
GAZİ ÜNİVERSİTESİ
Gazi
1926

YAZILIM TANIMLI AĞLAR İÇİN SİBER TEHDİT İSTİHBARATI DESTEKLİ SAVUNMA SİSTEMİ GELİŞTİRME

Özgür YÜREKTEN

DOKTORA TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

EKİM 2020

Özgür YÜREKTEN tarafından hazırlanan "YAZILIM TANIMLI AĞLAR İÇİN SİBER TEHDİT İSTİHBARATI DESTEKLİ SAVUNMA SİSTEMİ GELİŞTİRME" adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalında DOKTORA TEZİ olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Mehmet DEMİRCİ

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum. ...……..……..

Başkan: Prof. Dr. Şeref SAĞIROĞLU

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum. ……..……

Üye: Prof. Dr. M. Ali AKCAYOL

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum. ...…….……...

Üye: Prof. Dr. Ali Aydın SELÇUK

Bilgisayar Mühendisliği Ana Bilim Dalı, TOBB ETÜ

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum. ...……

Üye: Prof. Dr. Ertan ONUR

Bilgisayar Mühendisliği Ana Bilim Dalı, Orta Doğu Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum. ...……

Tez Savunma Tarihi: 14/10/2020

Jüri tarafından kabul edilen bu tezin Doktora Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

……..…….

Prof. Dr. Cevriye GENCER

Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Özgür YÜREKTEN

14/10/2020

YAZILIM TANIMLI AĞLAR İÇİN SİBER TEHDİT İSTİHBARATI DESTEKLİ SAVUNMA SİSTEMİ GELİŞTİRME

(Doktora Tezi)

Özgür YÜREKTEN

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Ekim 2020

ÖZET

Sayısı ve çeşitliliği giderek artan siber tehditlere karşı koyabilmek için çevik ve proaktif ağ savunma yaklaşımlarına ihtiyaç duyulmaktadır. Bu nedenle siber tehdit istihbaratı (STİ) bilgilerinin hızlı şekilde paylaşılması ve bu bilgiler dikkate alınarak ağların yapılandırılması büyük önem taşımaktadır. Fakat klasik ağların yönetimi ve yapılandırılması oldukça karmaşık, zor ve zaman alıcıdır. Bu problemlerin çözümü için yazılım tanımlı ağlar (YTA), ağ fonksiyonu sanallaştırma (AFS) ve servis fonksiyon zincirleri (SFZ) gibi yeni ağ teknolojilerinin kullanılması gerekmektedir. Literatürde belirli siber saldırı türleri için YTA temelli siber savunma önerileri yer almaktadır. Buna karşılık STİ verilerini işleyerek ağ seviyesinde savunma servislerini otomatikleştiren ve yeni ağ teknolojilerini kullanan bütüncül güvenlik mimarisi önerisi bulunmamaktadır. Bu tez çalışması kapsamında literatürde ilk kez STİ verilerinden faydalanarak yazılım tanımlı ağlarda otomatik ağ seviyesi savunma servislerinin seçilmesini, oluşturulmasını ve SFZ olarak yönetimini sağlayan bir sistem önerisi sunulmuştur. Önerilen sistem için dört ağ savunma servisi (kara liste, karantina, tuzak sisteme gönderme, çoklu rota) içeren bir prototip geliştirilmiş ve STİ verilerinin analizi sonucunda belirlenen saldırgan modelleri ve saldırı senaryoları kullanılarak önerilen sistem sınanmıştır. Yapılan değerlendirmelerde, önerilen sistemin siber tehditlere karşı savunmada en fazla 6,99 MBit/sn büyüklüğünde bant genişliği kullandığı ve literatürdeki çalışmalara benzer veya bu çalışmalardan daha iyi sonuçlar elde edildiği görülmüştür. Önerilen sistemin, saldırılara karşı (DDoS, oltalama, zararlı yazılım vb.) savunma yaparken kara listede yer alan 8900 IP adresini 14 saniyede ağda yapılandırabildiği, aynı anda gerçekleştirilen 400 saldırıyı 0,4 saniyede tuzak sistemlere yönlendirebildiği görülmüştür. Gerçekleştirilen diğer bir testte de çoklu rota savunma servisinin dinleme saldırıları ile elde edilebilecek bilgiyi %4,0'ün altına indirebildiği ölçülmüştür. Değerlendirme sonuçları, önerilen sistemin yeni nesil ağlarda siber tehditlere karşı genişletilebilir, çevik ve proaktif ağ savunması sağlamakta başarılı ve ağ performansı açısından etkin olduğunu göstermiştir.

Bilim Kodu	:	92403
Anahtar Kelimeler	:	Siber güvenlik, ağ güvenliği, siber tehdit istihbaratı, STİ, yazılım tanımlı ağlar, YTA, ağ fonksiyonu sanallaştırma, AFS, servis fonksiyon zinciri, SFZ
Sayfa Adedi	:	182
Danışman	:	Dr. Öğr. Üyesi Mehmet DEMİRCİ

DEVELOPMENT OF CYBER THREAT INTELLIGENCE ASSISTED DEFENSE SYSTEM FOR SOFTWARE DEFINED NETWORKS

(Ph. D. Thesis)

Özgür YÜREKTEN

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

October 2020

ABSTRACT

Agile and proactive network defense approaches are needed in order to withstand cyber threats that have risen in number and variety; therefore, it is of great importance to share cyber threat intelligence (CTI) information and to make network security configurations faster. However, the management and configuration of traditional networks are very complex, difficult and time consuming. The use of new network technologies, such as software-defined networking (SDN), network function virtualization (NFV) and service function chaining (SFC) is required to deal with these problems. There are SDN-based cyber defense proposals for specific types of cyber-attacks in the literature. However, there is no holistic security architecture proposal for automating network-level security functions using CTI data and the new network technologies mentioned above. In this study, we propose a novel SDN defense system that uses CTI data and enables the selection, creation, and management of automatic network-level defense services. We have implemented a prototype of the proposed system and four network-level defense services (blacklist, quarantine, send-to-decoy, and multipath), and evaluated the proposed system using common cyber threat models and attack scenarios determined by analyzing CTI data. In the evaluations, the proposed system uses at most 6.99 Mbit/s of bandwidth, similar to or better than the studies in the literature. The proposed system configures 8900 blacklist IP addresses into the SDN-enabled test network in 14 seconds, and redirects 400 concurrent attack flows to decoy systems in 0.4 seconds to deal with common cyber-attacks, such as DDoS, phishing, malware, or drive-by download attacks. Moreover, the proposed system with multipath defense service reduces sniffed flow information to under 4.0% against eavesdropping attacks. The evaluations show that the proposed system is successful in providing extensible, agile and proactive network security against cyber threats in modern networks and is effective in terms of network performance.

Science Code	:	92403
Key Words	:	Cyber security, network security, cyber threat intelligence, CTI, software defined networking, SDN, network function virtualization, NFV, service function chaining, SFC
Page Number	:	182
Supervisor	:	Assist. Prof. Dr. Mehmet DEMİRCİ

TEŞEKKÜR

Tez çalışmam süresince bana sürekli yol gösteren ve destek olan danışmanım Dr. Öğr. Üyesi Mehmet DEMİRCİ'ye, tez izleme komitemde bulunarak çalışmalarıma katkı sağlayan Prof. Dr. Şeref SAĞIROĞLU'na ve Prof. Dr. Ertan ONUR'a, hiçbir zaman desteklerini esirgemeyen sevgili eşim Zeynep YÜREKTEN'e, kızım Havva Sena YÜREKTEN'e, oğlum Ali YÜREKTEN'e ve bu günlere gelmemde büyük emeği olan değerli aileme çok teşekkür ederim. Ayrıca siber güvenlik alanında yaptığım doktora çalışmalarımı desteklediği için TÜBİTAK BİLGEM Siber Güvenlik Enstitüsü'ne şükranlarımı sunarım.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil **Sayfa**

Şekil **Sayfa**

SİMGELER VE KISALTMALAR

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

Simgeler	**Açıklamalar**
çr	Çoklu rota
dp	YTA anahtarını tanımlayan veri yolu
Gbit	Giga bit
Kbit	Kilo bit
maks	En çok
Mbit	Mega bit
min	En az
ms	Milisaniye
nsh_{spi}	Ağ servis başlığı (NSH) servis yolu indeksi
nsh_{si}	NSH servis indeksi
ort	Ortalama
p	Anahtar bağlantı noktası (port)
per	Periyot
sn	Saniye

Kısaltmalar	**Açıklamalar**
5G	Fifth Generation Technology **(**Beşinci Nesil Teknoloji)
ACYDS	Adaptive Cyber Deception System (Uyarlanabilir Siber Aldatma Sistemi)
AEK	Ağ Erişim Kontrolü
AFS	Ağ Fonksiyonu Sanallaştırma
ANAH	Yazılım Tanımlı Ağ Anahtarı
APT	Advanced Persistent Threat (Gelişmiş Kalıcı Tehdit)

Kısaltmalar	Açıklamalar
ARP	Address Resolution Protocol (Adres Çözümleme Protokolü)
ATT&CK	Adversarial Tactics, Techniques & Common Knowledge (Kötücül Taktik, Teknik ve Genel Bilgi)
ATY	Anomali Tespit Yöneticisi
BASE	BGP-based Anti-Spoofing Extension (BGP tabanlı Sahtecilikle Mücadele Eklentisi)
BGP	Border Gateway Protocol (Sınır Geçit Protokolü)
BT	Bilgi Teknolojileri
BYOD	Bring Your Own Device (Kendi Cihazını Getir)
CAPEC	Common Attack Pattern Enumeration & Classification (Yaygın Saldırı Örüntüsü Sayımı ve Sınıflandırması)
CIF	Collective Intelligence Framework (Kolektif İstihbarat Çerçevesi)
CINDAM	Customized Information Networks for Deception and Attack Mitigation (Aldatma ve Saldırı Azaltma için Özelleştirilmiş Bilgi Ağları)
CSRF	Cross-Site Request Forgery (Siteler Arası İstek Sahteciliği)
CTI	Cyber Threat Intelligence (Siber Tehdit İstihbaratı)
CVE	Common Vulnerabilities and Exposures (Genel Güvenlik Açıkları ve Riskler)
CWE	Common Weakness Enumeration (Genel Zayıflık Sayımı)
CYBOX	Cyber Observable Expression (Siber Gözlenebilir İfade Dili)
DaMask	DDoS Attack Mitigation Architecture Using Software Defined Networking (Yazılım Tanımlı Ağları Kullanan DDoS Saldırı Etkisi Azaltma Mimarisi)
DBA	DDoS Blocking Application (DDoS Engelleme Uygulaması)

Kısaltmalar	Açıklamalar
DDoS	Distributed Denial of Service (Dağıtık Hizmet Engelleme)
DHC	Double Hopping Communication (Çift Atlamalı İletişim)
DHCP	Dynamic Host Configuration Protocol (Dinamik İstemci Yapılandırma Protokolü)
DNS	Domain Name System (Alan Adı Sistemi)
DoS	Denial of Service (Hizmet Engelleme)
DPDK	Data Plane Development Kit (Veri Katmanı Geliştirme Kiti)
EAP	Extensible Authentication Protocol (Genişletilebilir Kimlik Doğrulama Protokolü)
ETH	Ethernet frame (Ethernet çerçevesi)
FICUR	Traffic Pattern Based Solution to ARP Related Threats (ARP İle İlgili Tehditlere Trafik Örüntüsü Tabanlı Çözüm)
FPH	Fingerprint Hopping Method (Parmak İzi Atlama Yöntemi)
FTPS	File Transfer Protocol Secure (Güvenli Dosya Aktarım Protokolü)
GRE	Generic Routing Encapsulation (Genel Yönlendirme Sarmalama)
HHS	Hareketli Hedef Savunması
HTTP	Hyper Text Transfer Protocol (Üst Metin Aktarım Protokolü)
HTTPS	Hyper Text Transfer Protocol Secure (Güvenli Üst Metin Aktarım Protokolü)
ICMP	Internet Control Message Protocol (İnternet Kontrol Mesajı Protokolü)
IODEF	Incident Object Description Exchange Format (Olay Nesne Açıklama Paylaşım Biçimi)

Kısaltmalar	Açıklamalar
IoT	Internet of Things (Nesnelerin İnterneti)
IP	Internet Protocol (İnternet Protokolü)
İSS	İnternet Servis Sağlayıcı
İST	İstemci
K&K	Komuta Kontrol
KNTL	YTA Kontrolcüsü
KVM	Kernel Virtual Machine (Çekirdek Sanal Makinesi)
LFA	Link Flooding Attacks (Bağlantı Taşkın Saldırıları)
LLDP	Link Layer Discovery Protocol (Bağlantı Katmanı Keşif Protokolü)
MAC	Media Access Control (Ortam Erişim Kontrolü)
MARS	Malware Analysis Architecture (Zararlı Yazılım Analiz Mimarisi)
MNOS	Mimic Network Operating System (Taklit Ağ İşletim Sistemi)
MTD	Moving Target Defense (Hareketli Hedef Savunması)
NAT	Network Address Translation (Ağ Adresi Dönüştürme)
NDP	Neighbor Discovery Protocol (Komşu Keşif Protokolü)
NFGA	Network Flow Guard for ARP (ARP için Ağ Akış Koruması)
NFV	Network Function Virtualization (Ağ Fonksiyonu Sanallaştırma)
NSH	Network Service Header (Ağ Servis Başlığı)
OF-RHM	OpenFlow Random Host Mutation (OpenFlow Rastgele İstemci Mutasyonu)
OPENIOC	Open Indicators of Compromise (Açık Tehdit Göstergeleri)
OPERETTA	OpenFlow Based Remedy to TCP SYN Flood Attacks (TCP SYN Taşkın Saldırılarına OpenFlow Tabanlı Çözüm)

Kısaltmalar	Açıklamalar
OT	Operational Technology (Operasyonel Teknoloji)
PLC	Programmable Logic Controller (Programlanabilir Mantık Kontrolcüsü)
POF	Protocol Oblivious Forwarding (Protokol Habersiz Yönlendirme)
PPA	Phishing Prevention Algorithm (Oltalama Önleme Algoritması)
RADAR	Reinforcing Anti-DDoS Actions in Real-time (Anti-DDoS Eylemlerini Gerçek Zamanlı Olarak Güçlendirme)
RDAM	Random Domain Name and Address Mutation (Rastgele Alan Adı ve Adres Mutasyonu)
RDS	Reconnaissance Deception System (Keşif Aldatma Sistemi)
RRM	Random Route Mutation (Rastgele Rota Mutasyonu)
SARP	Secure Address Resolution Protocol (Güvenli Adres Çözümleme Protokolü)
SAVI	Source Address Validation Improvement (Kaynak Adres Doğrulama İyileştirme)
SCADA	Supervisory Control and Data Acquisition Systems (Denetleyici Kontrol ve Veri Toplama Sistemleri)
SDN	Software Defined Networking (Yazılım Tanımlı Ağlar)
SDNSOC	SDN Security Operation Center (YTA Güvenlik Operasyon Merkezi)
SDON	Software Defined Optical Networking (Yazılım Tanımlı Optik Ağlar)
SEFA	Software Defined Filtering Architecture (Yazılım Tanımlı Filtreleme Mimarisi)
SEHT	Self-adaptive Endpoint Hopping Technique (Kendinden Uyarlamalı Uç Nokta Atlama Tekniği)

Kısaltmalar	Açıklamalar
SFC	Service Function Chaining (Servis Fonksiyon Zinciri)
SFZ	Servis Fonksiyon Zinciri
SİY	Savunma İş Birliği Yöneticisi
SMB	Server Message Block (Sunucu Mesaj Bloğu)
SMT	Satisfiability Modulo Theories (Sağlanabilirlik Modülo Teorileri)
SÖS	Saldırı Önleme Sistemi
SQL	Structured Query Language (Yapısal Sorgu Dili)
SSL	Secure Sockets Layer (Güvenli Soket Katmanı)
STIX	Structured Threat Information Expression (Yapısal Tehdit Bilgileri İfadesi)
STİ	Siber Tehdit İstihbaratı
STİ-Y	Siber Tehdit İstihbaratı Yöneticisi
STS	Saldırı Tespit Sistemi
SUN	Sunucu
SYN	Synchronize packet of TCP (TCP Senkronize Paketi)
TAP	Terminal Access Point (Terminal Erişim Noktası)
TAXII	Trusted Automated Exchange of Intelligence Information (Güvenli Otomatik İstihbarat Bilgisi Paylaşımı)
TCP	Transmission Control Protocol (İletim Kontrol Protokolü)
TIDS	Transparent Intrusion Detection System (Şeffaf Saldırı Tespit Sistemi)
TLS	Transport Layer Security (Taşıma Katmanı Güvenliği)
TTL	Time to Live (Yaşam Süresi)
TTP	Tactics, Techniques and Procedures (Taktikler, Teknikler ve Prosedürler)
UDP	User Datagram Protocol (Kullanıcı Veri Bloğu Protokolü)
URL	Uniform Resource Locator (Birörnek Kaynak Bulucu)

Kısaltmalar	Açıklamalar
USOM	Ulusal Bilgisayar Olayı Müdahale Merkezi
UYG	Yazılım Tanımlı Ağ Uygulaması
UYG-ANAH	Ağ Anahtarına Erişen YTA Uygulaması
VAVE	Virtual Source Address Validation Edge (Sanal Kaynak Adresi Doğrulama Ucu)
VLAN	Virtual Local Area Network (Sanal Yerel Alan Ağı)
VxLAN	Virtual Extensible Local Area Network (Sanal Genişleyebilir Yerel Alan Ağı)
XAI	Explainable Artificial Intelligence (Açıklanabilir Yapay Zekâ)
XSS	Cross Site Scripting (Siteler Arası Komut Çalıştırma)
Y-S3	YTA Temelli Siber Savunma Sistemi
YTA	Yazılım Tanımlı Ağlar
YTA-AY	YTA Akış Yöneticisi
YTA-SSY	YTA Siber Savunma Yöneticisi
YTA-TY	YTA Topoloji Yöneticisi
YTD	Yazılım Tanımlı Düzenleyici

1. GİRİŞ

İnternet üzerinden sürekli bağlantı sağlayan modern teknolojiler, insanların iletişim alışkanlıklarını değiştirmektedir. Bu değişim aynı zamanda yeni teknolojilere olan talebi de sürekli olarak artırmaktadır. Bulut bilişim, akıllı cihazlar ve nesnelerin interneti (Internet of Things, IoT) gibi yeni teknolojiler, her alanda kullanılmaya başlandıkça da herkesi etkileyen yeni siber güvenlik risklerini ortaya çıkarmaktadır. Bu durum siber güvenlik konusunda herkesin farkındalığının artırmasını zorunlu kılmaktadır. İnternet kullanıcıları aldıkları dijital hizmetlerde; yüksek bağlantı hızı, kullanılabilirlik, erişilebilirlik ve mobil cihaz desteği gibi özelliklerin yanı sıra artık mahremiyetin ve güvenliğin sağlanmasını da talep etmeye başlamıştır.

Her gün yeni siber tehditler ve saldırı vektörleri ortaya çıkmakta, bu tehditlere karşı proaktif siber savunma yaklaşımları da daha önemli hale gelmektedir [1]. 2019'un ilk çeyreğinde, 65 milyondan fazla yeni zararlı yazılım örneği tespit edilmiş ve toplam zararlı yazılım sayısı yaklaşık 1 milyara ulaşmıştır [2]. Ayrıca, 2019'da 46 binden fazla kripto-fidye yazılımı varyasyonu ve 22 yeni kripto-fidye yazılımı ailesi tespit edilmiştir [3]. Zararlı yazılımlara ek olarak, dağıtık hizmet engelleme (distributed denial of service, DDoS), dinleme, web uygulaması saldırıları ve gelişmiş kalıcı tehditler (advanced persistent threat, APT) gibi tehditler de artmıştır. Örneğin, IoT cihazlarını kullanan Mirai adlı büyük botnet, Ağustos 2016'da tespit edilmiş [4] ve o zamandan bugüne önemli ölçüde büyümüştür.

Organizasyonlarda tüm siber tehdit türlerini el ile tanımlamak, analiz etmek ve önlemek oldukça zordur. Bu nedenle, siber tehdit bilgilerinin paylaşılması organizasyonlar için büyük önem taşımaktadır. Birden fazla kaynaktan toplanan ve doğrulanan siber tehdit bilgileri, siber tehdit istihbaratı (STİ / cyber threat intelligence, CTI) [1, 5] verisi olarak organizasyonlar arasında paylaşılabilmektedir. STİ verilerinin paylaşımı için çeşitli veri formatları tanımlanmıştır. Bu veri formatları kullanılarak paylaşılan STİ verileri siber tehditlerin tespit edilmesi, önlenmesi ve azaltılması için kullanılabilmektedir.

Geleneksel ağların yönetimi ve yapılandırılması oldukça karmaşık, zorlayıcı ve zaman alıcıdır. Geleneksel ağlarda, siber tehditleri önlemek, tespit etmek ve saldırıların etkisini azaltmak için farklı güvenlik çözümleri (güvenlik duvarı, derin paket denetimi, saldırı önleme sistemi vb.) kullanılmaktadır. Bu çözümlerin birçoğu özel donanım cihazları olarak

sunulmakta ve ağda konuşlandırıldıkları konumlar dikkatle belirlenmektedir. Bu tür ağlarda trafik yönlendirmesi, bal küpü kullanımı ve ağın ayrılması gibi ağ savunma yapılandırmaları maliyetli olabilmekte ve ağlarda kararsız durumlara yol açabilmektedir [6]. Yeni bir ağ teknolojisi olarak yazılım tanımlı ağlar (YTA / Software Defined Networking, SDN), ağlardaki kontrol ve veri katmanını ayırma ilkesine dayanmakta ve son derece esnek ağlar sağlayarak ağ yönetimini basitleştirmektedir [7]. YTA, OpenFlow [8] gibi protokoller aracılığıyla ağ programlanabilirliği sağlamakta, özel ağ cihazlarına olan ihtiyacı azaltmakta ve ağ cihazlarını değiştirmeden yeni algoritmalar ve protokoller geliştirilmesine olanak sunmaktadır [7]. Bu özelliklerinden dolayı YTA, yeni güvenlik çözümleri geliştirmek ve test etmek için ideal bir platform sağlamaktadır. YTA kontrolcüsünün bütüncül ağ görünümünü yönetmesi sayesinde gelişmiş saldırı tespit yeteneğine sahip olunabilmektedir. Ayrıca trafik yönlendirme, kaynak tahsisinde değişiklik yapma, paketleri filtreleme veya anahtarlar üzerindeki akış kurallarını kullanarak paket başlıklarını değiştirme gibi yeteneklerle saldırıları önleme ve saldırının etkisini azaltabilmektedir.

Gelişmekte olan diğer bir ağ teknolojisi de ağ fonksiyonu sanallaştırma (AFS / network function virtualization, NFV) [9] teknolojisidir. Bu teknoloji ile organizasyonların ağ cihazları ve ağ altyapıları (güvenlik duvarı, yük dengeleyici, anahtar, yönlendirici, VPN sunucusu vb.) sanal ortamlara taşınmakta ve bu ortamlar üzerinden ağ yönetimi yapılmaktadır. Ağ sanallaştırma mimarisine taşınan fonksiyonlar sanal ağ fonksiyonu olarak çalışmaktadır. AFS, organizasyonların yatırım ve işletim maliyetlerini azaltmakta, ağ altyapılarının yönetimini ve yapılandırmasını kolaylaştırmakta ve gelişmiş ağ yetenekleri sunmaktadır. Bu kazanımların yanı sıra AFS ayrıca; ağ servislerinin performanslı şekilde çalışabilmesini, farklı sanallaştırma altyapılarına taşınabilmesini, esnek şekilde genişletilebilmesini, hata durumunda tekrar başlatılabilmesini ve sürekliliğinin garanti edilmesini sağlayabilmektedir.

Servis fonksiyon zinciri (SFZ / service function chaining, SFC) [10], belirlenen politikalara göre paketlerin veya akışların farklı sanal ağ fonksiyonları üzerinden sıralı şekilde yönlendirilmesini sağlayan teknolojilerden biridir. Kullanılacak servis fonksiyonları ve sıraları belirlendikten sonra bu zincir için gerekli kurallar oluşturulmakta ve ilgili trafiğin bu zincir üzerinden geçmesi sağlanmaktadır. Servis fonksiyon zincirleri YTA kontrolcüsü tarafından önceden veya anlık olarak oluşturulabilmektedir. AFS ve SFZ teknolojileri birlikte kullanılarak daha dinamik ağ yönetimi ve savunma çözümleri geliştirilebilmektedir.

Tezin önemi

Siber tehdit istihbaratı verileri farklı kaynaklardan elde edilebilmektedir. STİ verilerini elde etmek için kullanılabilecek ilk kaynak, organizasyon içi bilgilerin ve yetkinliklerin kullanılmasıdır. Personel tarafından güvenlik cihazlarından toplanan olay kayıtları analiz edilir. Ancak bu süreç zaman alıcıdır, yoğun emek gerektirir ve potansiyel olarak hataya açıktır. Organizasyonlar STİ verilerini ticari veya açık topluluk kaynaklarından alabileceği gibi farklı devlet kurumları, özel şirketler ve ortak kuruluşlar arasında oluşturulmuş paylaşım ağlarından da alabilmektedir. Fakat mevcut durumda, farklı kaynaklardan elde edilen STİ verileri çoğunlukla tehdit göstergelerini alabilmek için kullanılmakta, saldırılara karşı otomatik ağ seviyesi savunma yöntemlerini paylaşmak için kullanılmamaktadır.

STİ verilerinin paylaşımı için Yapısal Tehdit Bilgisi İfadesi (Structured Threat Information Expression, STIX) [11], Olay Nesnesi Açıklama Paylaşım Biçimi (Incident Object Description Exchange Format, IODEF) [12], Açık Tehdit Göstergesi (Open Indicators of Compromise, OpenIoC) [13] gibi çeşitli STİ veri formatları tanımlanmıştır. Bu veri formatları kullanılarak saldırı modelleri ve örüntüleri tanımlanabilmektedir. Bu bilgiler, saldırıların nasıl yürütüldüğünü ve saldırı vektörlerini soyut olarak açıklamaktadır [14]. Ortak Saldırı Örüntüsü Sıralama ve Sınıflandırma (Common Attack Pattern Enumeration & Classification, CAPEC) [15], saldırıları açıklamak ve tehditler hakkında bilgi paylaşmak için oluşturulmuş yaygın bir saldırı örüntüleri kataloğudur. Bu katalog içinde saldırıların etkisini azaltmak için yapılabilecekler düz metin olarak sunulmaktadır. MITRE Kötücül Taktikler, Teknikler ve Ortak Bilgi (Adversarial Tactics, Techniques & Common Knowledge, ATT&CK) [16] çevrim içi kütüphanesi, kurumsal ağlara karşı gerçekleştirilen kalıcı tehditlerin kullandıkları ortak taktikleri, teknikleri ve prosedürleri listeleyerek üst seviyeden saldırı tespit ve etki azaltma yöntemleri tarif etmektedir. Bu veri formatlarının ve veri kataloglarının hiçbirinde siber saldırılar karşısında ağ seviyesinde savunma yapmak için kullanılacak otomatik hareket tarzları tanımlanamamaktadır. Otomatik olarak ağ savunma servislerinin oluşturulabilmesini sağlayacak şekilde mevcut STİ veri formatlarındaki bilgilerin genişletilebileceği değerlendirilmektedir.

YTA, AFS ve SFZ teknolojileri birlikte kullanılarak; tehditlere karşı daha etkili ağ savunma servisleri geliştirilebilmekte, organizasyonların ağ seviyesinde otomatik savunması yapılabilmekte ve iyileştirilmiş ağ güvenlik altyapıları oluşturulabilmektedir. Literatürde,

belirli saldırılar için çeşitli YTA temelli siber savunma önerileri yer almaktadır fakat STİ verilerini kullanan YTA temelli siber savunma konusunda çok az çalışma bulunmaktadır. STİ verilerini işleyerek ağ seviyesinde güvenlik işlevlerini otomatikleştiren ve YTA, AFS ve SFZ gibi yeni ağ teknolojilerini kullanan hiçbir güvenlik mimarisi, modeli veya çerçevesi literatürde yer almamaktadır.

Tezin amacı ve kapsamı

Bu tezin genel amacı, STİ verilerini işleyerek çeşitli savunma politikalarını otomatik olarak uygulayabilen bir YTA temelli ağ güvenlik sisteminin tasarlanması ve geliştirilmesidir. Bu genel amaç doğrultusunda belirlenen tez araştırma soruları aşağıda listelenmiştir:

1. Modüler, genişletilebilir, iş birliği içinde çalışabilen ağ savunma sistemi, YTA mimarisi çerçevesinde nasıl tasarlanmalıdır?
2. Yazılım tanımlı ağlarda, saldırıları önlemek için ağ seviyesinde ihtiyaç duyulacak temel savunma servisleri nelerdir?
3. STİ verilerini işleyerek siber saldırılara karşı otomatik ağ seviyesi savunma servisleri nasıl oluşturulabilir?
4. Farklı saldırı türlerine karşı kullanılabilecek savunma servisleri ve servis zincirleri nasıl belirlenebilir?
5. Ağ seviyesi savunma servisleri ile sanal ağ fonksiyonlarının orkestrasyonu nasıl sağlanabilir?

Bu tez çalışmasında birinci araştırma sorusu kapsamında; STİ verilerini kullanarak YTA temelli siber savunma yapan yeni bir sistemin bileşenleri belirlenmiş, bileşenlerin görevleri ve ara yüzleri detaylandırılmış ve ağ seviyesi savunma servislerinin modüler şekilde oluşturulabileceği genişleyebilir bir yazılım mimarisi tasarlanmış ve sistem prototipi geliştirilmiştir.

İkinci araştırma sorusu kapsamında; öncelikle literatür taraması yapılmış ve sonrasında YTA temelli ağ savunması öneren çalışmalar siber tehdit türlerine göre analiz edilerek yazılım tanımlı ağlarda uygulanabilecek temel ağ savunma servisleri belirlenmiştir.

Üçüncü araştırma sorusu kapsamında; STİ verileri ile paylaşılan genel saldırı modelleri belirlenmiş, bu tehdit modellerinin tümüne karşı savunma yapabilecek ve birden fazla

saldırıda kullanılabilecek dört temel ağ savunma servisinin (çoklu rota, tuzak sisteme gönderme, karantina, kara liste) tasarımı yapılmış ve önerilen sistemle entegre olacak şekilde geliştirilmiştir.

