beklenmedik bir artış olması halinde exact-STORM adlı algoritma ile paketlerde anomali yaşanıp yaşanmadığı kontrol ediliyordu. Algoritmanın çalışma prensibi akış başına toplam paket sayısı, akış başına toplam byte miktarı, akışın tabloda kalma süresi, akış başına saniyedeki paket sayısı ve akış başına saniyedeki byte miktarı olarak tanımlanmıştır. Cui ve arkadaşları bu çalışmada yapılan diğer çalışmalardan ayrı farklı olarak her bir YTA anahtarını sınıflandırarak saldırının yolunu da tespit etmişlerdir.[24]

Dao ve arkadaşları, yaptıkları çalışmada Y. Qian ve arkadaşlarının çalışmasına benzer bir şekilde kara listeleri kullanmışlardır. Bu çalışmada saldırı öncesinde güvenilir bir kaynağın normal durumlarda ne kadar istek paketi gönderdiği "Packet_In" istek paketlerinin ortalaması alınarak hesaplanmış ve hesaplanan bu değer için *k1* ve *k2* limit aralıkları belirlenmiştir. DoS saldırılarında, saldırganlar *k2* değerinin üzerinde istek mesajları göndermeye başlamaktadır. Saldırı esnasında bu değerin üzerinde kalan akışlar, akış tablosundan silinmektedir ve kaynak için blok kayıtları oluşturulmaktadır.

Wang ve arkadaşları SECOD sistemi adı altında bir kara liste uygulaması olarak ortaya koymuşlardır. Fakat SECOD kara liste uygulamasını farklı katmanlarda uygulamıştır. "Packet_In" istek mesajı sayılarını temel alarak anahtar, anahtar ara yüzü ve IP adresi olarak üç farklı eşik değeri uygulanmıştır. Anahtar ara yüzünde tutulan "Packet_In" istek sayıları eşik değerini aştığında anahtar üzerinde bulunan akış kuralları ile eşleşmeyen paketleri düşürülmeye başlanır. Eğer saldırı bu aksiyona rağmen hala devam ediyorsa kontrolcü saldırının yapıldığı anahtar üzerinden gelen istekleri göz ardı etmeye başlanır. Bu çalışma sonucunda %99,72'lik bir veri iletimi başarı oranı yakalanmıştır.[25]

5. SONUÇ

Geleneksel bilgisayar ağlarının ve YTA'ları hedef alan birçok saldırı tipi mevcuttur. Fakat bu saldırılar arasından en çok tercih edileni ve en tehlikelilerinden biri de DoS saldırılarıdır. Geleneksel bilgisayar ağlarını ve YTA'ları hedef alan DoS saldırıları ile kötü niyetli kişiler ağ hizmetlerini servis dışı bırakmayı amaçlamaktadır. Bu yapılan DoS saldırıları, geleneksel ağ mimarilerinde kontrol katmanını da kendi üzerinde bulunduran ağ cihazlarına doğrudan yapılırken, YTA mimarisinde ise genellikle YTA'da kontrol katmanını üzerinde bulunduran kontrolcülere yapılır. Kontrolcüyü hedef alan DoS saldırıları kontrolcünün kaynaklarını tüketerek kontrolcünün devre dışı kalmasını sağlamayı amaçlar. YTA'larda bulunan kontrolcüler ağın beyni olduklarından DoS saldırılarının hedefi haline gelmiştir ve eğer bir YTA'da kontrolcüye gerçekleştirilen bir DoS saldırısı başarılı olur ise YTA tamamen servis dışı kalabilir. Bu nedenle geleneksel bilgisayar ağları ile yazılım tanımlı bilgisayar ağları arasında farklı güvenlik zafiyetlerinin olduğu söylenebilir.

KAYNAKLAR

Bergel Ç. (2004). *Bilgisayar Ağları ve Güvenlik* (Yüksek lisans tezi). Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Sakarya.

Özer E. (2015). *Derin Palet Analizi İle DDoS Ataklarının Tespiti ve DLP Uygulaması* (Yüksek lisans tezi). Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Sakarya.

Aytaç T. (2020). DDoS Saldırılarının Tespit Edilmesinde Makine Öğrenimi Yöntemlerinin Uygulanması (Yüksek lisans tezi). İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Yavuz A. (2020). *Yazılım Tanımlı Ağların ve Geleneksel Bilgisyar Ağlarının Güvenlik Karşılaştırması* (Yüksek lisans tezi). Trakya Üniversitesi, Fen Bilimleri Enstitüsü, Edirne.