Dördüncü araştırma sorusu kapsamında; otomatik ağ savunmasının yapılabilmesi için STİ verilerinin içermesi gereken otomatik ağ savunması veri modeli oluşturulmuştur. Bu veri modeli ile birlikte kurumsal varlıkları, ağ topolojisini ve mevcut ağ güvenlik politikalarını dikkate alarak uygulanacak savunma servislerini ve servis fonksiyon zincirlerini belirleyen bir yöntem belirlenmiş ve önerilen sistemde uygulanmıştır.

Beşinci araştırma sorusu kapsamında; veri katmanında farklı akışların farklı servis fonksiyonu zincirleri üzerinden geçebilmesini sağlamak için ağ servis başlığı (network service header, NSH) protokolünden faydalanılarak savunma servislerinin ve sanal ağ fonksiyonlarının dinamik olarak yönetilebilmesi için tasarım yapılmış ve geliştirilmiştir. Açık kaynaklı projelerde NSH protokolünün desteklenmesi için geliştirme yapılmış ve bu projelerin kaynak kodları güncellenerek açık kaynak kodlu bu projelere katkı sağlanmıştır.

Önerilen sistem için geliştirilen prototipin ve ağ savunma servislerinin değerlendirilebilmesi amacıyla yazılım tanımlı ağ emülatörü üzerinde test altyapısı oluşturulmuştur. Önerilen sistemin performans ve güvenlik değerlendirmesi, tanımlanan saldırı senaryoları kullanılarak bu test altyapısı üzerinde gerçekleştirilmiştir. YTA temelli siber savunma ile ilgili bazı açık problemler ve gelecekte ele alınabilecek araştırma konuları, tez süresinin farklı aşamalarında belirlenmiş ve bu alanda çalışan araştırmacıların faydalanabileceği şekilde gruplandırılarak raporlanmıştır.

Varsayımlar ve sınırlılıklar

Bu tez kapsamındaki faaliyetler aşağıda belirtilen varsayımlar ve sınırlılıklar çerçevesinde yürütülmüştür:

- Tez kapsamında önerilen sistemin ve geliştirilen ağ savunma servislerinin, bir yazılım tanımlı ağda çalışacağı, veri katmanında bulunan ağ anahtarlarını OpenFlow [17] ve NSH [18] protokolleri ile yapılandıracağı varsayılmıştır.
- Tez çalışmasında, her bilgisayar ağını tehdit edebilecek bazı genel siber saldırı türleri için savunma çözümleri oluşturmaya odaklanılmıştır. YTA mimarisi kaynaklı ortaya

çıkan yeni siber tehditler genel olarak incelenmiş fakat bu tehditlerin önlenmesi ve bu saldırıların etkilerinin azaltılması tez kapsamı dışında bırakılmıştır.

- Tez çalışmasında siber tehditlerin önlenmesine ve azaltılmasına odaklanılmıştır. Sadece saldırı tespitine odaklanan, önleme veya etki azaltma yaklaşımı içermeyen literatür çalışmaları kapsam dışı bırakılmıştır.
- Tez çalışması kapsamında önerilen sistemin değerlendirmesi YTA testleri için kullanılan sanal ağ emülatörü ve sanal anahtarlar ile oluşturulmuş test altyapısı üzerinde gerçekleştirilmiştir.

Literatüre katkı

Bu tez çalışması ile literatürde ilk kez STİ verilerinden faydalanarak YTA'da otomatik ağ seviyesi savunma servislerinin seçilmesini, oluşturulmasını ve SFZ olarak yönetimini sağlayan bir sistem önerisi sunulmuştur. Geliştirilen sistem, STİ verileri analiz edilerek belirlenen saldırgan modelleri ve saldırı senaryoları ile değerlendirilmiştir. Bu tez kapsamında yapılan incelemeler ve gerçekleştirilen çalışmalar sonucunda literatüre sunulan katkılar aşağıda detaylandırılmıştır:

- STİ verileri dikkate alınarak siber tehdit sınıflandırması yapılmış, literatürdeki çalışmalar analiz edilerek bu tehditlere karşı kullanılabilecek ağ seviyesi savunma yaklaşımları tespit edilmiş ve bu savunma yaklaşımlarının YTA ağlarında nasıl uygulanabileceği belirlenmiştir.
- STİ verilerini işleyen, ağ topolojisini, varlıkları ve mevcut savunma mimarisini dikkate alarak değerlendiren, ağ seviyesinde otomatik savunma çözümü oluşturabilen yeni bir sistem önerilmiştir.
- Siber saldırılara karşı ağ seviyesinde savunmaları otomatikleştirmek için STİ verilerinde bulunması gereken bilgileri içeren yeni bir veri modeli önerilmiştir. Bu veri modeli, STIX gibi STİ veri formatlarına eklenti olarak tanımlanması ile organizasyonlar ağ altyapılarını otomatik olarak güncelleyebilecektir.
- Ağ seviyesi savunma servisleri ile sanal ağ fonksiyonlarını kullanarak SFZ oluşturulabilmesini ve siber saldırılara karşı savunma yapılabilmesini sağlayan bir yöntem önerilmiştir.
- STİ verilerinde yer alan yaygın saldırgan modellerine karşı savunma yapmak için kullanılabilecek dört temel savunma servisi geliştirilmiş ve test edilmiştir. Geliştirilen

kara liste, karantina ve tuzak sisteme gönderme servisleri; DDoS, oltalama ve zararlı yazılım saldırı için kullanılabilmektedir. Ayrıca çoklu rota savunma servisiyle, her bir anahtarda rastgele IP (İnternet protokolü) adresi üretilip periyodik olarak rota değiştirilmek suretiyle dinleme saldırılarına karşı savunma gerçekleştirebilmektedir.

- YTA temelli siber savunma alanındaki bazı açık problemler ve gelecekte ele alınabilecek araştırma konuları belirlenmiştir.
- Açık kaynak kodlu projelerde NSH protokolünün kullanılabilmesi için projelerin kaynak kodları güncellenerek açık kaynak kodlu projelere katkı sağlanmıştır.

Tez planı

Bu tez kapsamında yürütülen çalışmalar, sonraki bölümlerde detaylandırılmıştır. Tezin diğer bölümlerinin içeriği aşağıda özetlenmiştir.

İkinci bölümde kavramsal çerçeveyi açıklamak için yazılım tanımlı ağlar, AFS, SFZ, STİ, STİ verilerinde yer alan genel saldırgan modelleri ve tez çalışması kapsamında belirlenen siber tehdit kategorileri konularında genel bilgi sunulmuştur.

Üçüncü bölümde; literatürdeki STİ verileri konusunda yapılan çalışmalar incelenmiş, YTA temelli savunma mimarileri ve yaklaşımları derinlemesine analiz edilmiş ve tez çalışması kapsamında önerilen sistemin ve ağ savunma servislerinin literatürdeki çalışmalarla karşılaştırması yapılmıştır.

Dördüncü bölümde; proaktif ve çevik YTA temelli siber savunma için önerilen sistemin gereksinimleri ve bileşenleri YTA mimarisi çerçevesinde sunulmuş, tez çalışması kapsamında geliştirilen dört ağ savunma servisi hakkında detaylı bilgi verilmiştir.

Beşinci bölümde, tez çalışması kapsamında önerilen sistem ve ağ savunma servislerinin geliştirme ve test altyapısı anlatılmıştır. Sonrasında, yine tez çalışması kapsamında belirlenen saldırgan modelleri kullanılarak oluşturulan saldırı senaryolarına göre önerilen sistem test edilmiş ve değerlendirilmiştir.

Altıncı bölümde; YTA temelli siber savunma konusunda açık ve yeni araştırma konuları dört başlıkta detaylandırılmıştır. Öncelikle daha güvenli, güvenilir ve ölçeklenebilir YTA

mimarileri için yeni araştırma konuları paylaşılmıştır. Sonrasında YTA mimarilerinin diğer teknolojilerle entegrasyonu ve YTA temelli yeni savunma yaklaşımları konularındaki yeni araştırma alanları tartışılmıştır. Son olarak da YTA temelli savunma çözümlerinin değerlendirme kriterleri ve yöntemleri için yapılması gerekenler özetlenmiştir.

Son olarak yedinci bölümde de; tez kapsamında ele alınan problemler ve bu problemlere yönelik yapılan çalışmalar özetlenmiş, elde edilen sonuçlar değerlendirilmiş, tez çalışmaları süresince gerçekleştirilen çalışmalarda öğrenilen dersler paylaşılmış ve gelecekte yapılabilecek çalışma önerileri sunulmuştur.

2. YENİ AĞ TEKNOLOJİLERİ VE SİBER TEHDİT İSTİHBARATI

Bu bölümde; yazılım tanımlı ağların (YTA) genel yapısı açıklanmakta, AFS, SFZ kavramları anlatılmakta, STİ verilerinin oluşturulması ve kullanılması hakkında temel bilgiler paylaşılmakta, STİ verileri ile elde edilebilecek temel saldırgan modelleri ve siber tehdit kategorileri hakkında özet bilgi sunulmaktadır.

2.1. Yazılım Tanımlı Ağlar (YTA)

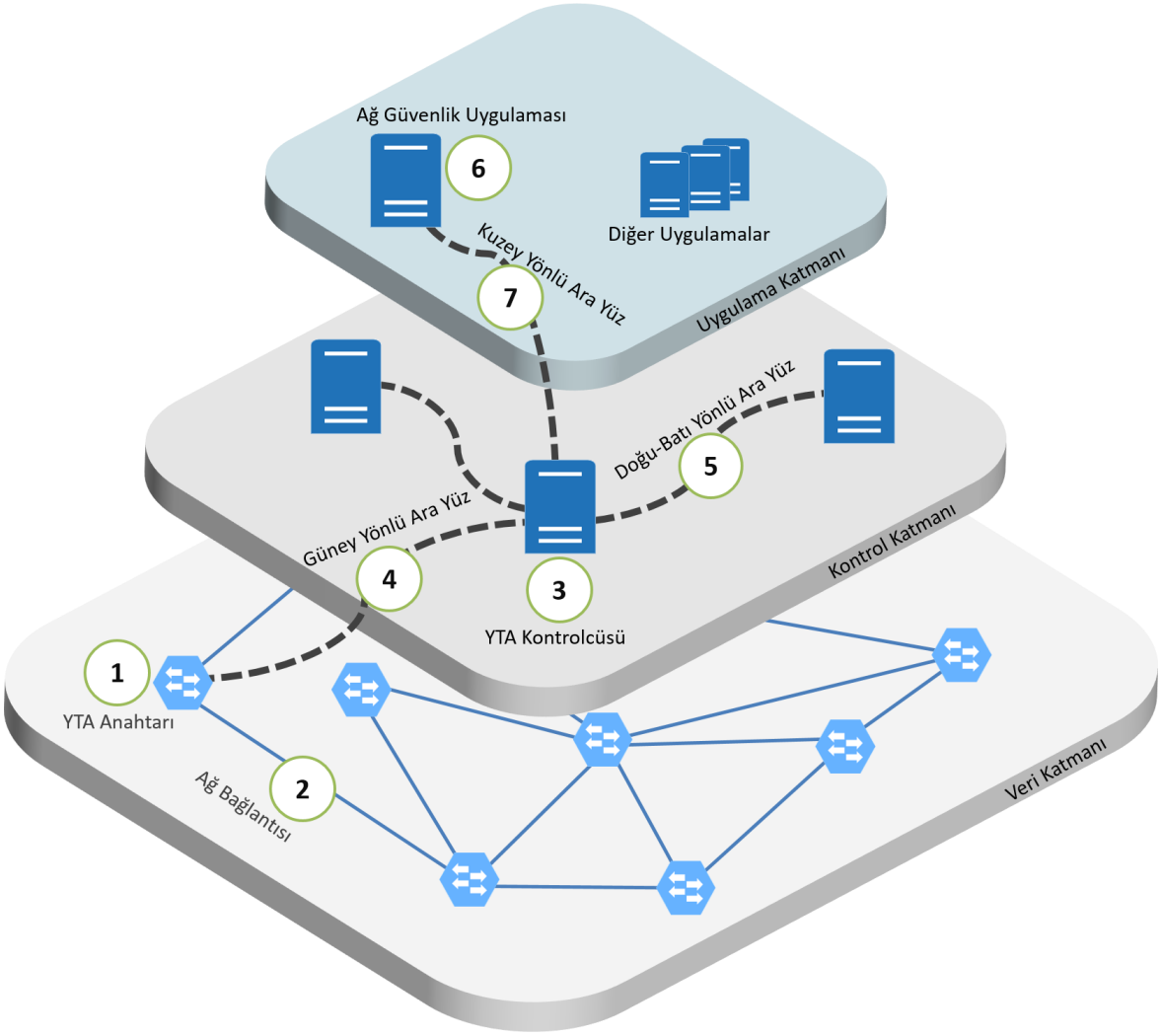
Geleneksel ağlar çoğu zaman karmaşıktır ve yönetilmesi zordur. Organizasyonların ağ yapılandırma ve değişiklik yönetimi süreçleri de bundan olumsuz etkilenmektedir. Ayrıca, mevcut ağların kontrol katmanı ve veri yönlendirme katmanı dikey olarak bütünleşiktir [19]. Bu durum da ağ cihazlarının bakımını daha da zorlaştırmaktadır. Yazılım tanımlı ağlar (YTA), veri yönlendirme katmanını ve kontrol katmanını ayırarak bilgisayar ağlarının esnek yönetimini sağlayan bir mimaridir. Yazılım tanımlı ağların temel özellikleri aşağıdaki şekilde özetlenebilir [20]:

- Mantıksal olarak merkezi yönetim sağlaması
- Açık standartlarla programlanabilir ara yüzler sunması
- Gelişmiş ağ anahtarı yönetim protokolleri kullanabilmesi
- Kolayca sanallaştırılmış mantıksal ağlar oluşturulabilmesi
- Merkezi izleme bileşenlerinin kullanılabilmesini sağlaması

YTA merkezi ve esnek yönetimi, programlanabilirlik, bakım kolaylığı gibi teknik kabiliyetler sunmaktadır. Bu özellikleri gibi özellikleri dolayısıyla merkezlerinde, omurga ağlarında, kurumsal ağlarda ve kablosuz ağlarda yaygın olarak kullanılmaktadır [21]. YTA mimarisi, Şekil 2.1'de gösterildiği gibi üç katmandan ve katmanlar arası iletişim ara yüzlerinden oluşmaktadır [22].

Veri katmanındaki YTA anahtarı (1), gelen trafiği (2) programlanabilir akış tablolarındaki kurallara göre iletir ve içinden geçen trafik hakkında istatistiksel bilgiler toplar. Gelen bir paket, akış tablolarındaki kurallardan hiçbiriyle eşleşmezse, YTA anahtarı, paketi güney ara yüzü (4) üzerinden kontrol katmanındaki YTA kontrolcüsüne (3) iletmek için varsayılan eylemi gerçekleştirir. YTA kontrolcüsü, YTA anahtarlarından gelen paketler için ne yapılacağına karar verir ve anahtarlar için yeni akış kuralları (yönlendir, güncelle, düşür vb.)

oluşturur. YTA kontrolcüsü periyodik olarak veya talep üzerine güney ara yüzü kullanarak anahtarlardan istatistiksel bilgiler de toplayabilir. Kontrol katmanında birden fazla YTA kontrolcüsü olabilir. Bu durumda, kontrolcüler birbirleriyle doğu-batı ara yüzü (5) üzerinden iletişim kurarlar. Uygulama katmanındaki uygulamalar (6) (ağ güvenliği, trafik izleme vb.), YTA kontrolcüsü üzerinden ağda değişiklikler yapma talebi gönderebilir veya YTA kontrolcüsünün kuzey ara yüzü (7) üzerinden ağı izleyebilir.



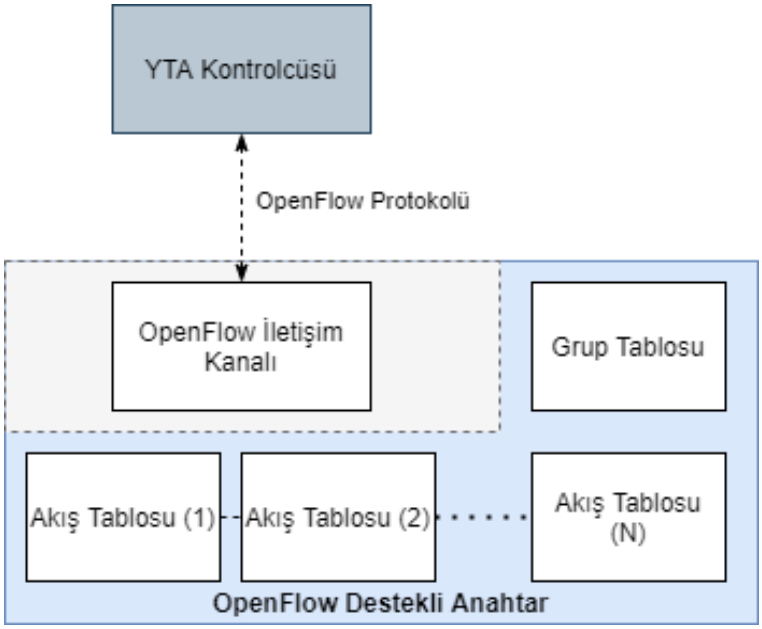
Şekil 2.1. Yazılım tanımlı ağ mimarisi

Yazılım tanımlı ağların programlanabilirliği ve kontrolün merkezîleştirilmesi gibi özellikleri kullanılarak daha iyi savunma mekanizmaları geliştirilebilirken, bu özelliklerinden dolayı daha tehlikeli siber saldırılar da yapılabilmektedir [22–25]. Literatürde, hem YTA sistemlerine karşı saldırılar [26, 27] hem de siber tehditlere karşı YTA tabanlı savunma çözümleri yer almaktadır. İnternet servis sağlayıcı (İSS) ağları ve bulut gibi daha geleneksel

ve yaygın olarak kullanılan alanların ötesinde, akıllı şebekeler [28, 29] ve endüstriyel ağlar [30] gibi diğer alanlarda da gelişmiş siber güvenlik için YTA güçlü bir araç olarak önerilmiştir.

OpenFlow protokolü

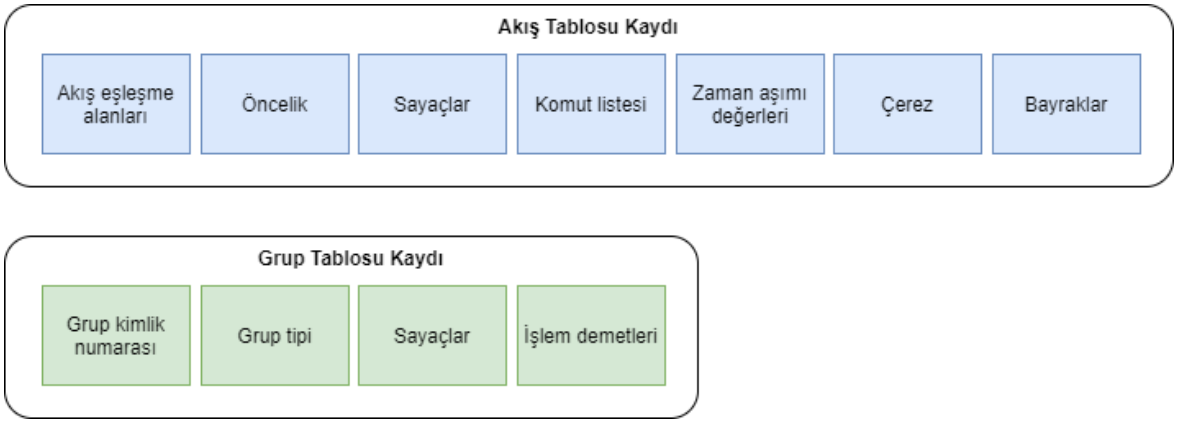
Yazılım tanımlı ağlarda YTA destekli anahtarlar kullanılmaktadır. Bu anahtarlar YTA kontrolcüsü ile iletişim kurarak ağ trafiğini yönlendirmektedir. YTA anahtarı ile YTA kontrolcüsü arasındaki iletişimin sağlanması için kullanılan endüstri standardı protokol OpenFlow [17] protokolüdür. 2009 yılında ilk sürümü yayımlanan protokol sürekli güncellenmektedir. Anahtar üreticileri de yayımlanan yeni protokol sürümlerine göre anahtar üretimlerini gerçekleştirmektedir. OpenFlow protokolünün v1.4 sürümü için bir YTA anahtarında bulunan temel bileşenler Şekil 2.2'de gösterilmektedir [17].



Şekil 2.2. OpenFlow destekli anahtarların iletişim kanalı ve bileşenleri [17]

OpenFlow destekli YTA anahtarında bir veya birden fazla akış tablosu, bir grup tablosu ve YTA kontrolcüsü ile iletişim kurulmasını sağlayan OpenFlow iletişim kanalları bulunmaktadır. YTA kontrolcüsü, OpenFlow protokolünü kullanarak akış ve grup tablolarında; akış kuralı ekle, sil ve güncelle işlemlerini yapabilmektedir. Akış tablosu ve grup tablosunda bulunan kayıtların veri yapıları Şekil 2.3'te gösterilmektedir. Her bir akış

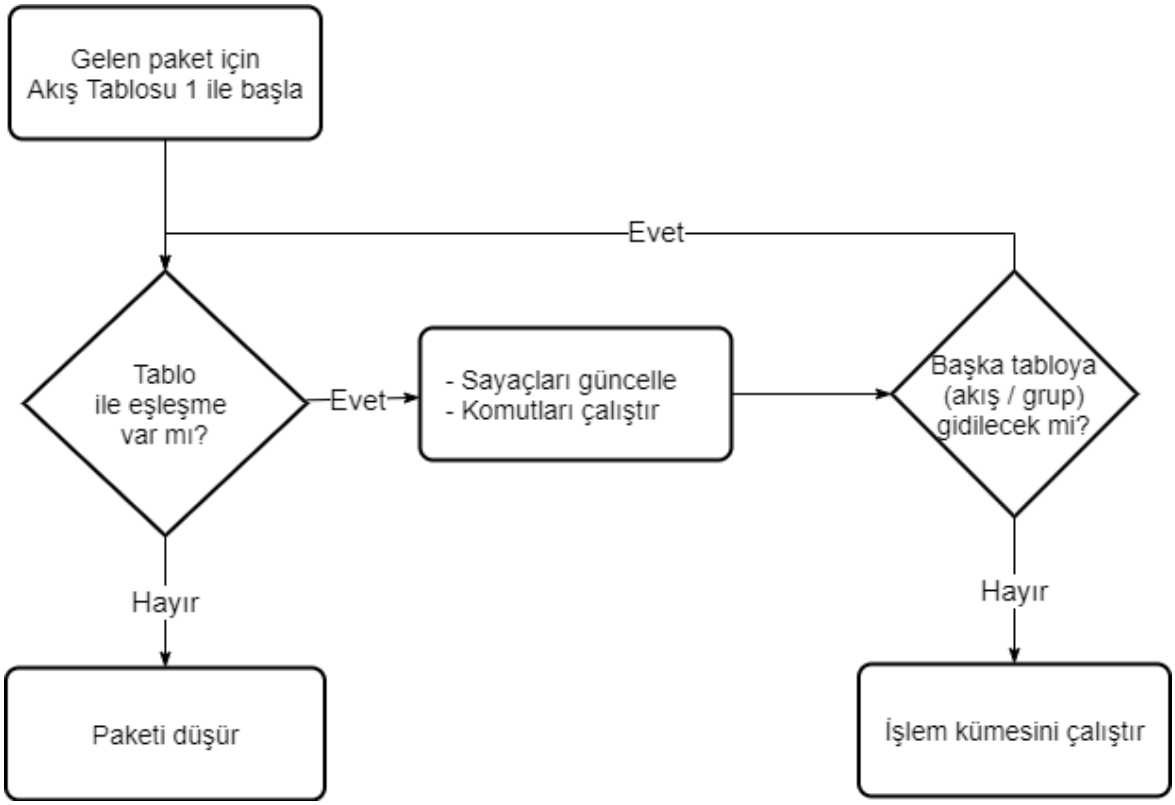
tablosunda; akış eşleşme alanları, eşleşme öncelik değeri, sayaçlar, eşleşme durumunda uygulanacak komut listesi, zaman aşımı değerleri çerez ve bayraklar yer almaktadır.



Şekil 2.3. OpenFlow destekli anahtarlarda bulunan akış ve grup tablolarının yapıları [17]

Bir paket için YTA anahtarında yürütülen işlem adımları Şekil 2.4'te verilmiştir [17]. Paket alınınca öncelikle ilk akış tablosu eşleştirme adımı ile başlanmaktadır. Bir paket bir akış tablosu girdisi ile eşleştiğinde o akış kuralında yer alan komutlar çalıştırılmaktadır. Bir paketin birden fazla kayıtla eşleşmesi durumunda sadece yüksek öncelikli kayıt ile işlem yapılmaktadır. Komutlar çalıştırıldığında; paket üzerinde değişiklik yapılabilmekte, paket düşürülebilmekte, belirli bağlantı noktalarına yönlendirilebilmekte, başka bir akış tablosunun işletilmesi başlatılabilmekte veya grup tablosunda işlenmesine karar verilebilmektedir. Paket üzerinde yapılacak işlemler, işlem anında yapılabileceği gibi sürecin sonunda yapılması için biriktirilebilmektedir. Paket işlenirken kayıt ile ilgili sayaçlar güncellenmektedir. İlk akış tablosu işlendikten sonra eğer süreç devam edecek ise diğer tablolar da sırası ile işletilmektedir. Bu aşamada işlem yapılan tablolar akış tabloları veya grup tablolarından biri olabilmektedir. Anahtar, sürekli olarak her bir kaydın zaman aşımı değerlerini kontrol etmekte ve zamanı dolan kayıtları silmektedir.

Grup tablosunda yer alan her bir kayıt bir kimlik numarası ile belirlenmektedir. Akış tablolarında grup kimlik numarası kullanılarak ilgili grup tablosu kaydının işletilmesi başlatılabilmektedir. Grup tablosunda bir veya birden fazla işlem demeti yer almaktadır. Her işlem demetinde işlem listeleri bulunmaktadır. Grup tipine göre; sadece belirli bir işlem demeti, aynı anda tüm işlem demetleri, bir algoritmaya göre herhangi bir demet veya hata durumuna yol açmayan bir demet çalıştırılabilmektedir.



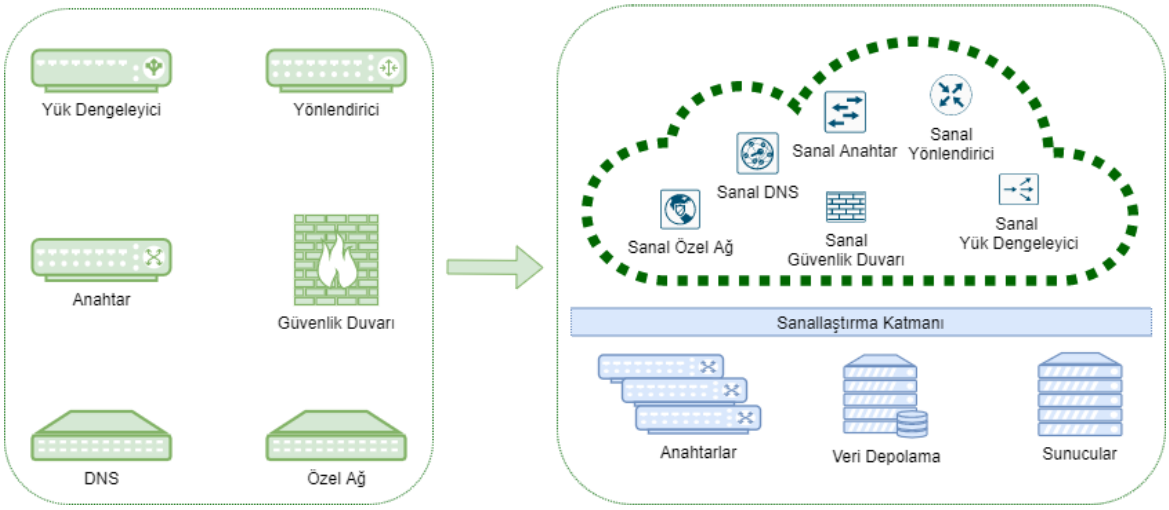
Şekil 2.4. OpenFlow destekli anahtarda paket işleme adımları [17]

OpenFlow iletişim kanalı üzerinden kontrolcü ve anahtarlar sürekli olarak mesajlarla iletişim kurabilmektedir. OpenFlow protokolü ile haberleşme, kontrolcü ve anahtar arasında el sıkışma mesajları ile başlamaktadır. Kontrolcü; anahtarların özelliklerini ve yapılandırmasını öğrenmek, anahtarların durumunu değiştirmek ve anahtarların sorduğu paketler ile ilgili kararı iletmek için anahtarlara mesajlar gönderebilmektedir. Anahtarlar da kendi durumlarını ve akış kurallarındaki değişiklikleri bildirmek için kontrolcüye mesaj gönderebilmektedir.

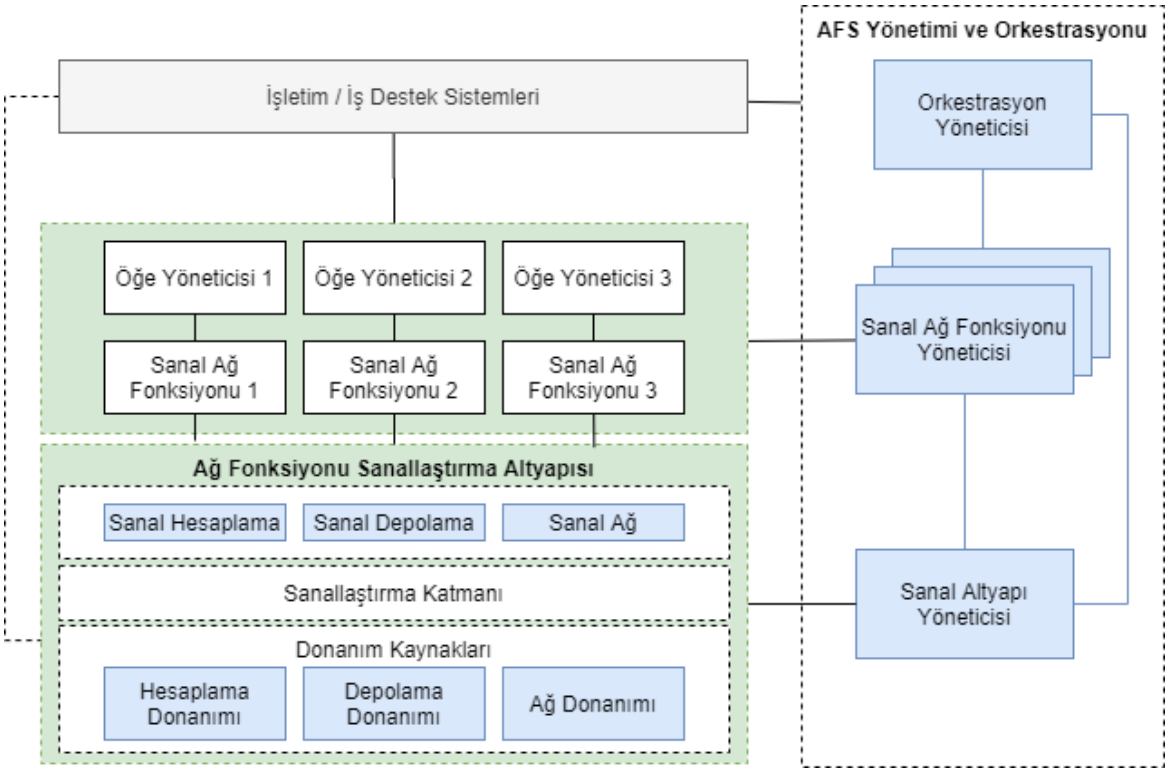
2.2. Ağ Fonksiyonu Sanallaştırma (AFS)

AFS, donanım tabanlı ağ teknolojilerinin ve fonksiyonlarının standart bilgi teknolojileri sanallaştırma altyapılarının içine taşınmasını (Şekil 2.5) ve sanallaştırma ortamlarından yönetilmesini (Şekil 2.6) [9] sağlayan bir ağ mimarisidir. YTA ve AFS ağ mimarisi teknolojileri, organizasyonların yatırım ve işletim maliyetlerini azaltmakta, ağ altyapılarının yönetimini ve yapılandırmasını kolaylaştırmaktadır. AFS teknolojisinin organizasyonların ihtiyaçlarını karşılayabilmesi için bazı gereksinimleri karşılaması gerekmektedir. Bu gereksinimlerin başında ağ fonksiyonlarının; performanslı şekilde çalışabilmesi, farklı

sanallaştırma altyapılarına taşınabilmesi, esnek şekilde genişleyebilmesi, hata durumunda tekrar başlatılabilmesi ve sürekliliğinin garanti edilmesi gelmektedir. Bunlara ek olarak AFS ağ mimarisinin; rol tabanlı yetkilendirme, kimlik doğrulama, etkin enerji yönetimi, servis izleme gibi gereksinimleri de karşılaması gerekmektedir.



Şekil 2.5. Ağ fonksiyonlarının sanallaştırılması



Şekil 2.6. AFS referans mimarisi [9]

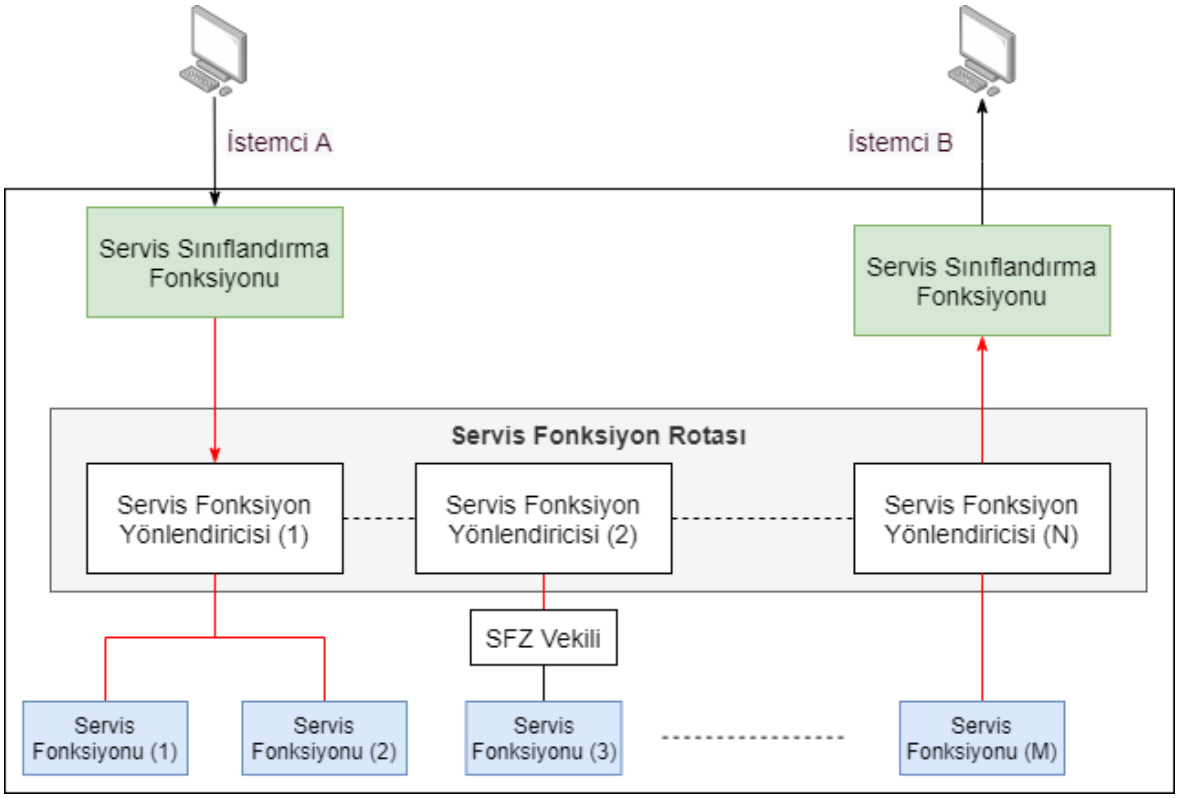
AFS referans mimarisi (Şekil 2.6) aşağıdaki yedi temel bileşenden oluşmaktadır [9]:

1. *Sanal ağ fonksiyonları*: Ağ fonksiyonlarının sanal ortamda gerçekleştirilmesini sağlayan bileşenlerdir. Bir sanal ağ fonksiyonu, bir sanal sunucuda kurulu olabileceği gibi birden fazla sunucuda kurulan sistemlerin birleşimi de olabilmektedir.
2. *Öğe yöneticileri*: Bir veya birden fazla sanal ağ fonksiyonunu yöneten bileşenlerdir.
3. *AFS altyapısı*: Sanal ağ fonksiyonlarının kurulduğu yönetildiği ve işletildiği fiziksel ve sanal ortamların bütünüdür. Altyapı; hesaplama, depolama ve ağ donanım kaynaklarını yöneterek bunların sanallaştırılmasını sağlar.
4. *Sanal altyapı yöneticisi*: AFS altyapısı üzerinde kaynak yönetimi ve işletimi faaliyetlerini yerine getirir.
5. *Sanal ağ fonksiyonu yöneticileri*: Sanal ağ fonksiyonlarının yaşam döngüsünün (oluşturma, güncelleme, genişletme, sonlandırma vb.) yönetimini sağlar.
6. *Ağ sanallaştırma orkestrasyon yöneticisi*: AFS altyapısı üzerinde ağ servislerinin sunulması ve yönetimi ile ilgili orkestrasyon faaliyetlerini yürütür.
7. *İşletim ve iş destek sistemleri*: Organizasyonların iş hedefleri doğrultusunda AFS altyapıları ile entegre çalışan hizmet servisleri yönetimi, ağ yapılandırma, hata yönetimi, envanter yönetimi, talep yönetimi gibi yazılımlardır.

AFS mimarilerinin, yatırım ve işletim maliyetlerini düşürmesi ve işletiminin kolay olmasının yanı sıra başka avantajları da vardır. Bu teknoloji ile organizasyonlar; yeni servisleri çevik şekilde hayata geçirebilmekte, esnek şekilde servisleri genişletebilmekte veya daraltabilmekte, donanım değiştirmeye ihtiyaç duymadan hızlı şekilde teknolojik yeniliklere ayak uydurabilmektedir.

2.3. Servis Fonksiyon Zinciri (SFZ)

SFZ, belirlenen politikalara göre paketlerin veya akışların farklı sanal ağ fonksiyonları üzerinden sıralı şekilde yönlendirilmesini sağlayan teknolojilerden biridir. Servis fonksiyonları da ağ üzerinde SFZ ile kendisine yönlendirilen paketlerde işlem yapan sanal veya fiziksel bileşenlerdir. SFZ dört mantıksal bileşenden oluşmaktadır [10]. Bunlar Şekil 2.7'de gösterildiği gibi sınıflandırıcılar, servis fonksiyonu yönlendiricileri, servis fonksiyonları ve SFZ vekilleridir. SFZ vekiline bağlı olan servis fonksiyonları arasındaki iletişim hariç tüm bileşenler kendi aralarında paketleri sarmalayarak ve NSH kullanarak oluşturulan paketler vb.) oluşturdukları paketlerle haberleşmektedir.



Şekil 2.7. SFZ mimarisi ve mantıksal bileşenleri [10]

Servis fonksiyonları ağ seviyesinde gerçekleştirilebilecek her türlü ağ fonksiyonunu yerine getirmek için ağda konuşlandırılmaktadır. Servis fonksiyonları, ağ üzerinden görüntü sıkıştırma, derin paket incelemesi, güvenlik duvarı, içerik filtreleme gibi işlemleri gerçekleştirebilmektedir. Bir servis fonksiyonu birden fazla SFZ içinde kullanılabilmekte, aynı zamanda bir SFZ içinde birden fazla servis fonksiyonu bulunabilmektedir.

Servis fonksiyonları SFZ teknolojisi ile uyumlu ise doğrudan SFZ altyapılarına entegre edilebilmektedir. Fakat SFZ teknolojisi ile uyumlu olamayan servis fonksiyonlarının entegrasyonu için SFZ vekilleri kullanılması gerekmektedir. SFZ vekili, zincirden gelen paketleri işleyip servis fonksiyonuna göndermekte, servis fonksiyonundan gelen paketleri sarmalayarak tekrar zincire dahil etmektedir.

Servis fonksiyon yönlendiricileri, ağ üzerinden paketleri servis fonksiyonlarına yönlendirerek paketlerin servis fonksiyonlarında işlenebilmesini sağlamaktadır. Servis fonksiyon yönlendiricileri SFZ sarmalamalarını anlayabilen ve buna göre paketleri yönlendirebilen anahtarlardır. Anahtarlar paketleri birden fazla bağlantı noktasına aktararak

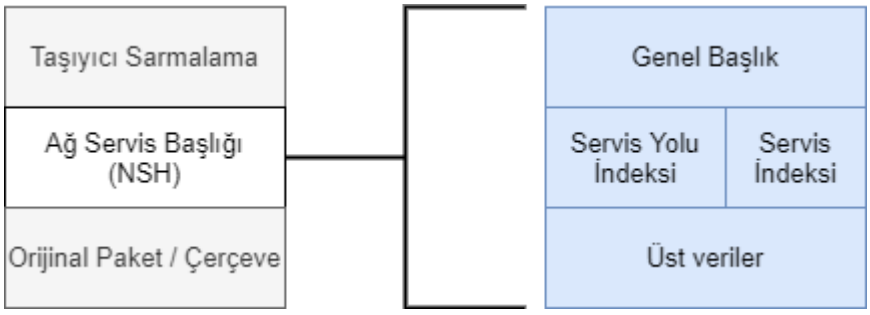
servis fonksiyonlarının eş zamanlı işletilebilmesini sağlayabilmektedir. Bu şekilde paralel işletilebilen servis fonksiyon zincirleri oluşturulabilmektedir.

Servis sınıflandırma fonksiyonları, ağ trafiğindeki paketleri eşleşmelere göre çeşitli servis zincirlerine yönlendirmekte ve servis zincirinin sonuna ulaşan paketleri zincirden çıkararak ağa dahil etmektedir. Servis sınıflandırma fonksiyonları, eşleşmeleri anahtarlar üzerinden yapabileceği gibi bir servis fonksiyonuna yönlendirilerek servis zinciri seçimini yapabilmektedir.

Servis fonksiyon zincirleri YTA kontrolcüsü tarafından önceden veya anlık olarak oluşturulabilmektedir. Servis fonksiyon zincirleri için kullanılacak servis fonksiyonları ve sıraları belirlendikten sonra bu zincir için gerekli kuralları YTA anahtarlarına yüklenmektedir. YTA anahtarları servis fonksiyon yönlendiricisi ve servis sınıflandırma fonksiyonu görevlerinden herhangi birini üstlenebilmektedir.

NSH protokolü

Servis fonksiyon zincirlerine dahil olan fonksiyonlar paketleri işlerken birbirleri arasında bilgi paylaşımı yapabilmektedir. SFZ oluşturmak için NSH [18] protokolü tanımlanmıştır (Şekil 2.8). Bu protokol paketlere veya Ethernet çerçevesine yeni bir protokol başlığı eklenmektedir. Bu başlık içinde servis yolu indeksi, servis indeksi alanları ve servis fonksiyonları arasında paylaşım yapılmasını sağlayacak veri alanları yer almaktadır.



Şekil 2.8. NSH sarmalama ve NSH yapısı [18]

Her bir servis zinciri için tekil servis yolu indeksi oluşturulmaktadır. Bir servis zincirindeki servis fonksiyonlarının sırasını belirlemek için servis indeksi alanı kullanılmaktadır. Bu alan 255'ten geriye doğru azalan değere sahip olmakta ve her bir servis fonksiyonu görevini

tamamladıktan sonra bu indeks bir azaltılmaktadır. Servis fonksiyonu yönlendiricileri de servis yolu indeksi ve servis indeksi bilgilerini eşleşme kriteri olarak kullanmak suretiyle paketleri ilgili servis fonksiyonlarına yönlendirmektedir.

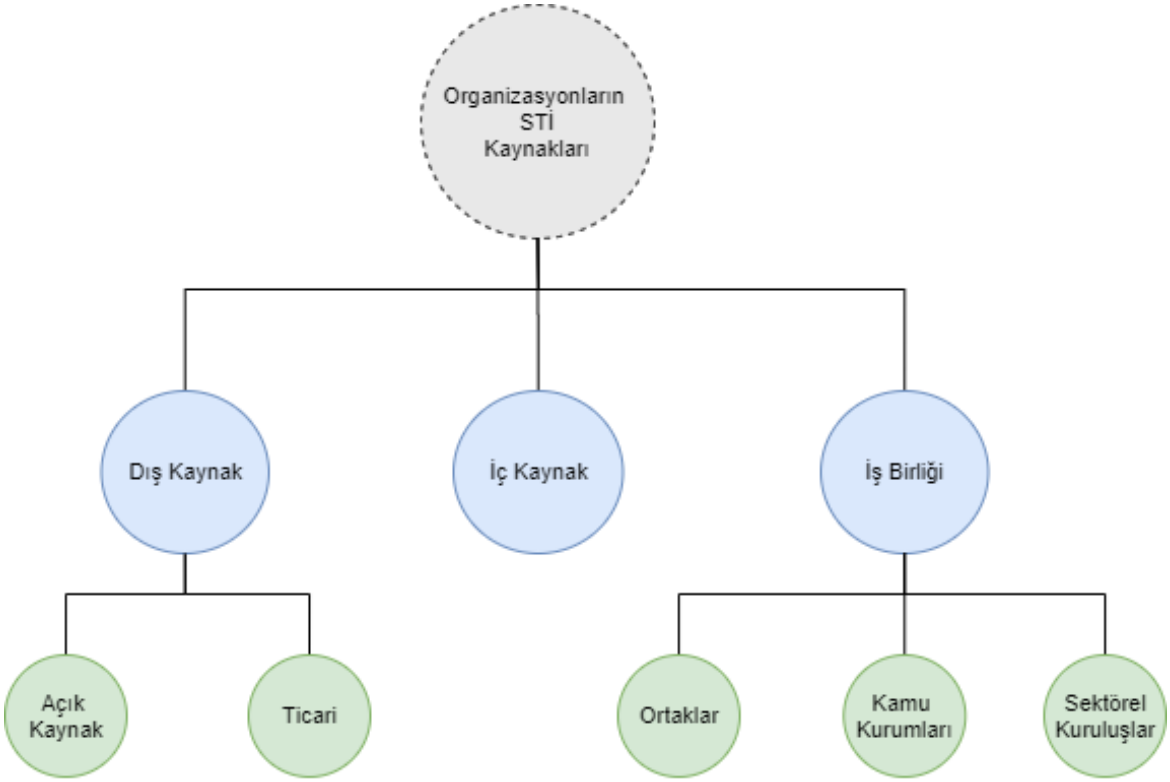
2.4. Siber Tehdit İstihbaratı (STİ)

Siber tehdit türlerinin ve tehdit vektörlerinin artmasıyla birlikte siber saldırı olayları da artmaktadır. Buna bağlı olarak, günümüzde devletlerin ve organizasyonların güvenlik kaygıları kritik bir düzeye ulaşmıştır. Geleneksel ağ savunma yaklaşımında; siber tehdit verileri toplanır, bilişim uzmanları tarafından gruplandırılır ve belirlenen tehditleri etkisiz hale getirmek için önlemler alınır. Günümüzde siber saldırganların amaçları ve yetenekleri hakkında güncel bilgileri almak ve proaktif siber savunma yaklaşımlarını kullanabilmek bir gereklilik haline gelmiştir [1]. Saldırganların faaliyetleri, kullandıkları saldırı araçları, zararlı yazılımlar ve diğer işaretler hakkındaki bilgileri organize etmek için yeni yaklaşımlar gerekmektedir [13].

Günümüzde saldırıların %75'inin bir günde yayıldığı ve %40'tan yüksek bir ihtimalle diğer bir organizasyonu bir saat içinde etkilediği belirtilmektedir [31]. Bu da saldırı bilgilerinin organizasyonlar arasında hızlı şekilde paylaşılmasının önemini göstermektedir. Genel olarak STİ, belirli hedefleri siber tehditlere karşı korumak amacıyla birden fazla kaynaktan toplanan ve doğrulanan bilgi olarak tanımlanmaktadır [1, 5]. Organizasyonlar, STİ verilerini Şekil 2.9'da da gösterildiği gibi üç farklı kaynaktan elde edebilir.

Siber tehdit istihbarat verisi yönetim süreci beş aşamadan oluşur [5]:

1. Sürecin planlanması ve yönetilmesi,
2. İlgili kaynaklardan insan istihbaratı, açık kaynak istihbaratı, sinyal istihbaratı, görüntü istihbaratı, ölçüm ve imza yoluyla ham verilerin toplanması,
3. Toplanan verilerin ayrıntılı analizinin yapılabilmesi için standart bir formata dönüştürülmesi,
4. Tehditleri tanımlamak ve uygun karşı önlemleri belirlemek için standart formata dönüştürülmüş verilerin analizi,
5. Uygun koruyucu önlemlerin alınabilmesi için sonuçların siber tehdit bilgileri ve göstergeleri olarak paylaşılması.

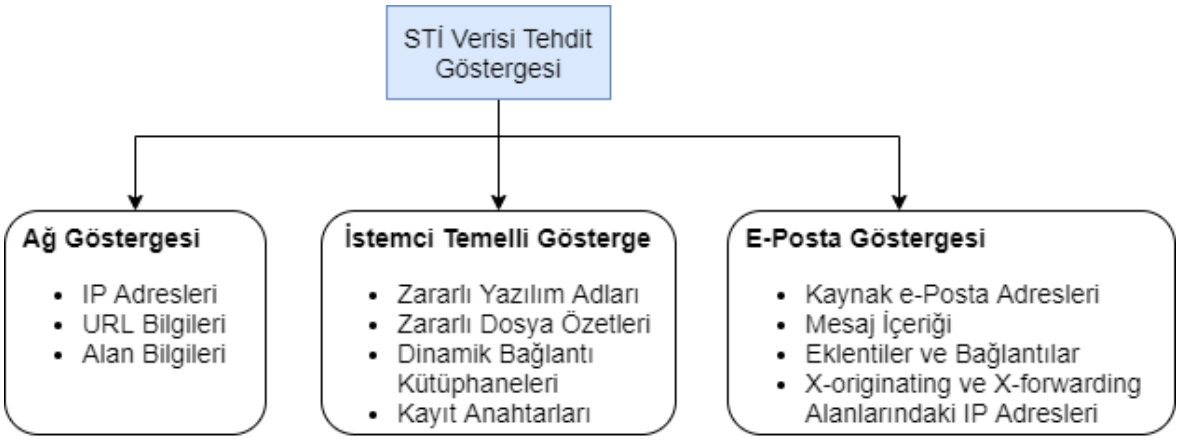


Şekil 2.9. Organizasyonların STİ veri kaynakları

STİ verilerinin ilk kaynağı organizasyonların kendi yeteneklerdir. Organizasyonun güvenlik cihazları kullanılarak siber güvenlik olayları personel tarafından toplanır ve analiz edilir. Ancak bu zaman alıcı, yoğun emek isteyen, potansiyel olarak da hataya açık bir süreçtir [5]. En çok kullanılan STİ verisi kaynağı, ticari hizmetlerin veya açık kaynakların kullanılmasıdır. Organizasyonlar bu şekilde kendileri veri paylaşmadan üçüncü partilerden STİ verisi alabilirler. STİ verisi elde etmenin üçüncü ve en etkili yolu da farklı organizasyonlarla siber istihbarat bilgilerinin paylaşılmasıdır. Bu şekilde organizasyonlar karşı karşıya kaldığı tehdit bilgilerini STİ veri formatına çevirerek paydaşlarına gönderir ve onlardan benzer formatta STİ verisi alır.

STİ verilerinin oluşturulmasını ve paylaşılmasını sağlayacak standartlar tanımlanmıştır. Bazı çalışmalar [32–34] STİ standartlarını ve platformlarını detaylı şekilde incelenmektedir. En yaygın kullanılan standartların bazıları STIX [11], OpenIoC [13], Güvenilir Otomatik İstihbarat Bilgisi Paylaşımı (Trusted Automated Exchange of Intelligence Information, TAXII) [35] ve Siber Gözlenebilir İfade Dili (Cyber Observable Expression, CybOX) [36] olarak sıralanabilir. Fakat bu standartlara göre paylaşılan STİ verileri günümüzde çoğunlukla insanların anlamlandırabileceği şekilde kullanılmaktadır [37, 38]. Siber tehdit

istihbaratı çoğunlukla tehdit göstergelerini paylaşmak için kullanılmaktadır. Bu göstergeler; ağ göstergeleri, istemci temelli göstergeler ve e-posta göstergeleri olarak gruplandırılmaktadır (Şekil 2.10) [39].



Şekil 2.10. STİ verisinde yer alan tehdit göstergeleri [39]

Çoğu STİ verisi, otomatik olarak önleme dönüştürülmeye uygun bilgileri içermemektedir. Bu sebeple organizasyonlar, STİ verilerini kullanarak güvenlik duvarlarının, uç nokta koruma sistemlerinin, saldırı önleme sistemlerinin yapılandırmalarını uzman personelin çalışmaları ile gerçekleştirmektedir.

Siber tehdit istihbaratı bilgilerinin paylaşılması önemli olmasına rağmen organizasyonlar farklı sebeplerden dolayı siber istihbarat bilgilerini paylaşmayı tercih etmemektedir. Bu tercihin sebeplerinden en önemlileri aşağıda listelenmiştir [31, 40, 41]:

- Toplumda negatif etki yaratması endişesi
- Yasal ve mahremiyet ile ilgili hususların olması
- Kaliteli istihbarat bilgilerinin oluşturulamaması
- Paylaşılacak ağda güvenilmeyen organizasyonların olması
- Bütçe ile ilgili hususların olması
- Siber saldırıların doğasının sürekli değişiyor olması
- Hedef alınan organizasyonun gerçekleşen siber olayların farkında olmaması
- STİ verisi paylaşarak siber saldırıların ilerlemesinin önlenemeyeceği inancının olması
- STİ verisini paylaşmama yönünde doğal bir eğilim veya içgüdünün olması

STİ verilerinin kaliteli ve sürekli olarak paylaşılabilmesi için organizasyonların yukarıda belirtilen engelleri aşması gerekmektedir. Aşağıda; STİ verilerinin oluşturulması ve paylaşılması için yaygın olarak kullanılan STIX ve TAXII protokolleri ile ilgili özet bilgi paylaşılmaktadır.

2.4.1. STIX siber tehdit istihbaratı veri standardı

STIX [42] standardı, günümüzde STİ verisi paylaşmak için endüstri standardı olarak kullanılan veri yapısıdır. Bu standart, siber tehditleri ve örnek olayları tanımlamak için aşağıdaki özelliklere sahiptir:

- Siber tehdit alanındaki bilgilerin ifade edilebilmesi
- İnsan ve makine tarafından okunabilir şekilde olması
- Kullanım durumlarına göre özelleştirilebilmesi
- Tüm siber tehditleri tekrar tanımlamak yerine mevcutta olan veri tanımlarının kullanılabilmesi
- Veri formatında yer alan zorunlu bilgilerin azaltılarak daha çok siber güvenlik senaryosunda kullanılabilmesi

STIX v1.2 standardı temel olarak aşağıdaki veri yapılarını bünyesinde barındırmakta ve bu veri yapıları ile istihbarat verilerinin formatını belirlemektedir.

- Gözlemlenen veri, ağ ve bilgisayarla ilişkili ölçülebilir olay veya bilgidir. Bu kapsamda CybOX [43] standardı ile ifade edilen dosyalar (adı, boyutu, özet değeri vb.), ağ bilgileri (IP adresi, port vb.), dizin bilgileri gibi bilgiler tanımlanmaktadır.
- Taktikler, Teknikler ve Prosedürler (Tactics, Techniques and Procedures, TTP), siber olayı tanımlayan davranışları ifade etmek için kullanılmaktadır. TTP ile zararlı yazılımlar, saldırı örüntüleri ve kullanılan zafiyetler tanımlanmaktadır.
- Gösterge, siber güvenlik bağlamında bir varlığı veya davranışı ifade eder. Göstergeler; gözlemlenen durumları, ilgili diğer göstergeleri, eylemleri, hareket tarzlarını ve kullanılan TTP bilgilerini içerebilmektedir.
- Olay, gerçekleşmiş bir olayı ifade eden veri tanımıdır. İlgili göstergeler ve diğer olaylar, kullanılan yöntemler, alınan önlemler, talep edilen önlemler ve gözlemlenenleri içerebilmektedir.

- Saldırı dizisi, bir kurum ve organizasyona yapılan eylemler bütününü tanımlamak için kullanılmaktadır.
- Tehdit aktörü, zararlı faaliyetlerde bulunan aktörlerin karakterlerini tanımlamak için kullanılmaktadır. İlgili diğer aktörleri, önceki eylemleri ve kullandıkları yöntemleri açıklamaktadır.
- Hedef istismar, zararlı aktörlerin taktik, teknik ve yöntemlerinde kullandığı ve yazılımlarda, sistemlerde, ağda veya yapılandırmada olan zayıflık veya kırılganlıkları tanımlamak için kullanılmaktadır. Hedef istismar ile ilgili diğer hedef istismarlar ve bu istismar için alınabilecek önlemler de tanımlanabilmektedir.
- Hareket tarzı, olay veya göstergeler için alınabilecek önlemleri ve önlem için izlenecek yolu tanımlamaktadır.

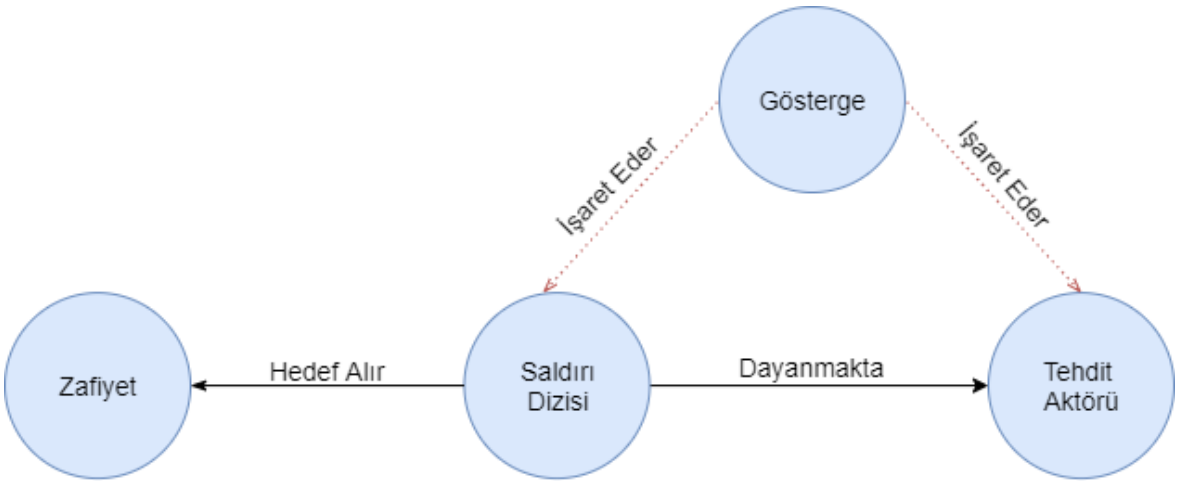
STIX standardı belirli aralıklarla güncellenmektedir. Son olarak STIX v2.1 [11] standardı yayımlanmıştır. STIX v2.1 ile STIX v1.2'de yer alan veri modeli tamamen güncellenmiştir. Bu güncellenen standart ile veri modelinin çizge gösterimine uygun olması, JSON veri formatında verilerin paylaşılması, CybOX [36] eş değeri verileri içinde barındırması, makineler arası veri paylaşımının daha sistematik olması hedeflenmiştir.

STIX 2.1 ile 18 veri yapısı 2 tane ilişki yapısı tanımlanmıştır. Bu yapılarla siber istihbarat veri ağları oluşturulabilmektedir. Bu veri yapıları arasında önemli olanlar aşağıda özetlenmiştir:

- Saldırı örüntüsü: Kurbanları ele geçirmeye çalışan saldırıların izledikleri yolları ifade eder.
- Saldırı dizisi: Bir zaman aralığında bir grup hedefe karşı yapılan zararlı aktivite veya saldırı dizisidir.
- Hareket tarzı: Siber tehdit istihbaratı verisi ile paylaşılan tehdide karşı uygulanabilecek işlem önerileridir.
- Kimlik: Gerçek kişi, organizasyon, grupları tanımlamak için kullanılan tekil kimlik bilgisidir.
- Gösterge: Zararlı veya şüpheli siber aktiviteleri tespit etmek için kullanılabilecek örüntülerdir.
- Altyapı: Saldırılarda kullanılan sistem, servis, fiziksel veya sanal kaynaklardır.
- Yer: Coğrafi yeri tanımlamak için kullanılır.