Çil A.E. (2021). *DDoS Saldırılarının Tespiti ve Sınıflandırılması İçin Derin Öğrenme Modeli* (Yüksek lisans tezi). Marmara Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Makori D.O. (2018). *Machine Learning Based DDoS Attack Detection For Software-Defined Networks* (Yüksek lisan tezi). Sakarya University, Master of Science In Computer and Information Engineering Department, Sakarya.

Gözütok A. (2018). *Data Plane-Based Defense System Against DDoS Attacks For Software Defined Networks* (Master lisans tezi). Middle East Technical University, Master of Science In Electrical and Electronics Engineering Department, Istanbul.

Türker C. (2019). *Management of The Internet Of Things(IoT) Wıth Software-Defined Networking* (Yüksek lisan tezi). Işık University, Master of Science In Computer Engineering Department, Istanbul.

Dayal, N., Maity, P., Srivastava S., & Khondoker, R. (2017). Research Trends in Security and DDoS in SDN.,*Security and Communication Networks*, 6388-6404.

Cicioğlu, M., & Çalhan, A. (2017). Yazılım Tanımlı Ağlar - YTA., *Karaelmas Fen ve Mühendislik Dergisi*, 690-693.

Polat, H., Polat, O., & Cetin, A. (2020). Detecting DDoS Attacks in Software-Defined Networks Through Feature Selection Methods And Mechine Learning Models., *Sustainability 2020, 12, 1035*, 3-5.

Kia, M. (2015). *Early Detection And Mitigation of DDoS Attacks In Software Defined Network*(Yüksek lisan tezi). Ryerson University, Master of Science In Computer Engineering Department, Toronto.

Aliyu, L.A., Bull, P., & Abdallah A. (2017) Investigating The Security Aspect of Software Defined Networking.,*Birmingham University*, 6388-6404.

Vizvary M., & Vykopal J. (2014). Future of DDoS Attacks Mitigation in Software Defined Networks, *Conference Paper in Lecture Notes in Computer Science*, Masaryk University, Czech Republic.

Göksel N. (2019). *Yazılım Tanımlı Ağlarda Hizmet Dışı Bırakma Saldırılarının Tespiti ve Engellenmesi* (Yüksek lisans tezi). Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

Hoque, N., Bhattacharyya, D., & Kalita, J. (2016). Denial of Service Attac Detection Using Multivariate Correlation Analysis. *Second International Conference on Information and Communication Technology for Competitive Strategies*, ACM.

Özçelik, İ., & Brooks, R.R. (2016). Cusum-Entropy: An Efficient Method for DDoS Atack Detection. *Smart Grid Congress(ICSG)*, IEEE.

Al-Mamory, S.O., & Ali, Z.M. (2015). Cusum-Entropy: Using DBSCAN Clustering Algorithm in Detection DDoS attack. *Journal of Babylon University*, 1412-1424

Hsieh, C., & Chan, T. (2016). Detection DDoS attacks based on neural-network using Apache Spark. *Applied System Innovation Conference(ICASI)*

Phyu, T., & Oo, T.T. (2014). Analysis of DDoS Detection System Based on Anomaly Detection System. *International Conference on Advances in Engineering and Technology(ICAET)*

Yadav, S., & Subramanian, S. (2016). Detection of Application Layer DDoS Attack by Feature Learning Using Stacked AutoEncoder. *Internation Conference on Computational Techniques in Information and Communication Technologies(ICCTICT)*

Kato, K., & Klyuev, V. (2014). Large-Sclae Network Packet Analysis for Intelligent DDoS Attack Detection Development. *Internet Technology and Secured Transactions(ICITST) on IEEE*

Chen, X., & Yu, S. (2016). A Collaborative Intrusion Detection System Against DDoS for SDN. *Information and System on IEICE*, 2395-2399

Qian, Y., You, W. & Qian, K. (2016). OpenFlow Flow Table Overflow Attacks and Countermeasures. *European Conference on Networks and Communications(EuCNC)*

Cui, Y., Yan, L., Li, S., Xing, H., Pan, W., & Zhu, J., Zheng, X., (2016).SD-Anti-DDoS: Fast and Efficient DDoS Defense in SDN. *Journal of Network and Computer Applications*, 65-79

Dao, N.N., Kim, J., Park, M., & Cho, S. (2016). Adaptive Suspicious Prevention for Defending DoS Attacks in SDN Based Convergent Networks. *Journal of PIoS one*, 27-35

Wang, S., Chandrasekharan, S., Gomez, K., & Kandeepan, S., (2018). SECOD: SDN sEcure Control and Data Plane Algorithm for Detecting and Defending against DoS Attacks. *Northbound Networks, Melbourne*