- Zararlı yazılım: Tehdit oluşturabilen zararlı kodlar veya yazılımlardır.
- Zararlı yazılım analizi: Zararlı yazılım için gerçekleştirilen statik veya dinamik analiz sonuçlarıdır.
- Gözlemlenen veri: Ağ ve bilgisayarla ilişkili ölçülebilir olay veya bilgileri içeren verilerdir.
- Tehdit aktörü: Zararlı eylemlerde bulunma ihtimali olan kişi, grup ve organizasyonlarıdır.
- Araç: Tehdit aktörleri tarafından saldırılarda veya saldırı hazırlıklarından kullanılan kötücül betikler ve yazılımlardır.
- Zafiyet: Sisteme veya ağa erişmek için saldırganlar tarafından doğrudan kullanılan yazılım hatalarıdır.

Şekil 2.11'de STIX 2.1 veri formatına göre oluşturulmuş örnek bir siber istihbarat veri yapısı yer almaktadır. Bu veride bir saldırı dizisi tanımı yapılmıştır. Bu saldırı dizininin belirli bir zafiyeti hedef aldığı ve belirli bir tehdit aktörüne dayandığı belirtilmiştir. Ayrıca STİ verisi içinde bu saldırı dizisini ve tehdidi tanımlayan bir göstergeye yer verildiği görülmektedir.



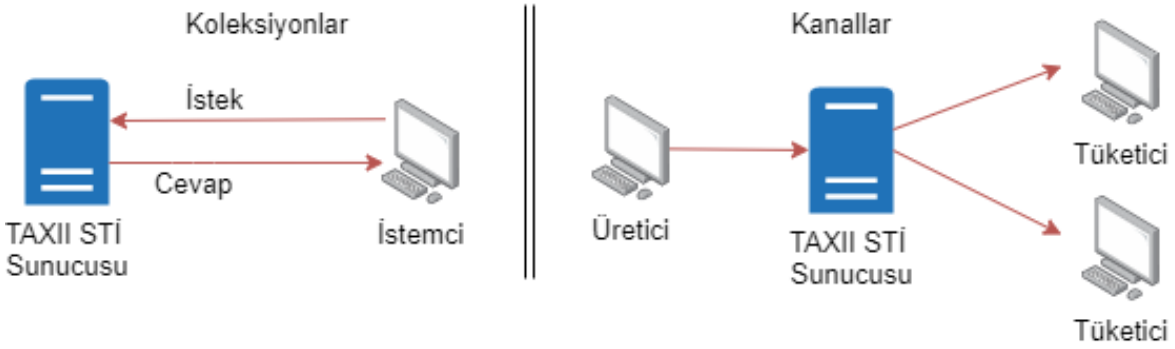
Şekil 2.11. STIX 2.1 ile oluşturulan örnek bir STİ veri yapısı [44]

2.4.2. TAXII siber tehdit istihbaratı veri paylaşım standardı

Güvenilir Otomatik Gösterge Bilgisi Değişim (TAXII) [35] standardı siber tehdit verilerinin otomatik olarak paylaşılabilmesini sağlayan bir standarttır. STİ verilerini Güvenli Üst Metin Aktarım Protokolü (Hyper Text Transfer Protocol Secure, HTTPS) protokolü üzerinden sunucular arasında paylaşılmasını sağlamaktadır. Protokol, servis ve mesaj paylaşımı için

bir dizi RESTful web servisi sunmaktadır. Bu web servisleri ile STIX veya farklı formatlardaki veriler taşınabilmektedir. TAXII standardı aşağıda belirtilen ve Şekil 2.12'de gösterilen yöntemlerle tehdit verilerinin paylaşılabilmesini desteklemektedir [45]:

- Koleksiyon paylaşımı yönteminde sunucuda veriler bulunmakta, istemciler bu sunucundan bilgileri almaktadır.
- Abonelik ile paylaşım yönteminde veriyi kullanmak isteyen uç noktalar kaynağa abone olmakta ve güncel bilgiler aboneler sunucular tarafından paylaşılmaktadır. Üretici olanlar da sunuculara istihbarat verisi göndermektedir. Bir istemci hem üretici hem de tüketici rolü ile TAXII protokolünü kullanabilmektedir.

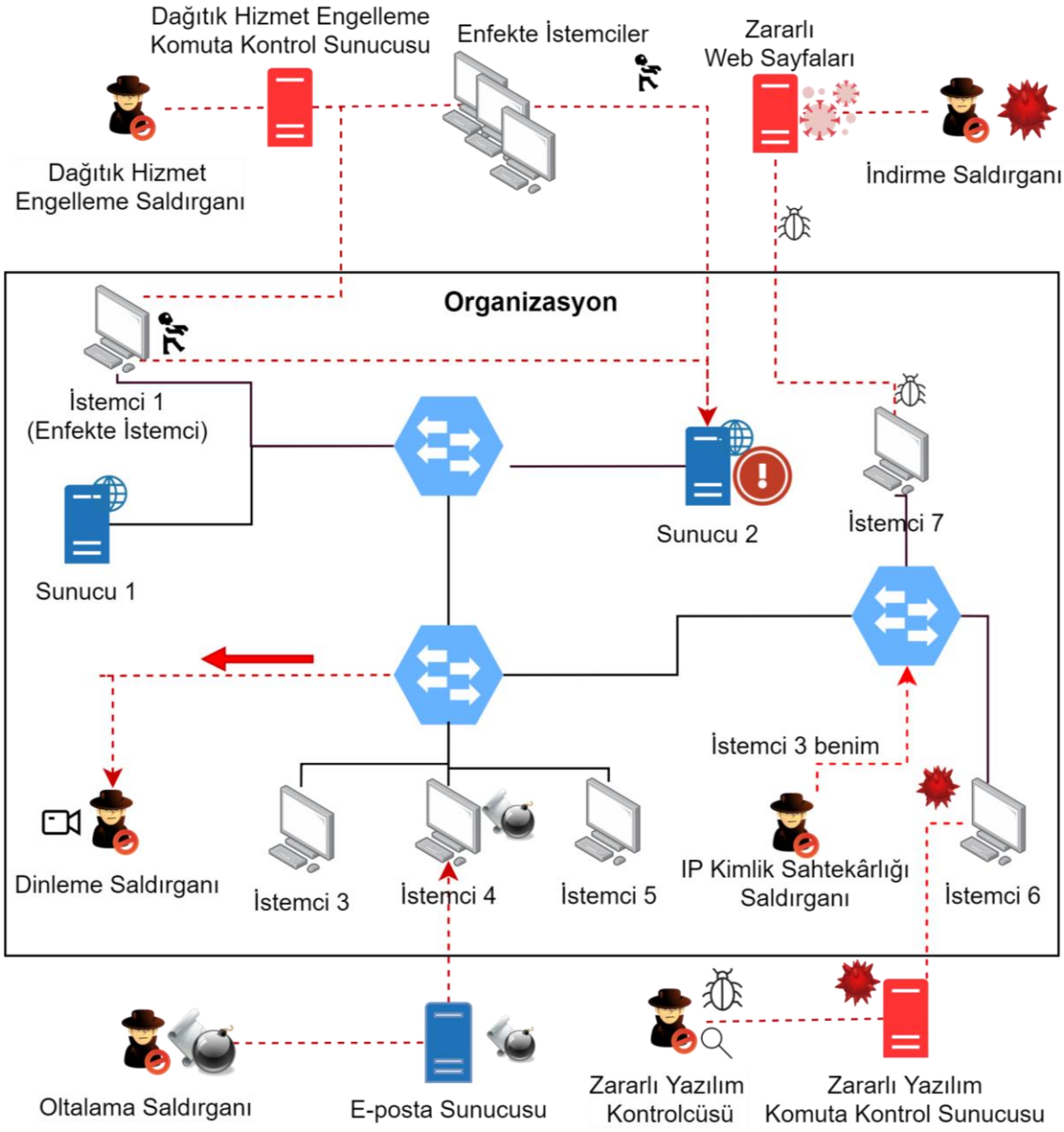


Şekil 2.12. TAXII protokolü ile STİ verisi paylaşım yöntemleri [44]

Bir TAXII sunucusu bir veya birden fazla kökü desteklemektedir. Sunuculardaki kökler TAXII koleksiyonları veya kanalları mantıksal olarak gruplandırır. TAXII protokolü ile iletişimde HTTPS kullanılır ve içerik el sıkışması ve kimlik doğrulama Üst Metin Aktarım Protokolü (Hyper Text Transfer Protocol, HTTP) başlığında yer alan bilgiler ile gerçekleştirilir.

2.5. Siber Tehdit İstihbaratı Odaklı Genel Saldırgan Modelleri

Bir tehdit aktörü, bir veya daha fazla saldırgan modeli olarak sınıflandırılabilir. STİ verisi ile paylaşılan ve ağ seviyesinde önleme ve etki azaltma faaliyetlerinin uygulanabildiği genel saldırgan modelleri Şekil 2.13'te gösterilmektedir.



Şekil 2.13. Siber tehdit istihbaratı ile paylaşılan genel saldırgan modelleri

2.5.1. Dağıtık hizmet engelleme (DDoS) saldırganı

DDoS saldırganları, zararlı yazılımlardan etkilenen çok sayıda başka istemcileri kullanarak sistemlerin kaynaklarını tüketir ve sistemlerin hizmet veremez hale gelmesine sebep olmaktadır. DDoS saldırganlarının kazancı, sistemlerin kullanılabilirliğini azaltmak veya farklı saldırılar başlatmak için sistemleri çevrim dışı bırakmaktır. En yaygın DDoS saldırıları; iletim kontrol protokolü senkronizasyon paketi (TCP SYN) sel saldırısı, gözyaşı saldırısı, şirin saldırısı (smurf attack), ölüm ping saldırısı ve botnet ağlarıdır.

DDoS saldırganı, hedef ağlara veya sistemlere saldırmak için ağ protokol [IP, TCP, İnternet kontrol mesajı protokolü (Internet Control Message Protocol, ICMP) vb.] zayıflıklarını kullanır. Saldırgan, kötücül yazılım bulaşmış milyonlarca bilgisayarı hedef sistemlere zararlı trafik göndermek için kontrol eder. Kötücül trafik, sistemlerin bant genişliğini veya işlem kapasitesini tüketir. DDoS saldırganı, sistem yönetimi hakkında ön bilgiye sahiptir. Saldırganın ağ protokolleri, zararlı yazılımlar ve siber saldırı süreçleri hakkında ileri düzeyde bilgiye sahip olduğu varsayılmıştır. Saldırgan, organizasyon ağının içinde olabilir ve organizasyonun sistemleri ve ağ mimarisi hakkında ayrıntılı bilgilere sahip olabilir.

2.5.2. İndirme (drive-by download) saldırganı

Kurbanlar web sayfalarını ziyaret ederken, bazı bağlantılara tıklar, komut dosyalarını çalıştırır ve bilgisayarlarına yazılım yükleyebilirler. İndirme saldırganı, güvenli olmayan web sitelerinin sayfalarına zararlı komut dosyaları ekler. Bu komut dosyaları, kötücül yazılımları doğrudan kurbanın bilgisayarına yükler veya kurbanı saldırgan tarafından kontrol edilen farklı bir web sitesine yönlendirir. Kötücül komut dosyaları, web sitelerini ziyaret ederken, e-posta mesajlarını görüntülerken veya bir açılır pencere tetiklendiğinde çalıştırılabilir. Web tarayıcılarının daha fazla eklentisi varsa, indirme saldırganları eklentilerin güvenlik açıklarını kullanarak da bilgisayarlardan daha fazla yararlanma fırsatına sahip olabilir.

İndirme saldırganlarının, kurbanların veya organizasyonun yönetimindeki bilgi teknolojileri altyapısı hakkında ön bilgiye sahip olmasına gerek yoktur. İndirme saldırganının web teknolojileri, işletim sistemleri, zararlı yazılımlar ve siber saldırı süreçleri hakkında ileri düzey teknik bilgisi olduğu varsayılmalıdır.

2.5.3. Zararlı yazılım kontrolcüsü

Zararlı yazılım programlarının veya komut dosyalarının çoğu, ağda yayılır ve uzaktan zararlı komuta kontrol (K&K) sunucularıyla iletişim kurar. Zararlı yazılımlar; arka kapı açmak, dosyaları silmek, DDoS saldırılarında bir ajan olarak hareket etmek, diğer zararlı yazılımları indirmek, verileri şifreleyerek fidye istemek, casusluk yapmak gibi birçok amaçla kullanılabilir.

Zararlı yazılım kontrolcüsü, zararlı yazılımları hazırlar ve bunları disk, web sayfası, e-posta gibi yollarla kurbanlara gönderir. Kurbanın bilgisayarına virüs bulaşmışsa, zararlı yazılım kendisini diğer istemcilere yaymaya çalışır. Zararlı yazılım, K&K merkezine bağlanır, K&K sunucusundan aldığı komutları yürütür ve bazı bilgileri (şifreleme anahtarı, gizli veriler, kişisel bilgiler vb.) ele geçirerek K&K sunucularına gönderir. Saldırgan, kuruluşun ağ bilgilerini biliyor olabilir. Zararlı yazılım kontrolcüsünün ağ protokolleri, istismar ve siber saldırı süreçleri hakkında ileri düzeyde bilgiye sahip olduğu varsayılmalıdır.

2.5.4. Oltalama saldırganı

Oltalama (phishing), güvenilen kişi veya kuruluştan geliyormuş gibi görünen e-postalar gönderme faaliyetidir. Bu e-postaların amacı, kişisel bilgileri elde etmek veya kötü niyetli faaliyetler yapmak için kurbanın bilgisayarına zararlı yazılım yüklemektir. Oltalama, sosyal mühendislik ve siber saldırı araçlarının birleşimidir. Oltalama e-postaları, bilgisayara zararlı yazılım yüklemek için eklenmiş bir eklenti, kişisel bilgileri toplamak için web sayfası yönlendirmesi veya zararlı yazılım indirmek için bir bağlantı içerebilir.

Oltalama saldırganı, organizasyonun dışarısından kurbanları aldatmak için bir e-posta veya gerçek bir web sitesinin benzerini hazırlar. Saldırgan, personelin iletişim adreslerini bilir. Saldırganın organizasyonun ağ bilgilerini bilmesine gerek yoktur. Oltalama saldırganının web teknolojileri, e-posta protokolleri, işletim sistemleri, zararlı yazılım ve siber saldırı süreçleri hakkında ileri düzeyde bilgiye sahip olduğu varsayılmalıdır. Saldırgan, personelin en çok kullanıldığı veya ilgisini çekebilecek web sitelerini bildiği kabul edilmelidir.

2.5.5. Dinleme saldırganı

Ağ dinleme (sniffing / eavesdropping) saldırısı, ağ iletişim güvenliğine yönelik en yaygın saldırılardan biridir ve ağların statik yapısı bu saldırıyı kolaylaştırır. Saldırgan, ağdaki iletişim bağlantısını dinleyebilir, ağ durumunu izleyebilir ve hassas verileri çalabilir. Dinleme saldırganı genellikle organizasyonun içinden saldıran veya organizasyonun içindeki bir bilgisayarı ele geçirmiş kişidir. Saldırgan pasif durumdayken, ağa herhangi bir paket göndermez ve mesajlara yanıt vermez. Saldırgan aktif duruma geçtiğinde, paketleri K&K sunucusuna veya diğer sunuculara gönderilebilir. Ağ trafiğini dinlemek için basit veya gelişmiş ağ dinleme araçları kullanılabilir Ağ trafiğindeki veriler şifrelenmemişse, bağlantı

verileri çıkarılarak kaydedilebilir. Ağ verileri şifrelenmişse [HTTPS, güvenli dosya transfer protokolü (File Transfer Protocol Secure, FTPS) vb.], saldırgan daha sonra analiz etmek için paketleri dosyalara depolayabilir.

Saldırganın ağ protokolleri, ağ sistemleri hakkında ileri düzeyde bilgiye sahip olduğu varsayılmalıdır. Saldırganın, organizasyonun bilgi teknolojileri (BT) altyapısını biliyor olabileceği veya tarama saldırısı ile öğrenebileceği kabul edilmelidir. Saldırgan, organizasyonun istemci sayılarını, aktif IP adreslerini ve çalışan servislerini ağı dinleyerek keşfedebilir.

2.5.6. IP sahtekârlığı saldırganı

IP sahtekârlığı, bir tür ortadaki adam saldırısıdır. Saldırgan, IP sahtekârlığı ile bir sistemi aldatır ve saldırganın isteğine güvenildiğine ve isteğin sisteme erişim iznine sahip olduğuna ikna eder. Saldırgan trafiği dinler ve bilinen istemcinin kaynak IP adresini içeren bir paket gönderir ve sistem yanıtlarını bekler. Ayrıca, IP sahtekârlığı saldırganı, ağ mimarisini tanımak ve gerçek IP aralıklarını kullanarak servisleri bulmak için ağı tarayabilir. Saldırgan, organizasyonda ve ağ mimarisinde kullanılan servisleri ve olası IP aralıklarını önceden biliyor olabilir. Saldırganın, ağ protokolleri ve güvenlik sistemleri hakkında ileri düzeyde bilgiye sahip olduğu kabul edilmelidir.

2.6. Siber Tehdit İstihbaratı Odaklı Tehdit Kategorileri

Siber tehdit, herhangi bir bilgisayar ağına, BT (bilgisayar, cep telefonu vb.) veya operasyonel teknoloji (Operational Technology, OT) cihazlarına [programlanabilir mantık kontrolcüsü (Programmable Logic Controller, PLC), denetleyici kontrol ve veri toplama sistemleri (Supervisory Control and Data Acquisition Systems, SCADA) vb.] erişmek ve bunları kontrol etmek için gerçekleştirilebilecek zararlı eylemdir. Her gün yeni siber tehdit vektörleri ortaya çıkmakta ve bu durum yeni siber tehdit türleriyle mücadele etmek için geliştirilen önleyici ve savunma eylemlerini daha karmaşık hale getirmektedir. Saldırganların saldırılarını gerçekleştirmek için kullandıkları davranışlar ve kaynaklar, STIX v1.2[42] standardında; TTP olarak tanımlanır. Aynı kavram STIX v2.1 [11] standardında TTP yerine farklı bir şekilde temsil edilir. Bu standartta TTP verisi, saldırı örüntüsü, zararlı yazılım ve araç türlerine ayrılmıştır. Saldırı örüntüsü, bir saldırı türünün nasıl yürütüldüğünü

tarif etmek için kullanılır [14] bunun yanı sıra mevcut veya yeni saldırı vektörlerini açıklamak için yararlıdır. CAPEC [15], saldırıları açıklamak ve tehditler hakkında bilgi paylaşmak için sınıflandırılmış, genel kullanıma açık bir ortak saldırı örüntü kataloğudur. Ayrıca, MITRE ATT&CK [16] kütüphanesi, kurumsal ağlara karşı gelişmiş kalıcı tehditlerin kullandığı ortak taktikleri, teknikleri ve prosedürleri barındırmaktadır.

Ağ ve bilgisayar güvenliği taksonomileri siber güvenlik tehditlerinin sınıflandırmasına yardımcı olur [46]. Bu taksonomileri oluşturmak ve sunmak için farklı yaklaşımlar vardır. Literatürdeki çalışmalarda siber tehdit kategorizasyonu ve taksonomi için farklı yaklaşımlar ve farklı terminolojiler kullanılmıştır [46–51]. Hansman ve Hunt [47] ile Lough [51] çalışmalarında siber güvenlik taksonomilerini saldırılardan ziyade güvenlik açıklarına dayandırmışlar ve 1975'ten 2001'e kadar olan süreci tarihsel olarak açıklamışlardır. Jin ve diğerleri [48] 1976-2008 yıllarında yayınlanan sınıflandırma çalışmalarını gözden geçirmişlerdir. Canbek ve diğerleri [46], 1993'ten 2015'e kadar olan siber güvenlik ile ilgili saldırıları, olayları, zararlı yazılımları, tehdit ve güvenlik açıklarını dikkate alan 28 farklı güvenlik sınıflandırma çalışmalarını analiz etmiştir.

En yaygın ağ saldırıları; web saldırıları, keşif saldırıları, hizmet engelleme (DoS) saldırıları, servislere özel saldırılar ve bilinen kötü kaynaklar olarak belirtilmiştir [52]. Yapılan bir anket araştırmasında [53], organizasyonlarda en çok gözlenen tehditlerin; oltalama saldırıları (katılımcıların %72'si), casus yazılımlar (katılımcıların %50'si), fidye yazılımları (katılımcıların %49'u) ve truva atları (katılımcıların %47'si) olduğu tespit edilmiştir. Aynı raporda, organizasyonları en çok etkileyen saldırıların sırasıyla oltalama, DDoS ve APT saldırıları olduğu belirtilmiştir.

Yaygın siber tehditlerin yanında yeni nesil tehditler de ortaya çıkmıştır. Yeni nesil tehditlerin başında bilgi çalmak için yürütülen APT tehditleri ile birlikle kendilerini sürekli değiştiren polimorfik tehditler, sıfırıncı gün tehditleri ve birden fazla tehdidin bir arada kullanıldığı birleşik saldırılar gelmektedir [39].

Literatürdeki siber saldırı sınıflandırma çalışmaları Çizelge 2.1'de gösterilmiştir. Çizelgedeki siber saldırıların ve saldırıların sınıflandırmalarının farklı bakış açıları ile kategorize edildiği görülmektedir. Sınıflandırmalar incelendiğinde aşağıdaki çıkarımlara ulaşılabilir:

- Saldırı sınıflandırması 1975'ten beri aktif bir araştırma alanıdır ve siber saldırı adları üzerinde fikir birliği yoktur.
- Siber saldırı kategorilerinin ayrıntıları farklılık göstermektedir.
- DoS, tarama, kimlik sahtekârlığı gibi çoğu saldırı tipi çalışmalarda yaygın olarak yer almaktadır.
- Aynı saldırıların kategorileri, araştırmanın odağına bağlı olarak çalışmalar arasında değişebilmektedir.

Çizelge 2.1. Literatürdeki siber saldırı sınıflandırmaları

Çalışmanın Odağı	Siber Saldırı Örnekleri
Saldırı vektörleri, saldırı hedefleri ve güvenlik açıkları [47]	Virüs, solucan, truva atı, tampon taşması, hizmet engelleme (DoS) saldırısı, kimlik sahtekârlığı, oturum ele geçirme, kablosuz ağ saldırısı, web uygulama saldırısı, fiziksel saldırı, parola saldırısı, bilgi toplama saldırısı, karışık saldırı
Saldırı vektörleri, hedefleri ve etkileri [49]	Yanlış yapılandırma, çekirdek kusurları, tasarım hataları, tampon taşmaları, yetersiz kimlik doğrulama, yetersiz girdi doğrulama, sembolik bağlantı, dosya tanıtıcı saldırısı, yarış durumu, yanlış yetkilendirme, sosyal mühendislik
Uç birim tehdit kategorileri [53]	Oltalama, fidye yazılımı, DDoS, APT, yetki yükseltme, truva atı, web uygulama saldırısı, karma tehdit, casus yazılım, rootkit, ortadaki adam saldırısı, zincirleme sömürüler, tuş kaydedici, zararlı yazılım, çekirdek modu istismarları
Uç birim tehdit kategorileri [54]	• Hedefli saldırılar (casusluk, yıkma, sabotaj), • E-posta saldırıları (oltalama, zararlı yazılım, istenmeyen e-posta), • Web saldırıları, • Fidye yazılımları
Karşılıklarına göre ağ ve bilgisayar saldırılarının sınıflandırması [55]	• Enfeksiyon aşaması (virüs, solucan, truva atı), • Patlatma (tampon taşması), • Araştırma (dinleme, port haritalama, güvenlik taraması), • Kandırma (IP sahteciliği, Ortam Erişim Kontrolü (Media Access Control, MAC) sahteciliği, alan adı sistemi (Domain Name System, DNS) sahteciliği, oturum ele geçirme, çapraz site komut dosyası oluşturma, gizli alan işlemleri, giriş parametresi hilesi), • Deneme (kaba kuvvet, sözlük saldırısı, kapı tokmağı saldırısı), • Eş zamanlılık (taşkın, DDoS)

Çizelge 2.1. (devam) Literatürdeki siber saldırı sınıflandırmaları

Çalışmanın Odağı	Siber Saldırı Örnekleri
Operasyon yöntemleri [56]	Operasyonel etki (kaynakların kötüye kullanılması, kullanıcı güvenliği, web güvenliği, zararlı yazılım, DoS)
Dağıtık sistemler için örüntü temelli güvenlik tedbirleri [57]	• Ağ iletişim saldırıları (pasif ve aktif gizlice dinleme, kaynak sahteciliği, protokol dinleme, gizli ağ kanalı, oturum ele geçirme), • Yasal olmayan veri iletme (enjeksiyon), • Uzaktan bilgi çıkarma (tarama, bilgi açıklama, veri çıkarımı), • Kontrolsüz işlemler (yetkisiz erişim, yetkili işlemleri taklit etme, işlem taşması)
Ağ güvenlik araçları sınıflandırması [58]	Truva atları, DoS, DDoS, paket oluşturma saldırıları, tarayıcı saldırıları (XSS istek sahteciliği), sunucu saldırıları (protokol saldırısı, yapısal sorgu dili (SQL) enjeksiyonu, kod yerleştirme, tampon hatası, birörnek kaynak bulucu (uniform resource locator, URL) yanlış yorumlama), parmak izi saldırısı, kullanıcı saldırısı (kullanıcıdan uzağa, uzaktan yerele), dinleme, ağ tarama
Yönlendirme sistemi saldırı tespiti [59]	• Saldırı katmanı (dinleme, trafik taşması, kimlik sahtekârlığı, MAC adres tablosu taşması, yönlendirme sızdırma, DoS, yönetim protokolü saldırısı), • Saldırı vektörü (virüs, solucan, tampon taşması, fiziksel saldırı, şifre saldırı, bilgi toplama)

Çizelge 2.1'deki çalışmalar STİ veri kullanımı ile sınırlı değildir. Genel olarak, siber saldırılar ağlarda veya uç birimlerde (sunucu, bilgisayar, IoT cihazı vb.) gerçekleştirilebilir. Bu tez çalışmasında, STİ verilerinde yer alabilen siber tehditlere ve bir ağ üzerinde yapılabilecek eylemlere odaklanmaktadır. STİ verilerinde yer alan bilgiler çok çeşitlidir. Bu nedenle siber tehditleri organize etmek için bir sınıflandırma oluşturmak önemlidir. Literatürdeki çoğu çalışma belirli siber tehditlere odaklanmakta ve bunlara karşı savunma çözümleri önermektedir.

Çizelge 2.2'de, STİ verilerinde yer alan yaygın tehditlerin bir listesi sunulmuştur. Bu çizelgede ayrıca saldırıların aralarındaki ilişkilere ve literatürdeki çalışmalara dayanarak kategoriler oluşturulmuştur.

Çizelge 2.2. Bu çalışmada kullanılan siber tehdit kategorileri

Kategori	Siber Tehditler
Tarama Saldırıları	Ağ tarama, derinlemesine araştırma / parmak izi saldırısı
Kimlik Sahtekârlığı Saldırıları	IP adresi kimlik sahtekârlığı, adres çözümleme protokolü (address resolution protocol, ARP) kimlik sahtekârlığı, DNS kandırma, medya erişim kontrolü (MAC) adres tablosu taşması, yönlendirme sahtekârlığı, ağ yönetim protokolü saldırısı, kablosuz ağ saldırısı, güvenli soket katmanı (secure sockets layer, SSL) / taşıma katmanı güvenliği (transport layer security, TLS) saldırısı, oturum açma [47, 55, 57, 58, 60]
Ağ Seviyesi DoS ve DDoS Saldırıları	Kullanıcı veri bloğu protokolü (user datagram protocol, UDP) sel, SYN sel, ölüm pingi, gözyaşı saldırısı, düşük oranlı DoS saldırısı, ICMP sel, DNS saldırısı, trafik taşması, DDoS, UDP taşması, SYN taşması, gözyaşı saldırısı, düşük hızlı DoS saldırısı, ICMP taşması, DNS saldırısı, trafik taşması [47, 49, 53, 55, 57–60]
Dinleme Saldırıları	Kimlik bilgileri dinleme, bilgi toplama saldırısı, bilgi ifşa saldırısı, gizli ağ kanalı, kimlik bilgisi güvenliği, gizlice dinleme, casus yazılım [47, 54, 55, 57–59]
Zararlı Yazılım ve Sosyal Mühendislik Saldırıları	Fidye yazılımı, solucan, truva atı, reklam yazılımı, tuş kaydedici, casus yazılım, virüs, zararlı komut dosyaları, tarayıcı saldırıları, istenmeyen e-posta, oltalama, URL yanlış yorumlama saldırısı [47, 49, 61, 53–60]
Web Uygulama Saldırıları	Uygulama düzeyinde DoS saldırısı, XSS, geçersiz giriş parametresi saldırısı, enjeksiyon saldırısı (SQL, komut, basit indeks erişim protokolü enjeksiyonu), siteler arası istek sahteciliği (Cross-Site Request Forgery, CSRF), kimlik doğrulaması katyon ve oturum yönetimi saldırıları, oturum ele geçirme, yanlış yapılandırma istismarları, kaba kuvvet saldırıları, uygulamanın kötüye kullanımı [47, 49, 53, 54, 56–58, 61]
Diğer Saldırılar (Donanım, İşletim Sistemi ve İşlem Saldırıları)	Bellek tabanlı saldırılar, ayrıcalık yükseltme, tampon taşması, çekirdek modu açıkları, rootkit, proses istismarları, donanım arka kapısı, APT, karma tehditler, zincirleme istismarlar, şifre saldırısı [47, 49, 61, 53–60]

Bu tezde kullanılan ve Çizelge 2.2'de listelenen tehdit kategorileri, STİ veri modellerine ve ağ savunmasına göre Çizelge 2.1'de gösterilen saldırılardan türetilmiştir. Çalışma sonucunda çok daha geniş kapsamlı tehdit türü sunulmuş ve bu siber tehdit türlerine karşı önleme ve / veya etki azaltma yöntemleri öneren YTA temelli siber savunma çalışmaları incelenerek tezin 3. bölümünde ayrıntılı biçimde tartışılmıştır.

Çizelge 2.2'deki tehdit kategorilerinden *Diğer Saldırılar* kategorisi altında yer alan saldırılar istemci seviyesinde gerçekleştirildiği için önlemlerin de istemci düzeyinde alınması gerekmektedir. Bu sebeple tez çalışması kapsamında bu kategoriye odaklanılmamıştır. Ayrıca zararlı yazılım ve sosyal mühendislik saldırıları kategorisindeki saldırılar ile web uygulama saldırıları ağırlıklı olarak istemci üzerinde gerçekleştirilse de ağ savunması ile bu saldırıların önlenebileceği veya etkilerinin azaltılabileceği görülmüştür. Literatür taraması çalışmalarında bu iki kategori birleştirilerek 3. Bölümde tek başlık altında incelenmiştir. Gelişmiş kalıcı tehditler ve büyük ölçekli DDoS saldırıları gibi bazı karmaşık saldırıların çizelgede birden çok kategoride yer alabilmesi mümkündür fakat bu tehditler en çok ilgili olduğu tehdit kategorisi altında ele alınmıştır.

3. LİTERATÜRDEKİ YTA TEMELLİ SAVUNMA ÇALIŞMALARI

Literatürde YTA temelli ağ güvenliği hakkında çeşitli derlemeler [20, 62–67] yapılmıştır (Çizelge 3.1). Bu çalışmalar hem YTA temelli ağ güvenliği hem de YTA mimarisinin güvenliği ile ilgili çalışmaları gözden geçirmiştir.

Çizelge 3.1. Literatürde YTA temelli savunma çözümleri için oluşturulan taksonomiler

Çalışma	İnceleme Taksonomisi	İncelenen Kategoriler
Scott-Hayward ve diğerleri [20]	Güvenlik iyileştirmeleri	Toplama - algılama - dinleme, trafik analizi ve kural güncelleme, DoS / DDoS koruması, orta kutu mimarileri ve hizmetleri, kimlik doğrulama - yetkilendirme - denetim, güvenli - ölçeklenebilir çoklu kiracılık
Alsmadi ve Xu [62]	Güvenlik kontrolleri	Güvenlik duvarları, erişim kontrolü, saldırı tespit sistemi (STS) / saldırı önleme sistemi (SÖS), politika yönetimi, izleme ve denetim, mobil güvenlik kontrolü, kablosuz ağların güvenliği, gizlilik koruması, kendi cihazını getir (bring your own device, BYOD) güvenlik kontrolleri, açık laboratuvarların güvenlik kontrolü
Ahmad ve diğerleri [63]	Güvenlik tipi ve Güvenlik gereksinimleri	Akış örnekleme, paket örnekleme, SÖS, güvenlik duvarı, akış tabanlı STS, ağ esnekliği, güvenlik izleme, ağ izleme, akış kuralı doğrulaması, yapılandırma doğrulaması, denetleyici kullanılabilirliği Erişim denetimi, kimlik doğrulama, inkâr edilemezlik, veri gizliliği, iletişim güvenliği, veri bütünlüğü, kullanılabilirlik, mahremiyet
Shaghaghi ve diğerleri [64]	YTA temelli güvenlik hizmetleri	Saldırı önleme sistemleri, gizliliği artırıcı hizmetler, güvenlik orta kutuları, bulut hizmetlerini koruma, güvenli veri boşaltma, IoT güvenliği ve diğerleri
Rawat ve Reddy [65]	YTA güvenlik çözümleri	STS / SÖS olarak YTA, anormal durum tespiti için YTA, DDoS atak tespiti ve önlenmesi

Çizelge 3.1. (devam) Literatürde YTA temelli savunma çözümleri için oluşturulan taksonomiler

Çalışma	İnceleme Taksonomisi	İncelenen Kategoriler
Chica ve diğerleri [66]	Ağ güvenliği	Tehdit algılama, saldırı düzeltme, kimlik ve erişim yönetimi, ağ durumu izleme ve analizi
Farris ve diğerleri [67]	YTA temelli güvenlik özellikleri	Trafik izolasyonu, merkezi görünürlük yoluyla güvenlik ağı izleme, dinamik akış kontrolü, istemci ve yönlendirme gizlemesi, güvenlik ağı programlanabilirliği
Swami ve diğerleri [68]	YTA ile DDoS savunması	İstatistiksel / politika tabanlı savunma, makine öğrenmeye dayalı savunma, uygulamaya özgü savunma
Bu tez çalışması	Siber tehdit kategorileri	Tarama saldırıları, IP sahtekârlığı saldırıları, ağ düzeyinde hizmet engelleme (DoS) saldırıları, dinleme saldırıları, zararlı yazılımlar ve sosyal mühendislik saldırıları, web uygulama saldırıları

Çizelge 3.1'de verilen her çalışma farklı bir taksonomi kullanarak YTA temelli güvenlik çözümlerini sınıflandırmış ve buna göre literatürü incelemiştir. Bu sınıflandırmaların ana kategorileri genellikle çözüme yönelik veya güvenlik gereksinimlerine yöneliktir. Sadece Swami ve diğerleri [68] tehdit odaklı bir sınıflandırma oluşturmuş ve sadece DDoS saldırılarına karşı YTA temelli savunma çözümlerini üç tehdit kategorisine göre (sel, güçlendirme ve uygulama katmanı saldırıları) incelemiştir. Bu tez çalışması kapsamında siber tehditler, STİ verileri ile paylaşılabilen tehdit kategorilerine göre gruplandırılarak literatür taraması yapılmıştır.

Literatürde yer alan ilgili çalışmalar aşağıdaki dört alt başlıkta incelenmiştir. Öncelikle bu tez kapsamında önerilen sisteme benzer hedeflere sahip YTA temelli siber savunma mimarileri ve çerçeveleri değerlendirilmiştir. Sonrasında YTA temelli siber savunma yaklaşımları detaylı olarak analiz edilmiştir. Üçüncü başlıkta tez çalışmasında sunulan YTA temelli ağ savunma servisleri ile ilgili çalışmalara yer verilmiş ve tez çalışmasındaki servislerle karşılaştırılmıştır. Son başlıkta da STİ verilerini kullanan YTA temelli ağ güvenliği önerileri bu çalışma ile karşılaştırılarak incelenmiştir.

3.1. YTA Temelli Siber Savunma Mimarileri ve Çerçeveleri

Shin ve diğerleri [69], YTA kullanarak güvenlik uygulamaları geliştirmeyi sağlayan FRESCO adlı bir çerçeve önermiştir. FRESCO içinde 16 yeniden kullanılabilir bileşen mevcuttur. Bu bileşenler ile farklı güvenlik uygulamaları oluşturulabilmektedir. Uygulamalar, FRESCO betik dilini kullanarak; akışlar için kısıtlamaya dayalı düşürme, iletme, yönlendirme, yansıtma, ağ ayırma gibi eylemleri geliştirebilir. Çerçevede yer alan Güvenlik Uygulama Çekirdeği bileşeni, güvenlikle ilgili uygulamaların yürütülmesini sağlar. FRESCO, daha az kod satırı ile popüler güvenlik işlevlerinin hızlı geliştirilmesini sağlar. Bileşenler bir araya getirilerek güvenlik hizmeti bileşimi oluşturulabilir. Her bileşen olay odaklı olarak geliştirilir ve bazı ara yüzlerin (giriş, çıkış, parametre, eylem ve olay) tanımlanması gerekir. Bileşenlerin yapabileceği eylemler; düşürme, yönlendirme veya karantina olabilir. Çalışmada; tarama algılama, ağ düzeyinde botnet analizi ve zararlı yazılım algılama servisleri örnek olarak uygulanmıştır. Bu tez çalışmasında önerilen sistemde FRESCO'dan farklı olarak hiçbir betik dili kullanılmamaktadır. Servisler, OpenFlow asenkron mesajlarına entegre edilir ve harici kaynaklar (STS, STİ Yöneticisi vb.) tarafından tetiklenir. Ek olarak, savunma servislerinin eylemleri düşürme, yeniden yönlendirme veya karantina ile sınırlı değildir, güvenlik eylemleri ile OpenFlow tarafından desteklenen herhangi bir akış veya ölçüm güncelleme eylemi oluşturabilir.

Reich ve diğerleri [70], YTA programlama için Pyretic dili kullanımını önermiştir. Pyretic dili, düşük seviyeli ayrıntıları gizler ve programcıların politikaları kolayca geliştirmesine olanak tanır. Pyretic sadece güvenlik işlemlerine özel olmasa da, rutin güvenlik görevleri için hazırlanan tüm komut dosyalarını yorumlamak ve OpenFlow mesajlarına dönüştürmek için kullanılmaktadır.

Dhawan ve diğerleri [71], güvenlik ilkesi motoruna sahip olan, ayrıntılı güvenlik ilkelerini oluşturmaya ve kolay eylem ilişkilendirmesine olanak tanıyan SPHINX'i önermiştir. Yüksek seviyeli diller ve politika motorları ile bir olay veya zamanlayıcı tetiklendiğinde akış kuralları oluşturmak gibi gelişmiş güvenlik işlevlerinin gerçeklemesi karmaşık olabilir. Tez çalışmasında önerilen sistemin, SPHINX rutin güvenlik işlevleri olarak eklenebileceği değerlendirilmektedir.

Chowdhary ve diğerleri [72], uygulama katmanında politika bileşimini, akış kuralı çakışma algılamasını ve kontrol katmanında çözümlemeyi yöneten nesne yönelimli çok katmanlı bir ağ güvenlik mimarisi olan YTA Güvenlik Operasyon Merkezini (SDN Security Operation Center, SDNSOC) sunmuştur. Önerilen çerçeve kullanılarak SFZ uygulama ayrıntıları soyutlanmış ve farklı ağ fonksiyonlarının birleştirilmesi otomatikleştirilmiştir. Tez kapsamında ağ düzeyinde savunma servislerine ve bu servislerin bileşimine odaklanılmıştır. Bu tezde önerilen mimaride, ağ seviyesi servis fonksiyonlarına ek olarak, SFZ kullanılarak farklı sanal ağ fonksiyonları kullanılabilir.

Zarca ve diğerleri [73], siber fiziksel sistemler ve IoT özellikli kritik altyapılar ile ilgili güvenlik ve gizlilik problemlerini dikkate alan ANASTACIA adlı bir mimariyi açıklamıştır. IoT güvenliğini artırmak için YTA ve AFS'ye dayalı bütünsel bir siber güvenlik çerçevesi sunmuştur. Önerilen çerçeve, güvenlik düzenleme, güvenlik uygulama, izleme ve reaksiyon katmanlarına sahiptir. Önerilen çerçeve IoT veya Beşinci Nesil Teknoloji (Fifth Generation Technology**,** 5G) altyapıları için kullanılabilecek AFS yönetimi ve orkestrasyonu mimarisine dayanmaktadır. Tez çalışması kapsamında sunulan servislerin, YTA kontrolcüsüne bir eklenti olarak eklenebileceği veya ANASTACIA'ya entegre edilebilen savunma servisleri olarak kullanılabileceği değerlendirilmektedir.

Karmakar ve diğerleri [74], politika tabanlı bir güvenlik uygulama mimarisi sunmuştur. Sunulan uygulama; politika yöneticisi, politika uygulayıcısı ve değerlendirme motoru bileşenlerine sahiptir. Mimari ayrıca, güvenlik yönlendirme ve izinsiz giriş tespit politikalarını depolar ve yönetir. Mimari, sadece belirli ağ saldırılarından (IP sahtekârlığı saldırıları, sel saldırıları vb.) YTA altyapısını korumak için tasarlanmıştır.

3.2. YTA Temelli Siber Savunma Yaklaşımları

Bu bölümde, tez çalışması kapsamında belirlenen tehdit kategorilerine (Bkz. Çizelge 2.2) göre literatürdeki YTA temelli saldırı önleme ve etki azaltma yöntemleri, teknikleri ve prosedürleri incelenmiştir. Literatürden çalışma seçerken, aşağıda listelenen kriterler dikkate alınmıştır:

- İncelenecek çalışma, YTA kullanarak ağları savunmak için bir yöntem, teknik veya prosedür önermelidir. Derleme makaleleri hariç tutulmuştur.

- İncelenecek çalışma, Çizelge 2.2'de listelenen en az bir siber tehdit kategorisi için bir önleme ve / veya etki azaltma yöntemi önermelidir. Tez çalışmasının amacı, ağları proaktif olarak korumak ve tehditlere çevik bir şekilde yanıt vermek olduğu için sadece siber saldırı tespit yaklaşımları öneren çalışmalar araştırma dışında bırakılmıştır. Fakat, saldırı önleme ve etki azaltma ile birlikte tespit yaklaşımları sunan çalışmalar tez çalışmasında inceleme kapsamı dışında bırakılmıştır.
- YTA altyapısına özgü saldırılara odaklanan ve bunlar için çözüm sunan çalışmalar inceleme kapsamı dışında tutulmuştur.
- Çizelge 2.2'deki Diğer Saldırılar kategorisindeki siber tehditlere karşı çözümler çoğunlukla ağ seviyesinde olmadığı için bu kapsamda inceleme yapılmamıştır.

Takip eden her alt bölüm, tez kapsamında belirlenen tehdit kategorinden birine odaklanmış ve literatürdeki çalışmalar bu alt bölümlerde savunma türüne, uygulanan YTA eylemlerine ve kurulum konumlarına / katmanlarına göre sınıflandırılmıştır. İncelenen çalışma birden fazla siber tehdit kategorisine odaklanıyorsa, yalnızca en ilgili olduğu alt bölümde ele alınmıştır.

Tehdit kategorilerine göre YTA temelli savunma yöntemleri için her alt bölümün sonunda bir özet tablosu sunulmuştur. Bu tablolarda çalışmanın adı, savunma türü, açıklama, yaklaşım ve kurulum sütunları yer almakta ve değerlendirilen literatür çalışmaları bu özelliklere göre özetlenmektedir. Tablolarda yer alan sütunlar ve sütunların açıklamaları aşağıda gösterilmiştir:

- *Çalışmanın adı*: İncelenen çalışmada önerilen metodoloji, teknik veya prosedürün adı ile çalışmanın referansı bu sütunda verilmiştir. İncelenen çalışmada önerilen yaklaşım için bir addan bahsedilmediği durumlarda sütunda sadece çalışmanın referansı yer almaktadır.
- *Savunma türü*: İncelenen çalışmada önerilen yöntem, teknik veya prosedürler; önleme, tespit veya etki azaltma savunması olarak kategorize edilmiştir. Bir çalışmada birden fazla savunma türü yer alabilmektedir. Çalışmada savunma türlerinden hangisinin veya hangilerinin önerildiği sırası ile gösterilmektedir.
- *Açıklama*: Önerilen savunma mekanizmasının önemli hususları için bir özet bu sütunda sunulmaktadır.

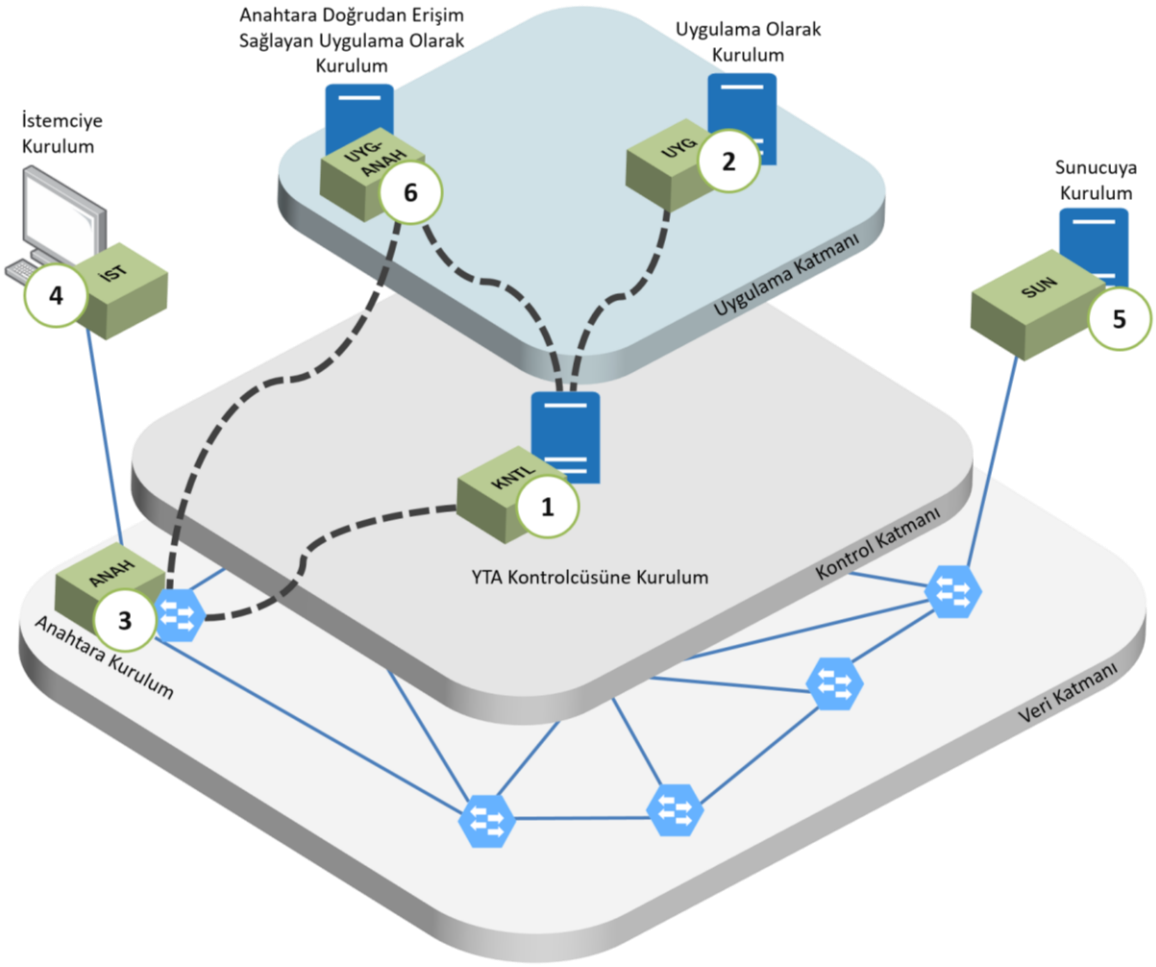
- *Yaklaşım*: YTA tabanlı savunma yaklaşımları Çizelge 3.2'de üç ana kategori altında sınıflandırılmıştır. Çalışmada önerilen çözümde yer alan yaklaşımlar, Çizelge 3.2'de listelenen tüm savunma yaklaşımlarına göre sınıflandırılmış olarak bu sütunda gösterilmektedir. Trafik engellemek için; paketlerin düşürülmesi, kara liste yönetimi ve ağın ayrıştırılması yaklaşımları tespit edilmiştir. Ağ ve veri koruma kapsamında ağ erişim kontrolü, trafik hızı sınırlama, yönlendirme, çoklu rota yönlendirmesi, rastgele tuzak sistem yerleştirme ve paket yükünün şifrelenmesi yaklaşımları yer almaktadır. Sanal görünüm oluşturma, paket başlığının değiştirilmesi, IP adresinin değiştirilmesi, paket yükünün güncellenmesi ve tuzak sisteme gönderme yaklaşımları aldatma kategorisinde yer almaktadır.
- *Kurulum*: Çoğu çözüm, mimaride yeni bileşenler veya mevcut bileşenlerde değişiklikler önermektedir. Önerilen çözümlerin bileşenleri Şekil 3.1'de gösterildiği şekilde; YTA kontrolcüsüne kurulum (KNTL), uygulama olarak kurulum (UYG), YTA anahtarlarına kurulum (ANAH), ağdaki istemcilere kurulum (İST), ağ sunucularına kurulum (SUN), YTA anahtarlarına doğrudan erişim sağlayacak uygulama olarak kurulum (UYG-ANAH) seçeneklerine göre gruplandırılmaktadır.

İncelenen çalışmalar, siber savunma operasyonlarında farklı savunma stratejilerini takip etmektedir. Çalışmalarda önerilen savunma stratejileri aşağıda gruplandırılmıştır:

- *Politika tabanlı savunma stratejisi*: YTA tabanlı ağlarda, sistem özelliklerini ve ağ istatistiklerini kullanarak güvenlik politikalarını dinamik olarak belirlemek mümkündür. Çoğu güvenlik mekanizmasının statik güvenlik stratejileri kullanılarak yönetildiği geleneksel ağlara göre YTA tabanlı ağlar üzerinde bu daha kolay yapılabilmektedir.
- *Makine öğrenmeye dayalı savunma stratejisi*: Bu strateji ile; saldırıları tespit etmek ve tehditlere yanıt vermek için makine öğrenmesi veya yapay zeka yöntemleri kullanılarak güvenlik kuralları oluşturulmaktadır.
- *Hareketli hedef savunma stratejisi*: Saldırı yüzeyinin rakipler tarafından öngörülememesi için bir veya daha fazla ağ sistemi özelliği değiştirilerek saldırgan kandırılmaya çalışılmakta veya daha az bilgi elde etmesi sağlanmaktadır.
- *İş birliğine dayalı / dağıtık savunma stratejisi*: Birden çok ağ etki alanı, siber tehdit bilgilerini iş birliği içinde paylaşmakta ve / veya saldırılara karşı ortak savunma mekanizmaları uygulamaktadır.

Çizelge 3.2. Ağ seviyesi siber savunma yaklaşımları

Kategori	Yaklaşım	Açıklama
Trafik Engelleme	DÜŞÜR	Paketi eşleşme kuralına göre düşürür
	KARA-LİSTE	Kaynak veya hedef adresi kara listede olan paketler düşürülür.
	AĞ-AYIR	Belirli bilgisayardan gelen ve giden tüm paketleri düşürülür (karantina) veya ağlar ayrılarak yönetilir.
Ağ ve Veri Koruma	AEK (Ağ Erişim Kontrolü)	Yetkisiz istemcilere veya akışlara izin verilmez, istemcilerin başlangıçta kimlik doğrulaması yapılır.
	HIZI-SINIRLA	Belirli ağ trafiğinin bant genişliği değiştirilir.
	YÖNLENDİR	Belirli ağ trafiğinin yönlendirme rotası değiştirilir.
	ÇOKLU-ROTA	Akışların parçaları farklı rotalardan gönderilir.
	RASTGELE-ALDATICI	Ağa rastgele bal küpü veya bal küpü ağı yerleştirilir.
	YÜKÜ-ŞİFRELE	Paket yükü şifrelenir.
Aldatma	SANAL-GÖRÜNÜM	Her istemci için farklı ağ görünümleri oluşturulur.
	BAŞLIĞI-DEĞİŞTİR	Paket başlıklarındaki bilgiler güncellenir.
	ADRESİ-DEĞİŞTİR	Paketlerdeki MAC adresleri, IP adresleri veya bağlantı noktası bilgileri değiştirilir.
	YÜKÜ-DEĞİŞTİR	Paketin yükünde yer alan bilgiler değiştirilir veya deforme edilir.
	ALDATICIYA-GÖNDER	Trafik bal küpüne veya bal küpü ağına yönlendirilir.



Şekil 3.1. YTA temelli siber savunma önerilerinin kurulum yerleri

Aşağıdaki alt bölümlerde sırasıyla ağ tarama, kimlik sahtekârlığı, ağ düzeyinde hizmet engelleme, dinleme ve son olarak zararlı yazılım, sosyal mühendislik ve web uygulama saldırılarına karşı YTA ortamında hayata geçirilebilecek savunma çözümleri sunulmaktadır. Her alt bölüm sonunda da önemli hususlar çizelge ile özetlenmektedir.

3.2.1. Ağ tarama saldırılarına karşı savunma

Siber saldırılar potansiyel hedefler hakkında bilgi toplamak ve güvenlik açıklarını belirlemek için keşifle başlar. Ağ taraması, daha kapsamlı siber saldırıların keşif aşamasındaki en kritik adımdır. Paketlerdeki IP adreslerini veya bağlantı noktası bilgilerini (ADRESİ-DEĞİŞTİR) değiştirmek, taşıdığı verilerdeki bilgileri (YÜKÜ-DEĞİŞTİR) güncellemek ve her bir bilgisayar için benzersiz bir ağ görünümü (SANAL-GÖRÜNÜM) sağlamak, ağ tarama saldırılarıyla başa çıkmak için kullanılan en yaygın savunma yaklaşımlarıdır. Bunlar arasında SANAL-GÖRÜNÜM, tarama ile mücadelede en kapsamlı

yaklaşımdır. Achleitner ve diğerleri [75, 76] keşif saldırılarına karşı savunma için Keşif Aldatma Sistemi (Reconnaissance Deception System, RDS) adlı bir ağ aldatma sistemi önermiştir. RDS, saldırganların tarama eylemlerini geciktirmekte ve toplanan bilgileri geçersiz kılarak ağ keşiflerini engellemektedir. RDS, ağdaki her bir bilgisayar için farklı sanal ağ topolojisi oluşturur ve DNS, ARP, dinamik istemci yapılandırma protokolü (Dynamic Host Configuration Protocol, DHCP) ve benzer veri iletim paketlerini yönetir. RDS'nin ağ keşif saldırılarına karşı koruduğu ve APT'leri tespit etmeye / yanıtlamaya yardımcı olabileceği savunulmuştur.

Chiang ve diğerleri [77], Uyarlanabilir Siber Aldatma Sistemi (Adaptive Cyber Deception System, ACyDS) adı verilen bir başka uyarlanabilir siber aldatma sistemi sunmuştur. ACyDS, DNS ve DHCP sunucularıyla koordineli olarak ağdaki her bir bilgisayara dinamik olarak farklı sanal ağ topoloji sağlar. IP başlıklarındaki yaşam süresi (time to live, TTL) alanı sanal görünümleri yönetmek için kullanılır ve bal küpleri her sanal görünüme rastgele dağıtılır. Çalışmada önerilen yaklaşımın keşif saldırılarını engellediği, tıkanıkları önlediği ve saldırganları tespit etme olasılığını ve güvenini arttırdığı savunulmuştur. Robertson ve diğerleri [78] her bilgisayar başına farklı topoloji oluşturarak saldırıları önleyen Aldatma ve Saldırı Etkisi Azaltma için Özelleştirilmiş Bilgi Ağları (Customized Information Networks for Deception and Attack Mitigation, CINDAM) adlı benzer bir aldatma sistemi önermiştir. Önerilen sistemin, ağ işlemlerini etkilemeden, istemci veya sunucu yazılımını değiştirmeden bir ağa saldırmayı zorlaştırabildiği belirtilmiştir. Belirtilen iki çalışmada [77, 78] ağ keşfi hakkında ayrıntılı bilgi vermemekte veya farklı saldırı stratejilerini değerlendirmemektedir.

ADRESİ-DEĞİŞTİR yaklaşımı, ağ tarama saldırılarına karşı yaygın bir savunma yaklaşımıdır. Jafarian ve diğerleri [79], IP adreslerini öngörülemeyen ve yüksek oranda şeffaf olarak mutasyona uğratmak için hareketli hedef savunmasına (HHS / moving target defense, MTD) dayalı OpenFlow Rastgele İstemci Mutasyonu (OpenFlow Random Host Mutation, OF-RHM) sunmuştur. YTA kontrolcüsü her istemciye rastgele bir sanal IP tanımlar. Gerçek IP adresleri değişmez ve mutasyon uç bilgisayarlar tarafından fark edilmez. Bilgisayarların isimlerine DNS sunucusundan alınan sanal IP adresleri aracılığıyla erişilebilir. Yalnızca yetkili birimler bilgisayarların gerçek IP adreslerini kullanabilir. OF-RHM'nin tarama ile bilgi toplamasını %99'a kadar geçersiz kılabileceği, sıfırıncı gün solucanlarından ve diğer tarama tabanlı saldırılardan %90 oranında koruduğu ifade edilmiştir. Ayrıca, önerilen mimarinin gizli tarama ve solucanlara karşı savunma sağladığı

belirtilmiştir. Jafarian ve diğerleri [80] çalışmalarında, kaynak IP adresi ve zamana bağlı olarak ağ üzerinde rastgele değişmesini sağlayan bir teknik önermiştir. Kaynak adresi ve zaman boyutları ile rastgele adresler kullanılması, zaman içinde toplanan keşif bilgileri ile elde edilen ağ topolojisini geçersiz kılmaktadır. Jafarian ve diğerleri diğer çalışmalarında [81, 82], tarama saldırılarını istemcilerden şeffaf bir şekilde yenmek için bilgisayarlarının adreslerini uyarlamalı olarak yeniden yapılandıran IP adresini rastgele değiştirme tekniği sunmuştur. Önerilen teknikle, zararlı davranışı karakterize etmek için hızlı ve doğru hipotez testi gerçekleştirilmiştir. Önerilen teknik, hem klasik ağlarda hem de yazılım tanımlı ağlarda test edilmiştir.

Ma ve diğerleri [83] HHS kullanarak kendinden uyarlamalı bir uç nokta atlamalı teknik (self-adaptive endpoint hopping technique, SEHT) önermiştir. Önerilen teknik; rastgele değişim denetleyicisinden, atlamalı anahtarlardan ve Linux terminal erişim noktası (TAP) Ethernet çerçevelerini yöneten atlamalı uç nokta bileşenlerinden oluşmaktadır. Rastgele değişim denetleyicisi, başarısız ağ isteklerini izler, ağ tarama stratejisinin türünü algılar, atlama stratejisini seçer ve önceden belirlenmiş bir süre için seçilen atlama stratejisini uygulamak için atlama anahtarı ve atlama uç noktası bileşenleri ile birlikte çalışır. Çalışmada önerilen teknik kullanılarak farklı türden tarama saldırılarının da engellenebileceği ifade edilmiştir. MacFarland ve Shue [84] farklı bir bakış açısıyla, ölçeklenebilirlik sınırlaması olmayan ve istemcileri değiştirmeyen bir istemci tabanlı hareketli hedef savunma tekniği sunmuştur. Sunulan teknik; öngörülemezlik, genişlik, periyodiklik, teklik, geri alınabilirlik, kullanılabilirlik ve ayırt edilebilirlik parametreleri kullanarak değerlendirilmiştir. Tekniğin temel güvenlik özelliklerini sağladığı ve savunma yaparken güvenilir ve güvenilir olmayan istemcileri ayırt edilmesine olanak sağladığı belirtilmiştir.

Wang ve diğerleri [85], rastgele etki alanı adı ve adres mutasyonuna (Random Domain Name and Address Mutation, RDAM) dayalı bir ağ savunma yöntemi önermiştir. Yöntemle, tüm istemcilerin etki alanı adları, dinamik şekilde oluşturularak periyodik olarak değiştirilmektedir. Bu şekilde saldırganın tarama alanı artırılmakta ve DNS sorgu listesi ile zaman penceresi yöntemleri kullanılarak hedeflere ulaşma olasılığı azaltılmaktadır. Araştırmacılar tarafından, önerilen yöntemin tarama saldırılarını ve solucan yayılımını engelleyebileceği belirtilmiştir. Hareketli hedef savunmaya dayalı bir başka çalışmada, Wang ve Wu [86], ağ keşiflerine karşı savunmaları için Sniffer Reflector adında yöntem sunmuştur. Yöntem, tarama yanıtlarının sanallaştırma kullanılarak oluşturulduğu ve

gizlendiği bir gölge ağına tarama trafiğinin yansıtılmasına dayanır. Sniffer Reflector'ün çeşitli ağ keşif saldırılarını engellemede etkili ve verimli olduğu ifade edilmiştir.

Zhao ve diğerleri [87] tarafından parmak izi saldırılarının en ciddi ağ tehditlerinden biri olduğu ve bu saldırıların gelecek saldırılar için hazırlık yapmak üzere hedef konakçılar hakkında bilgi edinmeyi amaçladığı belirtilmiştir. Yazarlar, parmak izi saldırılarına karşı savunma için YTA temelli parmak izi atlama yöntemi (Fingerprint Hopping Method, FPH) önermiştir. FPH, saldırganlara karşı atlamalı bir parmak izi göstermek için hareketli hedef savunma stratejisi kullanır. Çalışmada bir oyun modeli oluşturulmuş ve parmak izi saldırılarını modellemek için kullanılmıştır. Oyunun dengesi ile en uygun savunma yöntemi oluşturulmaya çalışılmıştır. Kampanakis ve diğerleri [88], istemci / bağlantı noktası tarama saldırıları ile işletim sistemi veya servis parmak izi saldırıları için yöntemler önermiştir. Önerilen yöntemlerde, her paket tutarlı davranış sağlamak için ara belleğe alınır, isteklere rastgele yanıtlar dönülür ve vekil TCP el sıkışmalarına rastgele gecikmeler eklenir. Yazarlar, önerilen yöntemlerin servis bulmaya çalışan saldırganlar için büyük bir gecikme yükü oluşturduğunu, bağlantı noktası tarama ve işletim sistemi parmak izi saldırıları için %100 - %200 oranında gecikmeye yol açtığı belirtilmiştir.

Shin ve diğerleri [69], YTA kullanarak güvenlik uygulamaları geliştirmek ve saldırı, zararlı yazılım, botnet algılama ve engelleme için güvenlik uygulamaları sunmak için FRESCO adlı bir çerçeve önermiştir. Uygulamalar FRESCO kodlama dilini kullanarak, akışlar için; düşürme, iletme, yönlendirme, yansıtma, ağ ayırma vb. eylemler oluşturabilmektedir. Yazarlar, FRESCO çerçevesinin çok az ek yük getirdiğini ve daha az satırla popüler güvenlik işlevlerinin hızlı bir şekilde geliştirilmesini sağladığını bildirmiştir.

Tarama genellikle diğer saldırı türlerinin habercisidir. Cabaj ve diğerleri [89], 5G mimari gereksinimlerine göre tasarlanmış Radyo Işığı İnterneti sistemindeki TCP SYN tabanlı tarama saldırılarını algılamak ve saldırıların etkisini azaltmak için YTA tabanlı bir güvenlik çerçevesi sunmuştur.

Önemli Noktalar: Ağ tarama saldırılarına karşı savunma yöntemleri ile ilgili özet bilgi Çizelge 3.3'te yer almaktadır. Tarama, karmaşık ve geniş çaplı saldırılarda çok önemli bir adımdır. Çünkü saldırgana hedef ağ hakkında değerli bilgiler sağlar. Taramaya karşı savunmanın ortak amacı, saldırganın ağ görüşünü gizlemek veya onları yanlış hedefe saldıracak şekilde kandırmaktır. Bu amaçla, hedef savunma ve bal küpleri gibi aldatmaya

yönelik teknikler, incelenen çalışmalarda önleyici tedbirler olarak popülerdir. Savunma stratejisi açısından, bu alt bölümde ele alınan 16 çalışmanın 11'i, hareketli hedef savunma stratejisini kullanmaktadır.

Çizelge 3.3. Ağ tarama saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
RDS [75, 76]	Önleme, Tespit	Her istemci için ağ topolojisi değiştirilir. Bal küplerine veya zafiyeti olan istemcilere bağlanmaya çalışan zararlı trafik tarama saldırısı olarak algılanır.	SANAL-GÖRÜNÜM, RASTGELE-ALDATICI	KNTL, UYG
ACyDS [77]	Önleme, Tespit	IP başlığındaki TTL alanı sanal topolojileri yönetmek için kullanılır. Her sanal ağa rastgele bal küpleri yerleştirilir. Bal küplerine erişmeye çalışan trafik tarama saldırısı olarak tespit edilir.	SANAL-GÖRÜNÜM, RASTGELE-ALDATICI	KNTL, UYG
CINDAM [78]	Önleme, Tespit, Azaltma	Her istemci için ağ topolojisi değiştirilir. Bal küplerine erişmeye çalışan istemciler saldırgan olarak tespit edilir.	SANAL-GÖRÜNÜM, RASTGELE-ALDATICI	KNTL
OF-RHM [79]	Önleme	Her bir istemciye belirli sürelerde rastgele sanal IP adresi atanır. Sanal IP adreslerinden gerçek IP adreslerine dönüşüm için DNS sunucusu kullanılır.	ADRESİ-DEĞİŞTİR	KNTL, UYG
[80]	Önleme	Ağ seviyesinde kaynağa ve zamana bağlı olarak istemci - IP eşleştirmesi rastgele şekilde uygulanır.	ADRESİ-DEĞİŞTİR	KNTL, UYG
[81]	Önleme	IP adresi değiştirilir ve yeni sanal IP adresi DNS sunucusundan alınır.	ADRESİ-DEĞİŞTİR	KNTL
RHM [82]	Önleme	Gerçek IP adresleri periyodik olarak YTA anahtarlarında sanal IP adreslerine dönüştürülür. Doğrudan istemcilere gerçek IP adresleri ile erişim, ağdaki erişim kontrol politikalarına göre yapılır. Ayrıca ters DNS adı değişikliği ve MAC adres değişikliği de yapılır.	ADRESİ-DEĞİŞTİR	KNTL

Çizelge 3.3. (devam) Ağ tarama saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
SEHT [83]	Önleme, Tespit, Azaltma	İstemcilerin IP adresi ve portları Sağlanabilirlik Modülo Teorilerine (Satisfiability Modulo Theories, SMT) göre rastgele olarak belirli periyotlarla değiştirilir. Tarama stratejilerinin tespiti için ağ trafiğinin entropisi hesaplanır.	ADRESİ-DEĞİŞTİR	KNTL, ANAH, İST, SUN
[84]	Önleme	İstemciler, sunucuların sentetik IP adreslerine ulaşmak için DNS sunucusuyla bağlantı kurarlar.	ADRESİ-DEĞİŞTİR	KNTL, ANAH, İST, SUN
RDAM [85]	Önleme	İstemciler, iş kanıtı şemalarını kullanarak kimlik doğrulama sunucusunda kimlik doğrulaması yapar. Kimlik doğrulama sunucusu, istemcilere göre farklı dinamik etki alanı adı listeleri yönetir.	AEK	KNTL, ANAH
Sniffer Reflector [86]	Tespit, Azaltma	Uyarlanmış Snort [90] STS uygulaması muhtemel tarama saldırılarını tespit eder. Tespit edilen trafik sanal gölge ağa yönlendirilir.	ALDATICIYA-GÖNDER	UYG, ANAH
FPH [87]	Tespit, Azaltma	STS ağdan parmak izi davranışlarını tespit eder. Eğer davranış şüpheli ise paketler işaretlenir. İşaretlenen paketler değiştirilerek cevaplar istemciye gönderilir.	YÜKÜ-DEĞİŞTİR	KNTL, UYG
[88]	Önleme	İşletim sistemi parmak izi ve port durumu tespiti için yapılan tarama saldırılarına sahte cevaplar hazırlar ve gönderilir.	YÜKÜ-DEĞİŞTİR	KNTL
FRESCO [69]	Tespit, Azaltma	Ağda durumsal filtreler uygulanabilmektedir. Paket Düşürme, iletme, yönlendirme, yansıtma, başlık değiştirme, ağ ayırma gibi programlanabilir işlemler kullanılarak savunma gerçekleştirilebilir.	YÖNLENDİR, DÜŞÜR, AĞ-AYIR, BAŞLIĞI-DEĞİŞTİR	KNTL, UYG
[89]	Tespit Önleme	Anahtarda TCP paketlerinin durumları takip edilir. Bekleme durumundaki TCP paketlerinin bağlantı sayıları yönetilir. Anomali tespit edildiğinde paketler düşürülür.	DÜŞÜR	KNTL

Taramaya karşı savunma mekanizmaları, yalnızca birkaç paket kullanarak bir yazılım tanımlı ağdaki akış kurallarının ayrıntılarını yüksek doğrulukla anlamlandıran kötücül ağ adli bilişim (adversarial network forensics) [91] adlı yeni bir olgu ile tehdit edilmektedir. Bu tür bilgilerle saldırganlar karşı önlemleri yenebilecek daha tehlikeli saldırılar yapabilmektedir. Bu olgu çok yeni bir gelişme olduğundan, literatürde onu etkisiz hale getirmek için etkili yöntemler bulunmamaktadır. Tarama saldırılarına karşı yeni savunma önerileri, YTA altyapısının kötücül ağ adli bilişime duyarlı olabileceğinin farkında olmalı ve ağları buna dirençli hale getirmek için yöntemler geliştirilmelidir.

3.2.2. Kimlik sahtekârlığı saldırılarına karşı savunma

Kimlik sahtekârlığında genellikle ağ protokollerinde kimlik doğrulama eksikliğinden yararlanılır. DDoS ve ortadaki adam gibi saldırılarda yaygın olarak kullanılır. Sahri ve Okamura [92–94], DNS sunucularını hedefleyen DDoS saldırılarına odaklanmıştır. Bu saldırılarda genellikle IP adresi kimlik sahtekârlığı gerçekleştirir. Gerçek bir sorguyu ve DNS sağlayıcısına hedef alan bir saldırganı ayırt etme zorluğu nedeniyle kimlik sahtekârlığının tespit edilmesi zordur. Yazarlar, gerçek DNS sorgularının kimliğini doğrularken kimlik sahtekârlığı sorgu paketini otomatik olarak engelleyen CAuth adlı YTA tabanlı bir mekanizma sunmuştur. Bir sunucuya sorgu gönderildiğinde, bir kimlik doğrulama paketi kontrolcü tarafından istemciye geri gönderilir ve kontrolcü tarafından bir kimlik doğrulama paketi ile yanıtlanır. Kontrolcü, sorguyu yalnızca istemciden kimlik doğrulama paketi alırsa DNS sunucusuna iletir. Bu mekanizma, mevcut DNS uygulamasında ve OpenFlow protokolünde değişiklik yapılmadan tasarlanmıştır. Yöntemin, böyle bir korumaya sahip olmayan duruma kıyasla ortalamadan sadece 1,2 kat fazla bant genişliği kullanarak DNS sunucusuna ulaşmadan gerçek ve saldırı paketleri arasında ayrım yapabildiği ifade edilmiştir.

Bazı çalışmalar ARP sahteciliğine odaklanmakta ve ARP sahteciliğine karşı savunma çözümleri önermektedir. Masoud ve diğerleri [95], ortadaki adam, DoS ve oturum çalma gibi birçok ağ saldırısının ARP sahtekârlığıyla başladığını belirtmişler ve YTA kullanarak bu sorunu çözmek için bir algoritma önermişlerdir. Algoritma, dinamik ve statik IP adresi atamalarını dikkate alır ve IP adreslerini doğrular. Cox ve diğerleri [96], OpenFlow destekli anahtarlarda MAC öğrenme protokollerini artıran ARP için Ağ Akış Koruması (Network Flow Guard for ARP, NFGA) adı verilen bir YTA güvenlik bileşeni sunmuştur. Modül, ARP

kimlik sahtekârlığını gerçek zamanlı olarak engellemek için dinamik veya statik IP ve bağlantı noktası ilişkilerine sahip bir düğümün fiziksel adresinin özetini (hash) oluşturur. Önerilen yöntemde, ağ topolojisi veya protokollerinde herhangi bir değişikliğe gerek olmamakta ve istemcilere yazılım kurulumuna ihtiyaç duyulmamaktadır.

Nehra ve diğerleri [97], ARP ile İlgili Tehditlere Trafik Örüntüsüne Dayalı Çözüm (Traffic Pattern Based Solution to ARP Related Threats, FICUR) adlı ARP tabanlı saldırıların doğrulanması ve tespiti için bir yöntem sunmuştur. Önerilen yöntemde, özelleştirilmiş bir YTA kontrolcüsü gerekli ağ verilerini toplar ve saldırıları doğrulamak ve tespit etmek için bu verileri analiz eder. Ayrıca, YTA yetenekleri kullanılarak anında etki azaltma gerçekleştirilir. Yazarlar, önerilen çözümün ağa sınırlı bir ek yük getirdiğini değerlendirmek için hem benzetim ortamında hem de gerçek ortamda test etmiştir. Ubaid ve diğerleri [98], bütçe kısıtlamaları ve sınırlı yetkinlikler nedeniyle organizasyon ağlarındaki eski cihazlar arasında sınırlı sayıda YTA cihazının konuşlandırılabileceğine dikkat çekmiştir. Yazarlar, saldırı durumunu otomatik olarak saptamak ve karma yazılım tanımlı ağlarda ARP kimlik sahtekârlığı saldırılarının etkisini azaltmak için başka bir teknik önermişlerdir. Saldırganın yerini göstermek için çizge tabanlı geçiş mekanizmaları kullanılmıştır. Benzer şekilde, Alharbi ve diğerleri [99], yazılım tanımlı ağların merkezi ağ kontrolünü kullanan ARP kimlik sahtekârlığı saldırılarına karşı Güvenli Adres Çözümleme Protokolü – Ağ Adresi Dönüştürme (Secure Address Resolution Protocol - Network Address Translation, SARP-NAT) adlı farklı bir etki azaltma yöntemi sunmuştur. Bu yöntemde tüm ARP istek ve yanıt paketleri doğrulanmak için işlenmektedir.

ARP gibi, IP v6 protokolünde kullanılan Komşu Keşif Protokolü de (Neighbor Discovery Protocol, NDP) kimlik doğrulamasından yoksundur ve kimlik sahtekârlığı saldırılarına karşı savunmasızdır. Lu ve diğerleri [100], NDP'nin koruma olmadan kolayca aldatılabildiğini belirtmiş ve yerel ağda iletilen NDP paketlerinin kimliğini doğrulamak için YTA tabanlı bir kimlik doğrulama mekanizması önermiştir. Bu mekanizma, YTA'nın merkezi kontrol ve programlanabilirlik özelliklerini kullanmakta, istemcilerde ve yönlendiricilerde ek bir protokol kullanımı veya yapılandırma gerektirmemektedir. Yazarlar, önerilen mekanizmanın kimlik sahtekârlığı saldırılarını ve kimlik sahtekârlığına dayalı diğer türetilmiş saldırıları etkili bir şekilde önleyebileceğini savunmuştur.

Mattos ve Duarte [101], istemci kimlik bilgilerine dayanarak AuthFlow adlı bir kimlik doğrulama ve erişim kontrol mekanizması önermiştir. AuthFlow, yazılım tanımlı ağlarda düşük ek yük getirerek istemci yetkilerine dayalı hassas erişim kontrolü sağlayacak şekilde IEEE 802.1x işlemlerini yönetmektedir. AuthFlow, YTA kontrolcülerinin, akış tablosu kurallarını oluşturmak için istemci kimliğini yeni bir akış alanı olarak kullanmasını sağlayan bir çerçeveye sahiptir. Yazarlar bir YTA kontrolcüsü kullanarak önerilen mekanizma için bir prototip geliştirmiştir. AuthFlow, geçerli kimlik bilgileri olmadan veya iptal yetkisi olmayan istemcilerin erişimini reddetmekte ve yetkisiz istemcilerin ağ kaynaklarına erişmesini önlemektedir. Ayrıca, çalışmada önerilen mekanizmanın, tarama ve dinleme saldırılarını önlemek için de kullanılabileceği ifade edilmiştir.

Kuliesius ve Dangovas [102], kimlik doğrulama, yetkilendirme ve denetleme gibi ağ güvenliği eylemleri uygulansa da, kullanıcı kutusu kimlik sahtekârlığı ve güvenlik cihazı atlama sorunlarının ağlarda görülebildiğini belirtmiştir. Yazarlar, ağı YTA tabanlı elemanlarla tamamlayarak erişim kontrol sistemini güvence altına almak için bir model önermiştir. Önerilen model, ağ cihazlarının, istemcilerin ve kullanıcıların kimlik doğrulaması, kaydı ve izlenmesi, veri akışlarının yönetimi ve kullanıcıların / istemcilerin hareketliliğini yöneten kontrol bileşenlerinden oluşmaktadır. Bileşenlerle erişim katmanı anahtarlarının kimliği doğrulanır, ağ topolojisiyle eşlenir ve izlenir. Önerilen model ile trafik politikasını net ve etkin bir şekilde kontrol etmek için kullanıcıları uygun anahtara / porta bağlamak mümkündür. Önerilen modelin tarama saldırılarını da engelleyebildiği belirtilmiştir.

Kwon ve diğerleri [103], Sınır Geçit Protokolü (Border Gateway Protocol, BGP) tabanlı Sahtecilikle Mücadele Eklentisi (BGP-based Anti-Spoofing Extension, BASE) adı verilen bir sahteciliğe karşı koruma mekanizması önermiştir. BASE, artırımlı kurulum özelliklerini yerine getirmek için tasarlanmış YTA tabanlı bir kimlik sahtekârlığı önleme protokolüdür. BASE üç tekniğe dayanır: mesaj kimlik doğrulama kodu, tek yönlü özet zincirleri ve paket işaretleme. Bir filtre düğümü için benzersiz değerler oluşturmak üzere mesaj kimlik doğrulama kodu ve tek yönlü özet zincirler kullanılır. Elde edilen değerleri hedeflere yüklemek ve göndermek için paket işaretleme kullanılır. Yazarlar, BASE'in kısmi kurulumunda istenen IP kimlik sahtekârlığı önleme yeteneklerini göstermiş ve yalnızca %30 BASE kurulumu ile sahte paketlerin yaklaşık %97'sinin tespit edebileceği ifade edilmiştir.

Bazı araştırmalar IP kimlik sahtekârlığıyla mücadele için Kaynak Adres Doğrulama İyileştirmesine (Source Address Validation Improvement, SAVI) dayanmaktadır. Liu ve diğerleri [104] YTA ağlarında SAVI işlevlerini etkinleştirmek için SDN-SAVI adlı bir YTA kontrolcü bileşeni sunmuştur. SDN-SAVI tasarımında üç zorluk ve çözüm sunmuştur. İlk olarak, anahtar bağlantı noktaları üç kategoride sınıflandırılır ve yedek anahtarlar atama mekanizmasının gözetlenmesini ve tüm anahtarlarda adres bağlanmasını önlemek için her bağlantı noktası bu kategorilere atanır. İkincisi, adres atama mekanizması paketleri sınırlıdır ve kontrolcünün doğrulama işi kaynak tükenme saldırılarını önlemek için anahtarlara devredilir. Üçüncü olarak, akış kural tablosu taşması sorununu çözmek için OpenFlow protokolü ile desteklenen birden çok akış kural tablosu kullanılır. Yao ve diğerleri [105], SAVI çözümlerini iyileştirmek için Sanal Kaynak Adres Doğrulama Kenarı (Virtual Source Address Validation Edge, VAVE) adlı koruyucu bir çevre yaklaşımı önermiştir. Belirlenen çevrenin dışından gelen bir paket, çevreye ulaştığında YTA kontrolcüsüne yönlendirilir. Kontrolcü, oluşturulan kurallara göre kaynak IP adreslerinin geçerliliğini kontrol eder. Sahte kaynak IP adresleri olan paketler bir süre engellenir. Yao ve arkadaşları [106] rota tabanlı IP kimlik sahtekârlığı filtrelemesi için Yazılım Tanımlı Filtreleme Mimarisi (Software Defined Filtering Architecture, SEFA) adlı bir çerçeve önermiştir. SEFA, OpenFlow protokolünün özelleştirilmiş bir sürümü olan OpenFlow+ sürümünü kullanmak için YTA anahtarının ve kontrolcünün değiştirilmesine dayanır. Yazarlar SEFA'nın kimlik sahtekârlığı için karmaşıklığı, ek yükü, kural üretimini ve yüklemelerdeki gecikmeleri azaltabileceğini ifade etmiştir.

Önemli Noktalar: Kimlik sahtekârlığı saldırılarına karşı savunma yaklaşımlarının özeti Çizelge 3.4'te verilmiştir. Kimlik sahtekârlığı, farklı protokollerde (ARP, IP, DNS, NDP vb.) birden çok ağ katmanında gerçekleştirilebilen bir saldırı türüdür. Kimlik sahteciliğine karşı savunma mekanizmalarının neredeyse tümü bir çeşit kimlik doğrulama ve kaynak doğrulama adımlarını içermektedir. Anahtarlar ve kontrolcü bu çözümlerin çoğunda önemli bir rol oynamaktadır. Bununla birlikte, derin paket incelemesi gibi ağ fonksiyonları, kimlik sahtekârlığına karşı yardım sağlayabilir. Bu nedenle, kimlik sahtekârlığını tespit etmek ve şüpheli trafik için bir SFZ parçası olarak derin paket incelemesi kullanımı konusunda yeni çalışmalar gerçekleştirilebilir.

3.2.3. Ağ seviyesinde hizmet engelleme saldırılarına karşı savunma

YTA, DoS saldırılarının potansiyel etkisinin ve savunma yaklaşımlarının etkinliğinin arttığı bir alan sağlar. YTA'da DoS ile mücadelede anomali tespiti için proaktif kaynak yönetimi ve saldırıları kandırma gibi farklı yaklaşımlar kullanılabilir.

Çizelge 3.4. Kimlik sahtekârlığı saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
CAuth [92–94]	Önleme	DNS istekleri birkaç saniye ara belleğe alınır ve her DNS isteği için kimlik doğrulama paketi gönderilir. İstemci kimlik doğrulama paketine yanıt verirse, DNS isteğine izin verilir.	AEK, DÜŞÜR	ANAH
[95]	Tespit, Azaltma	Dinamik veya statik IP - MAC adresi çiftleri doğrulanır. Doğrulanmamış ARP paketleri düşürülür.	AEK, DÜŞÜR	UYG
NFGA [96]	Tespit, Azaltma	IP-port çiftli MAC adresi, Pyretic [70] kullanılarak gerçek zamanlı olarak yönetilir. Geçersiz ARP paketleri düşürülür.	DÜŞÜR	UYG
FICUR [97]	Önleme, Tespit, Azaltma	Tüm ARP trafiği kontrolcüye yönlendirilir ve günlük dosyası kullanılarak doğrulanır. IP paketleri, ARP isteği ve cevap paketleri doğrulamak için işlenir. Gereksiz ARP paketleri engellenir.	AEK, DÜŞÜR	ANAH
[98]	Önleme, Tespit, Azaltma	Tüm ARP trafiği kontrolcüye yönlendirilir ve doğrulanır. ARP paketlerinin IP ve MAC adresleri topoloji bilgileri kullanılarak doğrulanır. Doğrulanmamış ARP paketleri düşürülür.	AEK, DÜŞÜR	ANAH
SARP-NAT [99]	Tespit, Azaltma	ARP istek ve yanıt paketleri doğrulamak için işlenir. ARP istek paketleri bekleme kuyruğa alınır ve yalnızca bu kuyruktaki ARP yanıt paketlerine izin verilir.	AEK, DÜŞÜR	ANAH

Çizelge 3.4. (devam) Kimlik sahtekârlığı saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
[100]	Önleme, Tespit, Azaltma	Tüm NDP trafiği kontrolcüye yönlendirilir ve ağ topolojisi bilgileri kullanılarak doğrulanır. Doğrulanmamış NDP paketleri düşürülür.	AEK, DÜŞÜR	ANAH
[101]	Önleme	IEEE 802.1X'e göre Genişletilebilir Kimlik Doğrulama Protokolü (Extensible Authentication Protocol, EAP) ile kimlik doğrulaması ağda gerçekleştirilir.	AEK	ANAH, UYG
[102]	Önleme	Akışların kullanıcı ve erişim hakları denetlenir ve akışlar yalnızca yetkilendirme veri tabanları (cihazlar ve kullanıcılar, grup hakları ve akış sayaçları) kullanılarak geçerli trafik için oluşturulur.	AEK, DÜŞÜR	ANAH
BASE [103]	Önleme	IP başlığındaki Hizmet Türü (ToS) alanı, akış işaretleme alanı olarak kullanılır ve otonom sistemlerde akışları yönlendirmek için kullanılır.	YÖNLENDİR, BAŞLIĞI-DEĞİŞTİR, DÜŞÜR	ANAH
SDN-SAVI [104]	Önleme	Her IP adresinin durumları (BAĞLI, BAĞLI-DEĞİL) tabloda saklanır. IP adresinin durumu BAĞLI olarak değişirse, IP adresi, anahtar ve anahtar bağlantı noktası kullanılarak bir bağlayıcı verisi oluşturulur. Durum BAĞLI-DEĞİL olarak değişirse, IP adresi tablodan kaldırılır.	AEK, DÜŞÜR	ANAH
VAVE [105]	Tespit, Azaltma	Kontrolcü, oluşturulan kurallara göre paket kaynağının geçerli olup olmadığını kontrol eder. Kaynak adresi, aldatma IP adresi olarak algılanırsa paket düşürülür.	AEK, DÜŞÜR	ANAH
SEFA [106]	Önleme	Her kaynak IP adresi denetlenir. Rotaya dayalı yönlendirme, kontrolcü ve anahtarların koordinasyonu ile gerçekleştirilir.	AEK, DÜŞÜR	ANAH, UYG-ANAH

Bazı yöntemler, saldırıları önlemek veya etkilerini azaltmak için ağ çevikliğine dayanır. Jafarian ve diğerleri [107] ağın güvenlik, performans ve hizmet kalitesi ile ilgili gereksinimlerini karşılarken en uygun stratejiyi belirlemek için oyun teorisini ve kısıtlama memnuniyeti optimizasyonunu birleştiren rastgele rota mutasyonu (Random Route Mutation, RRM) adı verilen çevik çok rotalı yönlendirme yaklaşımı önermiştir. RRM, seçimi kısıtlama memnuniyeti optimizasyonu olarak yönlendirir ve verimli pratik rotaları belirlemek için SMT kullanarak gerçekleştirir. Yazarlar, RRM'nin geleneksel ve yazılım tanımlı ağlara sağlam ve düzgün bir şekilde kurulabilmesi için algoritmalar sunmuştur. RRM'nin, tek rota yönlendirme şemalarıyla karşılaştırıldığında, akış paketlerinin %90'ına kadar kalıcı saldırganlara karşı koruyabileceğini belirtmiştir. Gillani ve diğerleri [108], kritik kaynakların altyapısını proaktif olarak değiştirerek DDoS saldırılarına karşı savunmak için Çevik Sanal Ağ Çerçeve Modeli adlı bir model önermiştir. Çevik Sanal Ağ Çerçeve Modeli, ağ kaynaklarını dinamik olarak yeniden tahsis etmek için sanal ağları kullanır ve ağ bütünlüğünü korurken altyapının yeni kaynaklara taşınmasını sağlar.

Fichera ve diğerleri [109] TCP SYN sel saldırılarına karşı OpenFlow temelli çözüm (OpenFlow Based Remedy to TCP SYN Flood Attacks, OPERETTA) adlı bir protokol ve uygulama önermiştir. OPERETTA YTA kontrolcüsüne uygulanır, gelen TCP SYN paketlerini yönetir ve sahte bağlantı isteklerini reddeder. Önerilen protokolün değerlendirmesinde OPERETTA'nın işlemci ve bellek tüketiminin düşük olduğu ve OPERETTA'nın TCP SYN sel saldırılarına karşı dayanıklı olduğu belirtilmiştir. SYN sel saldırılarını önlemeye yönelik başka bir öneri de Shin ve diğerlerinin [110] sunduğu, bağlantı azaltma ve tetikleyici bileşenlerinden oluşan AVANT-GUARD adlı çalışmadır. Tetikleyiciler, değişen trafik oranının algılanmasını ve buna tepki verilmesini artırmak için YTA kontrolcüsü ve YTA anahtarları arasında bulunur. Bu tetikleyiciler, üst katman uygulamalarına eş zamansız olarak erişmek ve koşullu akış kuralları eklemek için kullanılır.

Giotis ve diğerleri [111], OpenFlow ve sFlow özelliklerini kullanan bir algılama ve etki azaltma mimarisi önermiştir. Anormallikler, sFlow kullanılarak 30 saniyelik zaman aralıklarında toplanan ağ trafiğinden hesaplanan entropi değerleri kullanılarak tespit edilir. Etki azaltma yaklaşımları bir beyaz listeyi yönetmeyi ve diğer ağ trafiğini engellemeyi içerir. OpenFlow akış tablosu değişiklikleri, saldırıların etkisini azaltmak için kullanılır. Yazarlar çalışmalarında DDoS, solucan ve port tarama saldırıları için de değerlendirme yapmıştır. Hussein ve diğerleri [112], DDoS saldırılarını önlemek ve tespit edilen saldırının kaynağını

takip edebilmek için bir YTA güvenlik yaklaşımı önermiştir. Önerilen yaklaşım, veri katmanına ek olarak YTA kontrol katmanına paralel yeni bir YTA güvenlik katmanı sunmaktadır. Güvenlik katmanında, anahtar üzerinde bir üçüncü taraf ajanı ve kontrolcünün yanında bir üçüncü taraf yazılım bileşeni bulunur. Yazarlar, önerilen sistemin düşük ek yük ve minimum yapılandırma ile farklı düzeylerde gerçek zamanlı kullanıcı tanımlı güvenlik sağlayabileceğini belirtmiştir.

Joldzic ve diğerleri [113], alt düzey DoS saldırılarını tespit etmek ve önlemek için Şeffaf İzinsiz Giriş Tespit Sistemi (TIDS) adı verilen dağıtık, ölçeklenebilir bir çözüm sunmuştur. YTA özellikleri ile birden çok trafik işlemcisi arasında ölçeklenebilir etkin trafik dengelemesi, cihaz yoklamasına dayalı olarak uygulanmaktadır. TIDS, çok sayıda saldırıya uğramış istemciyi izler ve tüm etkin saldırıları aynı anda azaltır. Saldırının kaynağı ve hedefi yalıtıldıktan sonra işlemci sınırlı bir süre için engellenecek adreslerin listesini içeren bir akış değiştirme isteği mesajı gönderir. Yazarlar, önerilen sistemin istemcilerden bağımsız olduğunu ve geçersiz TCP sıra numaraları, SYN taşkınları ve diğer benzer saldırıların ek kaynak tüketimi olmadan kolayca tespit edilebileceğini ifade etmiştir. Li ve diğerleri [114], istemciler ve İSS arasında iletişimi sağlamak için DrawBridge adlı bir trafik mühendisliği sistemi önermiştir. DrawBridge, akış kurallarını bir İSS'deki YTA anahtarlarına aktarabilen veya İSS akış yönünde başka bir DrawBridge denetleyicisiyle iletişim kurabilen bir YTA kontrolcüsüdür. Yazarlar, önerilen sistemin bir kontrolcünün akış kurallarını doğrulamasına ve işlemesine izin verdiğini ve bunları DDoS trafiğini filtrelemek için en iyi konuma sahip YTA anahtarlarına veya İSS'lerine dağıtmasına izin verdiğini belirtmiştir.

Miao ve diğerleri [115] tarafından, saldırı tespiti için trafiği analiz eden bir dizi sanal makine örneği ve bir otomatik ölçek denetleyicisi içeren YTA tabanlı NIMBUS adında çerçeve önerilmiştir. Sunulan çerçeve, aşırı yüklenmeyi önlemek için sanal makine örneklerini ölçeklendirir, anomali detektörü ve etki azaltma modüllerini dinamik olarak başlatır. Yazarlar, kara listenin veya hız sınırlama stratejilerinin yönlendiricilerde tespit edilen saldırı trafiğine uygulandığını vurgulamaktadır. Oktian ve diğerleri [116], OpenFlow yerel ağlarında IP / MAC kimlik sahtekârlığı ve hantal / çöp mesajı saldırıları ile birlikte DoS saldırılarının etkisini azaltmak için YTA uygulama katmanında Dossy adlı bir uygulama tanıtmıştır. Uygulama ağdaki düğümleri izler. Başlangıçta, YTA kontrolcüsü tüm istemcilerin MAC ve IP bilgilerini toplar ve bunları bir özet tablosunda birleştirir. Her paket giriş mesajı işlenir ve IP / MAC adresleri yönetilen özet tablosu dikkate alınarak analiz edilir.

Önerilen uygulamanın standart anahtarlara kıyasla kontrolcünün verimini %7,76 azalttığı ve gecikmeyi %7,37 arttırdığı belirtilmiştir.

Ölçeklenebilirlik, hızlı reaksiyon ve düşük ek yük, herhangi bir DDoS savunma mekanizması için gerekli olan özelliklerdendir. Piedrahita ve diğerleri [117], hafif ve hızlı bir DoS algılama ve etki azaltma sistemi olan FlowFence yöntemini önermiştir. YTA kontrolcüsü, bağlantıların bant genişliği atamasını koordine eder ve düğümlerin tıkanmasını önlemek için bir rota boyunca akış iletim hızını sınırlar. Atanan sınırları aşan akışlar cezalandırılır. FlowFence yönteminin, küçük bir ek yük eklerken kullanıcıların ağ kaynaklarının tıkanıklığını önlediği ifade edilmiştir. Hız sınırlandırmayı kullanan bir diğer öneri de akış bilgilerinin ağda sürekli olarak izlendiği ve bant genişliği taleplerinin gelecek için tahmin edildiği SDNManager [118] mimarisidir. Bu mimaride bant genişliği tahmini genel olarak ekonometride kullanılan otoregresif koşullu hetero-esneklik modeli uyarlanarak gerçekleştirilmektedir. Bu model kullanılarak zaman serilerindeki değişiklikler ölçülür ve ağın toplam bant genişliğinin dalgalanması tahmin edilir. Tahmin sonrasında, mevcut periyotta tahmini bant genişliğinin üzerinde olan anahtarlar tespit edilerek cezalandırılır.

Wang ve diğerleri [119], saldırı algılamayı etkinleştirmek, hızlı ve özel saldırı reaksiyonuna izin vermek için YTA kullanarak DDoS Saldırı Etkisi Azaltma Mimarisi (DDoS Attack Mitigation Architecture Using Software Defined Networking, DaMask) adlı bulut tabanlı bir mimari önermiştir. DaMask, ağ saldırı algılama (DaMask-D) ve saldırı etkisi azaltma (DaMask-M) bileşenlerinden oluşur. DaMask-M tarafından bir uyarı alınırsa bileşen, uyarı için bir karşı önlem arar. Varsayılan önlem, uyarı için önceden ayarlanmış eşleşme yoksa paketi düşürür. DaMask-M, bir dizi ortak uygulama programlama ara yüzüne sahiptir, böylece farklı DDoS saldırıları için farklı savunma önlemleri özelleştirilebilir. Mimaride yer alan üç temel karşı önlem şu şekildedir: ilet, düşür ve değiştir. Karşı önlem seçilirse, DaMask-M akış kurallarını YTA kontrolcüsü üzerinden anahtara gönderilir. FloodDefender [120], YTA kontrolcüsünü DDoS saldırılarına karşı korumak için bir çerçeve olarak önerilmiştir. FloodDefender, paket iletilerini (bir anahtarda akış tablosu yokken oluşturulan) kontrolcüyü bombalayan saldırıları tespit etmek, saldırı paketlerini filtrelemek ve akış kurallarını yönetmek için kontrolcü ve uygulamalar arasına kurulur. FloodDefender ayrıca, paket giriş mesajlarının bazılarını komşu anahtarlara göndererek anahtarın kullandığı bant genişliğinin tükenmesine karşı korur ve daha sonra bunları kontrolcüye iletir. Çalışmada,

önerilen yöntem ile saldırı sırasında hem kontrolcüdeki işlemci kullanımının hem de anahtardaki akış tablosu kullanımının %15'in altında kaldığı bildirilmiştir.

Paket ve bayt sayıları ile ilgili çeşitli istatistikler YTA anahtarları tarafından tutulur. Bu istatistikler kontrolcü tarafından sürekli olarak toplanabilir ve DDoS saldırılarını tespit etmek için işlenebilir. Örneğin, bazı çalışmalar, paket giriş sayılarının [121] veya paket giriş / aktarılan paket oranlarının [122] dağılımını analiz ederek DDoS saptaması gerçekleştirmiştir. Bu tür yöntemler, özellikle güney yönlü ara yüz üzerinden YTA kontrolcüsüne hedef alan DDoS saldırılarına karşı etkilidir. Diğer bazı sistemler, anahtarlardan istenen bilgileri almak için yazılım tanımlı ağlardaki istatistik toplama yeteneğini kullanır. Düşük oranlı DoS saldırıları için bir çözüm olan SoftGuard [123], toplam veri miktarını periyodik olarak ölçmek ve toplam bant genişliğini sürekli olarak izlemek için her bir anahtara izleme kuralları ekleyerek çalışır. Bant genişliğinde önemli bir azalma meydana geldiğinde, saldırganı tespit aşaması başlar. Tespit aşaması sırasında, önceden belirlenmiş ortalama bant genişliğinin üzerindeki anahtar ara yüzleri işaretlenir ve gerekli ise düşürülür.

Bazı çalışmalar, çapraz ateşleme saldırısı [124] gibi bağlantı taşkın saldırılarını (Link Flooding Attacks, LFA) tespit etmek ve bu saldırıların etkisini azaltmak için savunma sistemleri önermiştir. LFA, bir grup seçili bağlantıyı taşırmak için geçerli görünen düşük yoğunluklu akışlar uygular ve geleneksel şemalarla ayırt edilmesi zordur. Wang ve diğerleri [125], LFA saldırıların etkisini azaltmak için Woodpecker adlı bir şema önermiştir. Woodpecker, artırımlı YTA kurulumu için sezgisel bir algoritma, LFA tespiti için bir mekanizma ve merkezi bir trafik mühendisliği bileşeni içerir. Önerilen şema, bir ağın bağlantısını değerlendirmek için kaynak-hedef anahtar çiftleri arasındaki ortalama ve minimum rota sayısını bulur. Bu ek rotalar trafiği dengelemek için kullanılır. Önerilen şemanın, LFA tarafından saldırıya uğrayan bağlantıların bant genişliği kullanımını yaklaşık %50 oranında azalttığı, ortalama paket kaybı oranını ve sapmayı da etkili bir şekilde azalttığı ifade edilmiştir. LFADefender [126] adı verilen bir başka öneride maliyet verimliliği ve esneklik sağlamak için yazılım tanımlı ağların özelliklerinden yararlanılmıştır. LFADefender, yüksek akış yoğunluğuna sahip potansiyel hedef bağlantıları belirlemek için ağı inceler, bağlantı sıkışıklığını sürekli olarak izler, devam eden bir saldırının etkisini azaltmak için trafiği sıkışık bağlantılardan yeniden yönlendirir ve zararlı trafiği engeller. Bu

görevlerin her biri, Floodlight [127] kontrolcüsünün üstündeki ayrı bir bileşen tarafından gerçekleştirilir.

Zheng ve diğerleri [128], uyarlamalı korelasyon analizi kullanarak DDoS saldırılarını tespit etmek ve etkilerini azaltmak için gerçek zamanda DDoS karşı eylemlerin zorlanması (Reinforcing Anti-DDoS Actions in Real-time, RADAR) adlı bir mimari önermiştir. RADAR; bağlantı seli, SYN seli ve DNS güçlendirme saldırıları için algılama bileşenlerine sahiptir. RADAR çeşitli konumlarda yakalanan trafiği analiz eder, şüpheli olarak sınıflandırılmışsa trafiği daraltır ve port tabanlı maksimum-minimum eşitlik tekniği kullanarak paketleri düşürür. Yazarlar mimarilerini gerçek bir test yatağında değerlendirmiş ve RADAR'ın çeşitli saldırıları kısa bir süre içinde etkili ve verimli bir şekilde tespit edebileceğini ifade etmiştir.

YTA ortamlarında DDoS tespiti için derin öğrenme tabanlı yöntemler [129] önerilmiştir. Kontrolcü, ağın net bir görüntüsüne sahip olduğu ve daha önce bilinen bir akışa ait olmayan her paketi gördüğü için, paketleri işleyerek başlık alanlarından özellikler çıkartır ve derin bir öğrenme modeli kullanarak trafiği geçerli veya DDoS olarak sınıflandırır. Daha sonra kontrolcü, saldırı paketlerini anahtarlarda düşürmek için akış kurallarını yükleyerek hızlı bir şekilde tepki verebilir. Yakın zamanda yapılan bir başka çalışmada, akış tablosu eşleşmelerinin zamansal dağılımı analiz edilmiş ve saldırgan davranışı yapay sinir ağı [130] kullanılarak karakterize edilmiştir. Yaklaşımın uygulanabilir olduğu gösterilmiş, ancak gerçek hizmetlerin kullanılabilirliğinin olumsuz etkilenebileceği belirtilmiştir.

Son yıllarda, DDoS saldırılarına karşı savunmada blok zincir kullanımı önerilmiştir. Singh ve diğerleri [131] blok zincir tabanlı DDoS saldırı etkisini azaltma mimarilerini literatürde gözden geçirip değerlendirmiştir. YTA'nın tek bir etki alanındaki DDoS saldırılarına karşı savunmada kullanılabileceği, ancak birden fazla etki alanına yapılan DDoS saldırılarına karşı blok zincir ve YTA'nın birlikte kullanılmasının daha etkili olacağı belirtilmiştir. El Houda ve diğerleri [132], yerel ağda makine öğrenimi tabanlı DDoS algılama ve etki azaltma bileşenlerinden oluşan bir çoklu YTA etki alanı çözümü olan CoChain-Sc ve etki alanları arası DDoS hafifletme için kullanılan Ethereum blok zinciri [133] tabanlı bir modül önermiştir. Buna ek olarak, diğer iki çalışma [134, 135] da yasaklı IP adresleri gönderen ve farklı akıllı sözleşmeler kullanarak bir Ethereum blok zincirinden diğer yasaklı IP adreslerinin bir listesini alan DDoS saldırısı etkisi azaltma bileşenleri önermiştir.

Literatürde, kontrolcü tabanlı veya orta kutu tabanlı çözümlere kıyasla daha iyi ağ performansı elde etmek için veri katmanında bulunan anti-DDoS önerileri de bulunmaktadır. Örneğin FastFlex [136], programlanabilir anahtarlardaki saldırıların etkisini azaltmak için çok modlu veri katmanı kavramına dayanmaktadır Bu yaklaşım, bir saldırı sırasında kontrolcü iletişimiyle ilişkili bant genişliği yükünü ve gecikmeyi ortadan kaldırmaktadır. Benzer şekilde, Poseidon [137], dinamik yüksek hacimli DDoS saldırılarıyla başa çıkmak için bileşenlere bölünebilir savunma ilkellerini kullanmakta ve kontrolcüde oluşacak ek yükü azaltmaktadır. Poseidon dili ile çalışma zamanı ve düzenleme bileşeni birlikte kullanılarak çok çeşitli DDoS savunma politikaları oluşturulabilmektedir.

Önemli noktalar: Ağ seviyesinde DoS saldırılarına karşı savunma yöntemleri konusunda buraya kadar özetlenen çalışmalar ile ilgili özet bilgi Çizelge 3.5'te yer almaktadır. YTA ve DoS arasındaki ilişki önemli bir konudur çünkü YTA mimarisi, kontrolün merkezileştirilmesi ve programlanabilirlik nedeniyle yeni risklere neden olurken bir taraftan da daha iyi DoS savunması için fırsatlar sunar. Bu nedenle, DoS'a karşı YTA tabanlı tekniklerin ve sistemlerin geliştirilmesi son yıllarda popüler bir araştırma konusu olmuştur.

Bazı çalışmalar belirli bir DoS saldırı tipine odaklanırken, diğerleri daha genel yaklaşımlar sunmaktadır. Önerilen yöntemlerde bu çalışmada belirlenen tüm savunma yaklaşımları uygulanmış ve YTA'nın tüm katmanlarını içerecek şekilde çözümler sunulmuştur. SYN seli ve bağlantı seli gibi iyi bilinen DoS saldırı türlerine karşı özel olarak geliştirilen birden çok çözüm vardır. En yaygın savunma ilkesi, paketleri düşürme, ardından akış yönlendirme ve hız sınırlamadır. Birçok çalışma, anahtarlardan bilgi toplamak ve çoğunluğu eşik değer tabanlı olan çeşitli yöntemlerle DoS saldırılarını tespit etmek için kontrolcünün merkeziyetinden yararlanır. Önleme ve etki azaltma çözümleri, trafiği yeniden yönlendirme ve istemcileri bir kara listeye göre engelleme yaklaşımlarıyla birlikte tıkanıklığı azaltma gibi ek savunma eylemlerini de kullandıklarında daha etkili olmaktadır. Ayrıca, ağ topolojisinden hedef bağlantıları tahmin eden LFADefender'da görüldüğü gibi, saldırganın eylemlerini tahmin etmeye çalışmanın iyi bir yaklaşım olduğu değerlendirilmektedir.

Çizelge 3.5. Ağ seviyesi hizmet engelleme saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
RRM [107]	Tespit, Azaltma	Rastgele rota, saldırı için en uygun stratejiyi belirlemek için oyun teorisi ve kısıtlama memnuniyeti optimizasyonu kullanılarak belirlenir. Akışlar, seçilen rotalara göre yeniden yönlendirilir.	YÖNLENDİR	KNTL
[108]	Azaltma	Göç mekanizması, sanallaştırılmış bir altyapıdaki çerçeve tarafından oluşturulan sanal ağ geçiş stratejisini yürütür.	YÖNLENDİR	UYG
OPERETTA [109]	Önleme	TCP SYN paketleri kontrolcüye gönderilir. Paket hedef istemciye gönderilmeden karşı tarafa yanıt verilir.	AEK	KNTL
AVANT-GUARD [110]	Önleme, Tespit, Azaltma	Paket ve akış istatistikleri, yüklü koşulların oluşup oluşmadığını belirlemek için kullanılır. İlgili koşul olayı tetiklendiğinde koşullu akış kuralları tetiklenir.	BAŞLIĞI-DEĞİŞTİR, DÜŞÜR, KARA-LİSTE, YÖNLENDİR, AEK	ANAH
[111]	Tespit, Azaltma	SFlow'dan toplanan akış bilgileri, entropi tabanlı algoritma kullanarak anormalliği tespit etmek için kullanılır. Zararlı trafiği engellemek için daha yüksek önceliğe sahip akış kuralları eklenir.	DÜŞÜR	UYG-ANAH
[112]	Önleme, Tespit, Azaltma	Anahtarlar, paketleri Pyretic kullanarak algılama motoruna gönderir. Algılama motoru, saldırıyı engellemek için kontrolcünün anahtarlara eklemesi için uygun kuralları gönderir.	DÜŞÜR	ANAH, UYG-ANAH
TIDS [113]	Tespit, Azaltma	Cihaz sorgulama işlemcisi, anormallikleri tespit etmek için akışları analiz eder ve önceden belirlenen bir süre için yasaklanacak adreslerin listesini içeren akış güncelleme kuralları gönderir.	DÜŞÜR	UYG-ANAH
DrawBridge [114]	Azaltma	Akış kuralları, İSS ağındaki YTA anahtarlarına gönderilir veya kuralları uygulamak için İSS'deki YTA kontrolcüsüne iletilir.	YÖNLENDİR, DÜŞÜR	KNTL

Çizelge 3.5. (devam) Ağ seviyesi hizmet engelleme saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
NIMBUS [115]	Tespit, Azaltma	Bulut veri merkezindeki trafik izleme bileşeni, genel olarak çok düşük örnekleme hızıyla tek tip örnekleme oluşturulur. Otomatik ölçeklenebilir sanal makinelerde kara liste veya hız sınırlama uygulanır.	HIZI-SINIRLA, KARA-LİSTE	KNTL
Dossy [116]	Tespit, Azaltma	DoS saldırısını tespit etmek için hem akış tabanlı hem de paket giriş analizi kullanılır. Akış güncelleme mesajları, DoS saldırı adreslerini engellemek için anahtarlara gönderilir.	DÜŞÜR	KNTL
FlowFence [117]	Tespit, Azaltma	YTA anahtarları tarafından trafik tıkanıklıkları tespit edilir. YTA kontrolcüsü akışların hızlarını yönetir.	HIZI-SINIRLA	KNTL, ANAH
SDNManager [118]	Önleme	Trafik izlenir, bant genişliği talepleri tahmin edilir ve tahminleri aşan akışlar anahtarlarda sınırlandırılır.	HIZI-SINIRLA	KNTL
DaMask [119]	Tespit, Azaltma	Çizge kullanarak anormallik tabanlı saldırı tespiti yapılır. Karşı önlem olarak yönlendir, düşür ve değiştir eylemleri kullanılır.	BAŞLIĞI-DEĞİŞTİR, DÜŞÜR	UYG
FloodDefender [120]	Tespit, Azaltma	Gelen paket mesajları kullanılarak saldırı tespit edilir, saldırı paketleri filtrelenir ve anahtarlarda akış kuralları yönetilir.	BAŞLIĞI-DEĞİŞTİR, DÜŞÜR	UYG, KNTL
[121]	Tespit, Azaltma	Gelen paket eşik değerlerine bağlı olarak paketler düşürülür.	DÜŞÜR	KNTL
[122]	Tespit, Azaltma	Gelen paket / gönderilen paket dağılımının eşitliği izlenir ve eşitlik dengesi önemli ölçüde değiştiğinde saldırı olarak algılanır.	KARA-LİSTE	KNTL
SoftGuard [123]	Tespit, Azaltma	Düşük hızlı TCP saldırılarını tespit etmek için akış kuralları oluşturulur, anahtarların girişinde kurallar eklenerek saldırılar önlenmeye çalışılır.	HIZI-SINIRLA, DÜŞÜR	KNTL, ANAH

Çizelge 3.5. (devam) Ağ seviyesi hizmet engelleme saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
Woodpecker [125]	Tespit, Azaltma	Anahtarlar, tıkanıklıkları YTA kontrolcüsüne iletir. Kontrolcü trafiği aldatıcılara yönlendirir ve tıkanıklık olan rotaları değiştirmeye veya paketleri düşürmeye çalışır.	ALDATICIYA-GÖNDER, YÖNLENDİR, DÜŞÜR	KNTL, ANAH
LFADefender [126]	Tespit, Azaltma	Ağdaki hedef bağlantıları inceler, tıkanıklıkları izler, trafiği tekrar yönlendirir ve kötücül paketleri düşürür.	YÖNLENDİR, DÜŞÜR	KNTL
RADAR [128]	Tespit, Azaltma	Şüpheli trafik hızı sınırlandırılır ve saldırı trafiği port temelli minimum - maksimum denge tekniği kullanılarak düşürülür.	HIZI-SINIRLA, DÜŞÜR	KNTL
[129]	Tespit, Azaltma	DDoS saldırılarının tespiti için derin öğrenme kullanılarak trafik sınıflandırılır ve anahtarlara paketlerin düşürülmesi için kurallar eklenir	DÜŞÜR	KNTL
[130]	Tespit, Azaltma	Kural tablolarındaki eşleşmelerin zamansal dağılımı analiz edilir ve yapay sinir ağları kullanılarak saldırgan tespit edilir.	DÜŞÜR	KNTL
Cochain-SC [132]	Tespit, Azaltma	İç ağda sFlow kullanarak entropi temelli akış anomali tespiti ve azaltma yöntemi, dış ağda ise Ethereum blok zincirdeki akıllı sözleşmeler kullanılır.	HIZI-SINIRLA, KARA-LİSTE, DÜŞÜR	UYG-ANAH, UYG
[134]	Azaltma	IP kara listeleri Ethereum blok zincirdeki akıllı sözleşmeler kullanılarak iş birliği ile yönetilir.	KARA-LİSTE, DÜŞÜR	UYG
Co-IoT [135]	Azaltma	Zararlı IP adresleri, birden fazla etki alanı tarafından blok zincirdeki akıllı sözleşmeler kullanılarak yönetilir.	KARA-LİSTE, DÜŞÜR	UYG
FastFlex [136]	Tespit, Azaltma	Veri katmanında, tehdit tespit edildiğinde paketler gönderilerek ağda alarm oluşturulur. Anahtarlarda mevcut moda göre paket yönlendirme, düşürme ve IP karartma işlemleri uygulanır.	YÖNLENDİR, BAŞLIĞI-DEĞİŞTİR, DÜŞÜR	ANAH
Poseidon [137]	Tespit, Azaltma	Kural yükleme motoru, programlanabilir anahtarlara üst seviye savunma kuralları yükler. Savunma kuralları veri katmanında uygulanır.	DÜŞÜR, KARA-LİSTE, HIZI-SINIRLA	ANAH

Bu alandaki çalışmaların çokluğu ve çeşitliliğine rağmen, çalışmaların çoğunda hala bazı ortak sorunlar bulunmaktadır. İlk olarak; çözümlerin çoğu, yazılıma daha fazla bileşen ve ağ kontrol iş akışına daha fazla adım eklenerek kontrolcüde gerçekleştirmektedir. Bu tür öneriler YTA mimarisinin DoS saldırılarına karşı temel zayıflığını gidememektedir hatta kontrolcüye daha fazla yük getirerek sorunu daha da kötüleştirebilmektedir.

Literatürdeki bazı yeni çalışmalar [136, 137], ağ savunmasını veri katmanında yaparak kontrolcüye olan bağımlılığın ortadan kaldırılması yönünde adımlar atmış ve ağ performansının iyileştirilmesini sağlamıştır. Bu tür çalışmaların artmasının fayda sağlayacağı değerlendirilmektedir. İkincisi; hem kapsamlı hem de gerçekçi bir değerlendirme içeren çok fazla çalışma bulmak zordur. Bazı çalışmalar, gerçeğe uzak çok küçük bir test ağı kullanarak çok çeşitli performans metriklerinde çeşitli sonuçlar sunmaktadır. Bazıları ise simülasyon araçları üzerinde daha büyük bir topoloji oluşturmaya çalışmış ancak ortam sınırlamaları nedeniyle bazı sorunlarla karşılaşmışlardır. Aslında ikinci sorun ilk sorunun çözülmesindeki ilerlemeyi engellemektedir. Etkili bir değerlendirme yöntemi olmadan yeni bir sistemin sağladığı fayda ve oluşturduğu ek yük tam olarak değerlendirilememekte, bu nedenle de önerilen sistemin uygulanabilirliği test edilememektedir.

3.2.4. Dinleme saldırılarına karşı savunma

Dinleme saldırısı, ağ iletişimi güvenliğine yapılan en önemli saldırılardan biridir. Ağların statik yapısı bu saldırıyı kolaylaştırmaktadır. Saldırganlar ağdaki iletişim bağlantısını dinleyebilir, ağ durumunu izleyebilir ve hassas verileri çalabilir. Zhao ve diğerleri [138], dinleme saldırılarıyla mücadele için YTA tabanlı çift atlamalı iletişim (double hopping communication, DHC) yaklaşımı önermiştir. Önerilen yaklaşım, ağ paketlerindeki ve yönlendirme rotalarındaki uç noktaları dinamik olarak değiştirir. Trafik birden fazla akışa dağıtılır ve farklı rotalar boyunca iletilir. Ayrıca, saldırganların iletişim verilerini elde edememesi ve çözememesi için birden çok anahtardan gelen veriler birleştirilir. Yazarlar, DHC'nin dinleme saldırganlarının yükünü artırdığını ve iletişim verilerinden çözümlenmesini zorlaştırdığını dile getirmiştir. Duan ve diğerleri [139], bir üst ağda SMT kullanılarak hesaplanan RRM adlı proaktif bir teknik sunmuştur. RRM, bir ağdaki çoklu akışların rotalarını aynı anda keşif, gizlice dinleme ve DoS saldırılarına karşı savunmak için rastgele değiştirmeyi sağlar. Liu ve diğerleri [140], ağ keşiflerine karşı YTA tabanlı

bir mimari geliştirmek için RRM'yi kullanır. Hacim ölçümleri ve ağ trafiğinin karakteristik ölçümleri, yapılandırılmış bir entropi matrisi kullanılarak saldırıları tespit edilmeye çalışılır. RRM, ağ anomali saptamasının sonucuna veya önceden belirlenen bir sürenin dolmasından sonra tetiklenir. Rastgele bir yönlendirme rotasının oluşturulması, 0-1 sırt çantası problemi (0-1 knapsack problem) olarak geliştirilmiş bir karınca kolonisi algoritması kullanılarak gerçekleştirilir. Önerilen yöntemin keşif ve gizlice dinleme zorluğunu artırdığı ve DoS saldırılarının etkisini azalttığı ifade edilmiştir.

Furukawa ve diğerleri [141], yeni bir ağ adresi çeviri yöntemi uygulayarak ortadaki adam saldırılarını ve ağı gizlice dinlemeyi önleyebilecek başka bir ağ mekanizması önermiştir. Önerilen mekanizma, kullanıcı taleplerine göre güvenlik seviyesini güçlendirmek için bir YTA anahtarında gerçekleştirilen birden fazla eyleme dayanmaktadır. Ma ve diğerleri [142] ağ iletişiminde gizli dinlemeye karşı özel veri koruma sorununu ele almış ve protokol-habersiz iletme (protocol oblivious forwarding, POF) protokol özelleştirme yeteneğini kullanarak gizlice dinleme saldırılarını önlemek için hareketli bir hedef savunma yöntemi önermiştir. Özelleştirme; tam protokol yığınını, mesaj paketi oluşturmayı ve yönlendirme rotasını rastgele oluşturmayı içerir. Anahtarlar, iletişim sırlarını korumak için dinamik mesaj paketleme ve dinamik yönlendirme rotaları kullanır. Yazarlar, önerilen stratejilerin bir kulak misafiri olan saldırgan için mesaj içeriğini yeniden düzenleme zorluğunu artırdığını ve oturum mesajı paketlerini yakalama olasılığını azalttığını dile getirmiştir.

Ağ esnekliğini ve güvenliğini artırmak için akıllı şebekeler için YTA tabanlı mimarilere büyük ilgi duyulmaktadır. Germano Da Silva ve diğerleri [143], olası bir gizli dinleyicinin SCADA bileşenleri arasındaki iletişim akışlarını tamamen yakalamasını önlemeyi amaçlayan bir YTA uygulama katmanı bileşeni sunmuştur. Mekanizma, SCADA cihazları arasındaki iletişim yollarını sık sık değiştirmek için dinamik ve statik çoklu rota yönlendirme kullanır, SCADA ağlarındaki bilgilerin gizliliğini artırır ve saldırganların SCADA cihazları arasındaki akışları yakalamasını zorlaştırır. Yazarlar önerdikleri mekanizma ile deneysel bir değerlendirme yapmış ve mekanizmanın daha kısa ömürlü dinamik kurallar ile bir saldırganın akışları dinlemesini zorlaştırdığını ancak YTA kontrolcüsündeki yönetim yükünü artırdığını bildirmiştir.

Güçlü ve güvenilir kimlik doğrulama, potansiyel rakipleri dışarıda tutmak ve ağlara yapılan saldırıları önlemek için önemlidir. Villain ve diğerleri [144], YTA kontrolcüsü kullanarak bir kimlik doğrulama mekanizması önermiştir. İnternete erişmeye çalışan kimliği doğrulanmamış bir kullanıcının istekleri, HTTP yeniden yönlendirmesi kullanılarak bir portale yönlendirilir. Portal kullanıcının kimliğini doğruladıktan sonra YTA kontrolcüsü kullanıcı için akış kurallarını yükler.

Önemli Noktalar: Bu bölümde ele alınan çalışmalar Çizelge 3.6'da özetlenmiştir. İlgili çalışmalar genellikle çoklu rota ve yük şifreleme / değiştirme gibi teknikleri kullanarak dinleme saldırılarının önlenebileceğini savunmuştur. Hareketli hedef savunma stratejisi dinlemeye ve casusluğa karşı sıkça kullanılmaktadır. Bu bölümde analiz edilen yedi çalışmadan üçü hareketli hedef savunma tabanlı strateji izlemektedir.

YTA'da yönlendirme davranışını özelleştirmenin kolaylığı, özellikle çoklu rota yönlendirme çözümlerini uygulamak için kullanışlıdır. Kontrolcü, gerekli tüm kural değişikliklerini yönetir ve yönlendirme kesintisiz olarak devam eder. Yine de YTA'nın kısmen kullanıldığı karma ağlarda çoklu rota yönlendirme uygulamak hala zor olabilmektedir. Bu nedenle, literatürde karma ağlarda çoklu rota yönlendirmeye bağlı olan güvenlik çözümlerinin pratikliğini analiz eden daha fazla çalışmaya ihtiyaç vardır. Araştırmacıların, böyle bir durumda etkin çözüm üretebilmeleri için kısmi YTA kurulum ve tasarım yöntemlerini dikkate almaları gerekmektedir.

3.2.5. Zararlı yazılım, sosyal mühendislik ve web uygulama saldırılarına karşı savunma

Cabaj ve Mazurczyk [145], popüler fidye yazılımı CryptoWall'un davranış analizi sonuçlarına dayanarak fidye yazılımı için iki gerçek zamanlı etki azaltma yöntemi önermiştir. Ayrıca yazarlar, YTA kontrolcüsü ile iş birliği yapan bir uygulama sunmuş ve önerilen etki azaltma sisteminin kavram ispatı için uygulamayı değerlendirmiştir. Önerilen yöntemler, kurban ile komuta ve kontrol sunucusu arasındaki iletişimi aktarmak için kullanılan fidye yazılımı vekil sunucularının dinamik olarak kara listeleye alınmasına dayanır.

Çizelge 3.6. Dinleme saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
DhcFlower [138]	Önleme	Rastgele atlamalı uç ve rota periyodik olarak uygulanır.	ÇOKLU-ROTA, ADRESİ-DEĞİŞTİR	KNTL
[139]	Önleme	Bir akıştaki paketler, SMT kullanılarak periyodik olarak değiştirilen farklı rastgele yönlendirme rotalarından iletilir.	ÇOKLU-ROTA	KNTL
[140]	Önleme, Tespit, Azaltma	Bir akıştaki paketler, önceden belirlenen periyotlarda karınca kolonisi algoritması ile hesaplanan farklı rastgele yönlendirme rotalarından iletilir. Ağ trafiği özelliklerinin entropi matrisi, trafik anormalliğini tespit etmek için dalgacık dönüşümü ve temel bileşen analizi kullanılarak oluşturulur.	ÇOKLU-ROTA	KNTL
[141]	Önleme	YTA anahtarının içinde akış şifrelemesi, uzamsal rastgele karıştırma, parçalama, çoğaltma işlemleri gerçekleştirilir.	YÜKÜ-ŞİFRELE, ÇOKLU-ROTA, YÜKÜ-DEĞİŞTİR	UYG, ANAH
[142]	Önleme	Özel protokol oluşturulur ve standart protokol verileri özel protokole rastgele olarak dönüştürülür. Alıcılar standart protokolleri almak için rastgele seçimi kaldırabilir.	YÜKÜ-ROTA	ANAH
[143]	Önleme	Her anahtar, iletişim sırasında paketlerin sadece bir kısmını iletir.	ÇOKLU-ROTA	KNTL
[144]	Önleme	DNS istekleri birkaç saniye ara belleğe alınır ve her DNS isteği için kimlik doğrulama paketi gönderilir. İstemci kimlik doğrulama paketine yanıt verirse, DNS isteğine izin verilir.	AEK	KNTL

Ceron ve diğerleri [146], YTA kullanarak ağ ortamını zararlı yazılım eylemlerine karşı dinamik olarak yeniden yapılandırmak için Zararlı Yazılım Analiz Mimarisi (Malware Analysis Architecture, MARS) adlı bir mimari sunmuştur. Sunulan mimari, bir korumalı alan, bir YTA kontrolcüsü ve bir kaynak havuzundan oluşur. Korumalı alan, zararlı yazılımı çalıştırır ve İnternet'e ve / veya yerel ağ öğelerine bağlantılar gerçekleştirir. Ağ trafiği olay kayıtları kontrolcüdeki bileşenlere iletilir. Kontrolcü, yerel ağ akışlarını kaynak havuzundaki ilgili cihazlara ileterek veya yönlendirilerek önceden belirlenen politikaları uygulamak için akış özelliklerini (paket hızı, işlem hacmi vb.) yönetir. Kontrolcü ayrıca korumalı alanla etkileşime girebilir ve yeni bir analiz başlatabilir. Yazarlar, önerilen çözümün geleneksel çözümlerden daha fazla zararlı yazılım olaylarını tetikleyebileceğini belirtmiştir.

Hu ve diğerleri [147], solucanlara karşı kapalı döngü bir savunma sistemi olan YTA tabanlı Worm-Hunter'ı önermiştir. Worm-Hunter, YTA anahtarlarının akış tablolarını yönetir ve farklı ağ yapılarına sahip farklı bal küpü sistemlerini dinamik olarak dağıtır. Worm-Hunter anormal bir trafik tespit ederse, şüpheli trafiği bal küpüne yönlendirerek güvenli bir şekilde analiz edilmesini sağlar. Tüm saldırı davranışları daha detaylı analiz için sistemde kaydedilir. Daha yaygın analiz için Worm-Hunter ile aynı topolojiyle birden fazla bal küpü oluşturulabileceği belirtilmiştir.

Jin ve Wang [148], mobil zararlı yazılım saldırı türlerini şu şekilde sınıflandırmıştır: saldırıları yeniden paketleme ve güncelleme, indirme saldırıları, uzaktan kumanda ve bilgi toplama. Yazarlar birkaç mobil zararlı yazılım algılama algoritması önermişlerdir. YTA kullanarak bir zararlı yazılım algılama sistemi tasarlamışlar ve uygulamışlardır. Tespit algoritmaları; YTA kontrolcüsü içindeki kara liste, bağlantı başarı oranı, hız sınırlandırma ve toplama analizi gibi yaklaşımlar kullanılarak uygulanmıştır. Önerilen sistem, yalnızca bağlantı kurma paketlerini kullanarak gerçek zamanlı trafikten şüpheli ağ etkinliklerini belirleyip mobil zararlı yazılımları tespit etmektedir. Bir bağlantı kurulduktan sonra daha sonraki paketler doğrudan anahtardaki kurallara göre yönlendirilebilmektedir.

Masoud ve diğerleri [149], web oltalama saldırılarının kullanıcıların davranışlarına bağlı olduğunu, protokollere bağlı olmadığını ifade etmiştir. Oltalama saldırılarını engellemek ve yapay sinir ağı tabanlı oltalama önleme algoritması (phishing prevention algorithm, PPA) uygulamak için özel bir YTA kontrolcüsü önermiştir. Önerilen algoritma, yerel ağdaki oltalama testi web sayfaları kullanılarak test edilmiştir. Yazarlar, PPA'nın web sayfalarının

sahte sürümlerini algıladığını ve bu sayfaların gerçek sürümlerine erişime izin verdiğini belirtmiştir.

Chin ve diğerleri [150], oltalama saldırısı imzalarını sınıflandırmak ve gerçek zamanlı derin paket denetimi uygulamak için yapay bir sinir ağı modeli kullanarak PhishLimiter adlı başka bir oltalama algılama ve etki azaltma yaklaşımı önermiştir. YTA kontrolcüsüne gelen her paket için bir oltalama puanı hesaplanır ve sınıflandırıcıya gönderilir. Analiz zararlı etkinlik tespit ederse paketler karantina alanına yönlendirilir. PhishLimiter'in iki inceleme modu vardır; hızlı modda, paket hedefe iletilir ve bir kopyası denetim için saklanırken yavaş modda tüm paketler denetimin sonucunu bekler. Yazarlar PhishLimiter'i gerçek dünyada test edilmiş bir ortam ve e-posta kullanarak değerlendirmişler ve PhishLimiter'in %98,39'luk bir doğrulukla oltalama saldırılarını tespit edip azaltabileceğini ifade etmiştir.

Lim ve diğerleri [151], sunucuları botnet saldırılarından korumak için DDoS Engelleme Uygulaması (DDoS Blocking Application, DBA) adında bir şema önermiştir. YTA kontrolcüsünde çalışan DBA ile saldırı altındaki sunucu arasındaki iletişim kullanılır ve savunma OpenFlow ara birimleri aracılığıyla düzenlenir. Korumalı sunucular DBA ile güvenli iletişim kanalları kurar ve DBA'yı herhangi saldırı hakkında bilgilendirir. DBA, yönlendirilen adresi bilgilerini sunucuya iletir.

Shtern ve diğerleri [152], uygulama düzeyinde düşük ve yavaş DDoS (LSDDoS) saldırılarını azaltan bir referans mimarisi önermiştir. Referans mimari, LSDDoS algılama sensörü, otomasyon kontrolcüsü, korumalı uygulama, izleme sistemi ve köpekbalığı tankından oluşur. Korumalı uygulama, LSDDoS saldırılarına karşı savunulacak uygulamaları temsil eder. Algılama sensörü LSDDoS saldırılarını algılar ve otomasyon denetleyicisini tetikler, bu da köpek balığı tankı adında kısıtlı bir alan sağlar ve zararlı trafiği iletir. İzleme sistemi, performans metriklerini mimarideki bileşenlerden toplar. Yazarlar örnek alternatifler göstermek için performans modeline dayalı ve kullanıma hazır bileşen tabanlı kalıcı mimariler sunmuştur.

Önemli Noktalar: Bu bölümde incelenen sistemlerin özet tablosu Çizelge 3.7'de yer almaktadır. Literatürde bu kapsama giren çalışma sayısı azdır ancak savunma yaklaşımlarındaki çeşitlilik nispeten yüksektir.

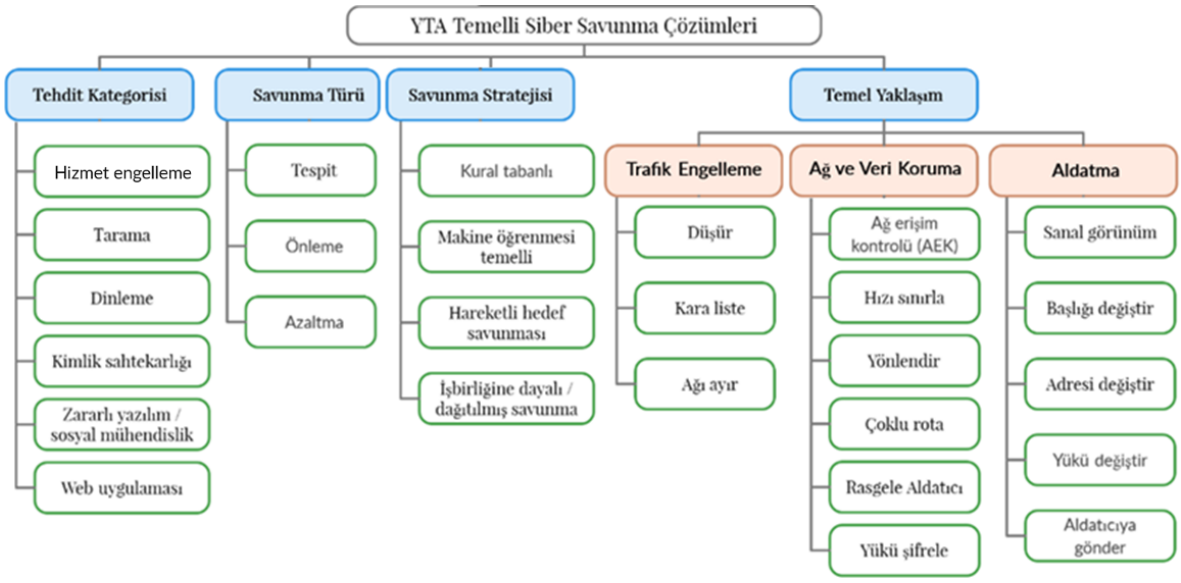
Çizelge 3.7. Zararlı yazılım, sosyal mühendislik ve web uygulama saldırılarına karşı YTA temelli savunma çözümleri

Çalışma	Savunma Türü	Önerilen Metodoloji, Teknik, Prosedür	Yaklaşım	Kurulum
[145]	Tespit, Azaltma	DNS isteklerinde kara liste veri tabanı kontrol edilir. Etkilenen istemcilerdeki tüm trafik engellenir.	AĞI-AYIR, KARA-LİSTE	KNTL, UYG
MARS [146]	Azaltma	Cihazlara bağlantı YTA kontrolcüsü tarafından yönetilir.	AĞI-AYIR	KNTL, UYG
Worm-Hunter [147]	Tespit, Azaltma	İstatistikleri kullanarak imza eşleştirme tespiti ve anomali tespiti yapılır. Şüpheli trafik yönlendirilir ve ardından dinamik sanal bal küpü içinde gözlenir.	ALDATICIYA-GÖNDER	UYG
[148]	Tespit, Azaltma	Zararlı yazılımları tespit etmek için IP kara listesi, bağlantı başarı oranı, hızı sınırlama ve trafik toplama yöntemleri kullanılır. Tespit edilirse ana bilgisayarın bağlantısı kesilir.	HIZI-SINIRLA, KARA-LİSTE	KNTL
[149]	Tespit, Azaltma	Oltalama önleme algoritması, geri yayılma sinir ağı modelini kullanır ve oltalama sitelerine erişmeyi engeller.	DÜŞÜR	KNTL
PhishLimiter [150]	Tespit, Azaltma	Kontrolcü akışlar için puanlama yapar ve puanları yönetir. Uygulama yapay sinir ağ modeli kullanarak akışları sınıflandırır.	AĞI-AYIR, DÜŞÜR	KNTL, UYG
DBA [151]	Tespit, Azaltma	Uygulamalara yapılan DDoS saldırıları tespit edilir ve DBA'ya bildirilir. Sunucu adresi dinamik olarak değiştirilir.	ADRESİ-DEĞİŞTİR	UYG, SUN
[152]	Tespit, Azaltma	Algılama sensörü, korumalı uygulamaların önüne yerleştirilir ve farklı algoritmalar kullanarak anormalliği algılar. Zararlı trafik, sağlanan bal küplerine yönlendirilir.	ALDATICIYA-GÖNDER	UYG

YTA uygulama katmanındaki tehditlere karşı savunma konularında yaratıcı çözümlerin sunulması gerekmektedir. STİ; zararlı yazılım, oltalama ve web uygulaması saldırılarına karşı çok önemli bir silahtır. Virüsten koruma yazılımı ve kişisel bilgisayar güvenlik duvarlarına ek olarak bu güvenlik sistemlerinin dinamik ve otomatik ağ savunma mekanizmalarıyla desteklenmeleri gerekir. Siber tehdit istihbaratı verilerini YTA ile entegre etmek, İnternette hızla büyüyen tehditlerden hem daha etkin şekilde korunmaya hem de daha hızlı yanıt verebilmeye yardımcı olacaktır [153].

3.2.6. YTA temelli siber savunma yaklaşımlarının değerlendirmesi

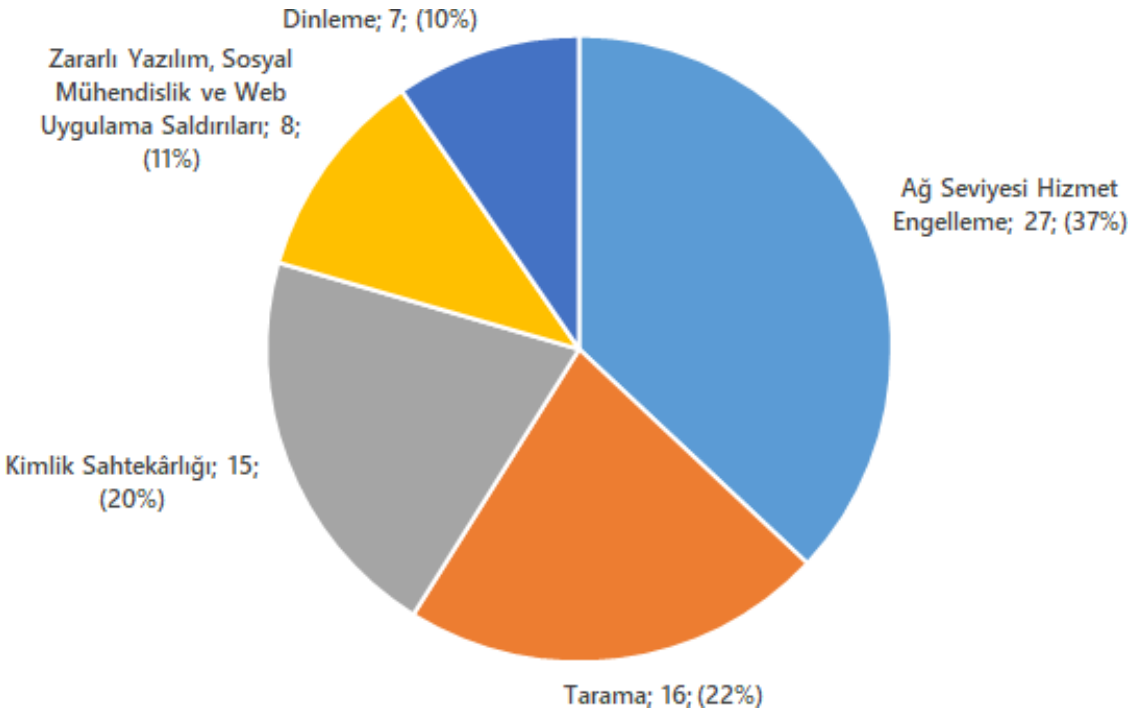
Bu bölümde tez çalışmasında belirlenen taksonomi ve literatürdeki çalışmalar ile ilgili çeşitli istatistikler sunulmaktadır. YTA temelli siber savunma çözümlerindeki çeşitliliği göstermek ve sınıflandırmak için oluşturulan taksonomi Şekil 3.2'de verilmiştir.



Şekil 3.2. YTA temelli siber savunma çözümlerinin taksonomisi

Taksonomi belirlenirken öncelikle literatürdeki çalışmaların siber tehditleri, saldırıları ve savunma yöntemlerini sınıflandırmak için kullandıkları yaklaşımlar incelenmiştir. Bu yaklaşımlarla birlikte STİ verileri ile paylaşılan tehdit türleri incelenerek tehdit kategorileri, savunma türleri ve savunma stratejileri belirlenmiştir. Ayrıca literatürdeki çalışmalarda YTA ağları kullanılarak sunulan çözümlerdeki temel savunma yapı taşları belirlenerek gruplanmış ve temel savunma yaklaşımları oluşturulmuştur.

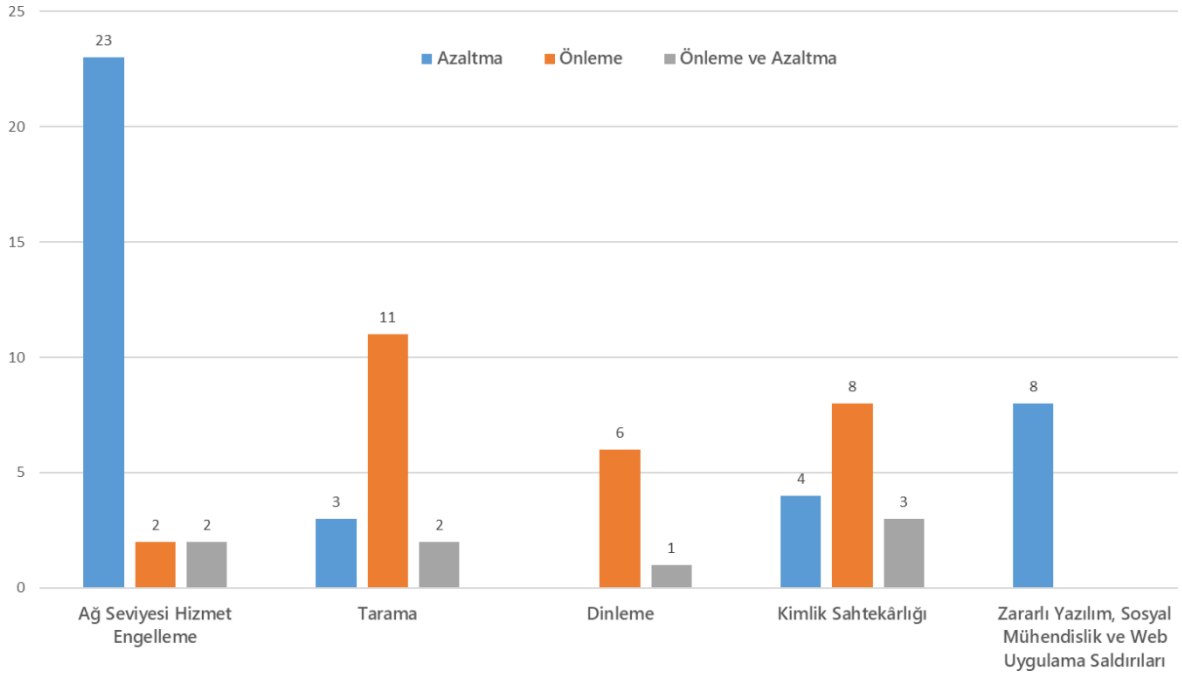
Literatürdeki çalışmaları siber tehdit kategorisine göre sınıflandırmak bu çalışmanın temel adımlarından biridir. Savunma türü olarak önleme ve etki azaltma çalışmalarına odaklanılmıştır. Bu tez çalışmasında siber tehditlere karşı YTA temelli önleme ve etki azaltma yaklaşımları öneren 73 çalışma gözden geçirilmiştir. Bahsedilen çalışmaların bazıları birden çok siber tehdit kategorisi içinde yer alabilmesine rağmen sadece bir ana siber tehdit kategorisinde özetlenmiş ve gerekli ise diğer tehdit kategorilerine referans verilmiştir. Şekil 3.3'te gözden geçirilen çalışmaların siber tehdit kategorilerine göre dağılımı gösterilmektedir. 73 çalışmanın %79'u temel olarak tarama, kimlik sahtekârlığı veya DoS saldırılarına odaklanmıştır. Çalışmaların sadece %21'inde dinleme, zararlı yazılım, sosyal mühendislik ve web uygulaması saldırılarına karşı yaklaşımlar önerildiği görülmüştür.



Şekil 3.3. YTA temelli savunma çözümlerinin tehdit kategorilerine göre dağılımı

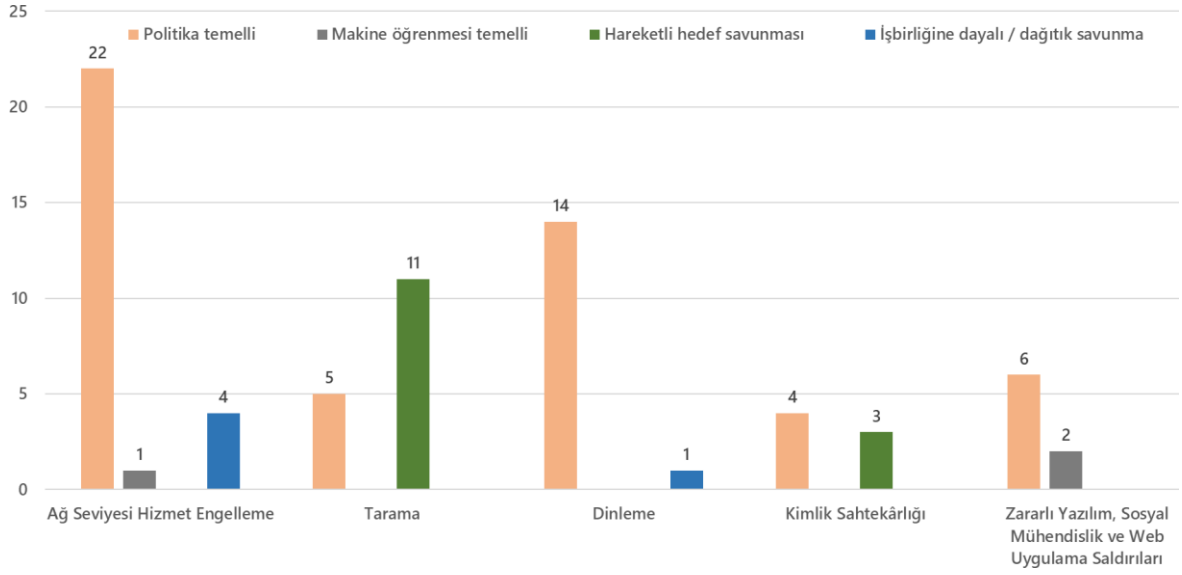
Çalışmalarda önerilen çözümlerin, savunma türlerine göre dağılımları Şekil 3.4'te gösterilmektedir. Önleyici savunma yaklaşımları öneren 27 çalışma vardır (tarama için 11, kimlik sahtekârlığı için 8, dinleme için 6, DoS saldırıları için 2 çalışma). Önleme yaklaşımları tarama, kimlik sahtekârlığı ve dinleme saldırılarına karşı yaygın olarak kullanılmıştır. 38 çalışma, siber tehditlere karşı savunma için etki azaltma yaklaşımları sunmuştur. Etki azaltma; DoS, zararlı yazılım, sosyal mühendislik ve web uygulaması saldırılarına karşı en yaygın kullanılan savunma türüdür. Ayrıca 8 çalışma tarama, kimlik

sahtekârlığı, dinleme ve DoS saldırılarına karşı hem önleme hem de etki azaltma yöntemleri önermiştir.



Şekil 3.4. YTA temelli savunma çözümlerinin savunma türlerine göre dağılımı

Şekil 3.5'te çalışmaların saldırı türüne ve kullanılan savunma stratejilerine dağılımı gösterilmektedir.

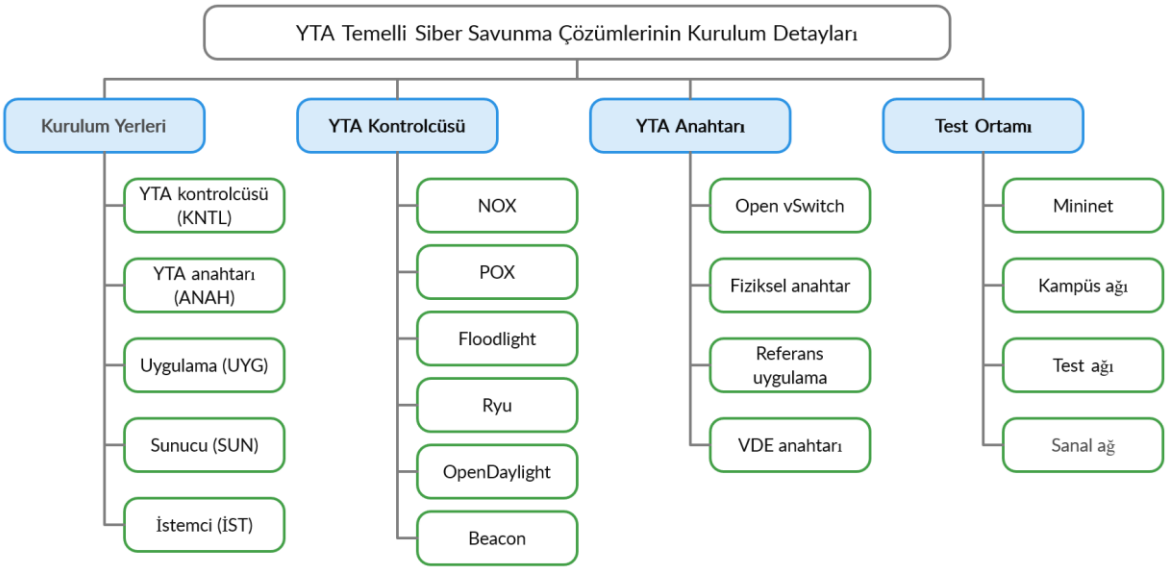


Şekil 3.5. YTA temelli savunma çözümlerinin savunma stratejilerine göre dağılımı

DoS, kimlik sahtekârlığı ve web uygulaması saldırıları için genel savunma stratejisi politika tabanlı stratejidir. Hareketli hedef savunma stratejisinin, genellikle kimlik sahtekârlığı ve dinleme saldırılarına karşı kullanıldığı görülmüştür. DoS ve web uygulaması saldırılarına karşı koymak için önerilen bazı çalışmalar, ağ trafiğini sınıflandırmak için makine öğrenimi tabanlı stratejiyi kullanmıştır. Son olarak, saldırılara karşı savunma yapmak için iş birliğine dayalı ve dağıtık savunma stratejisini kullanan çalışmaların sayısı son yıllarda artmıştır.

YTA temelli siber savunma çözümlerinde kullanılan altyapıların değerlendirmesi

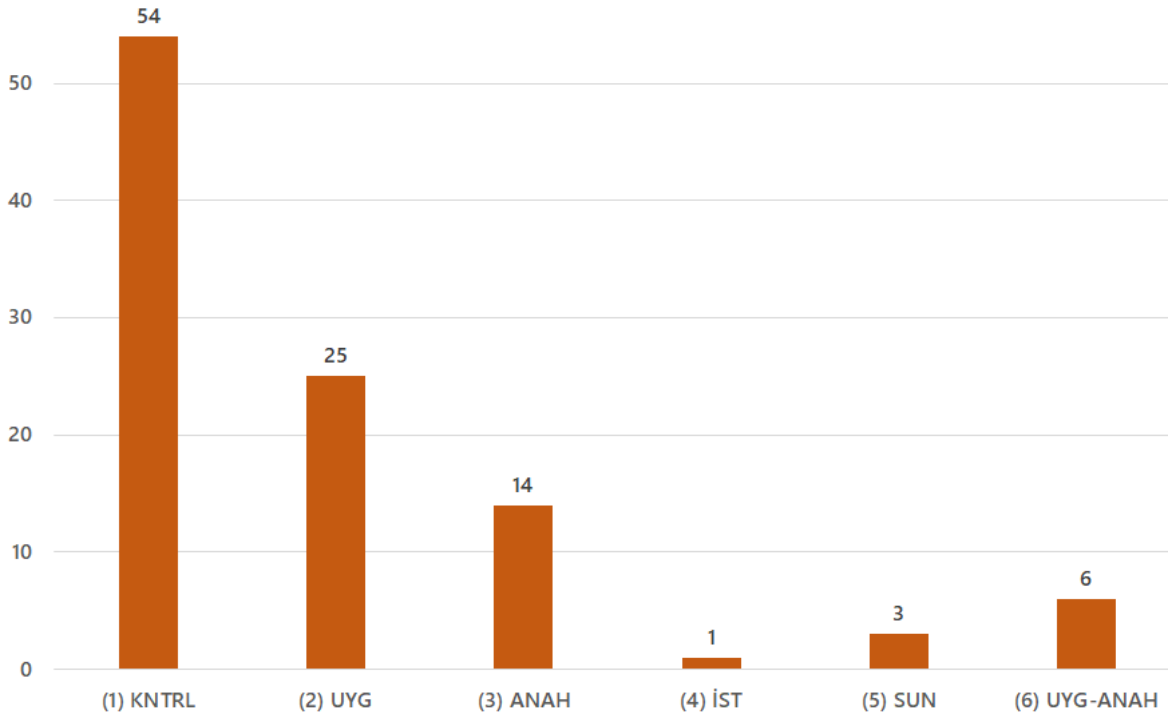
Literatürde, farklı YTA simülasyon platformları, anahtar türleri ve test ağı ortamları mevcuttur [154]. Çalışmamızda incelenen YTA tabanlı siber savunma çözümlerinin kurulumları ile ilgili taksonomi Şekil 3.6'da verilmiştir. Şekil 3.7'de YTA temelli savunma çözümü bileşenlerinin kurulduğu yerler ile ilgili sayısal bilgiler sunulmuştur. Şekil 3.7'de gösterildiği gibi YTA tabanlı savunma çözümleri en çok kontrolcüde (KNTL), uygulama katmanında (UYG) ve anahtarlarda (ANAH) değişiklik yapılmasına dayanmaktadır.



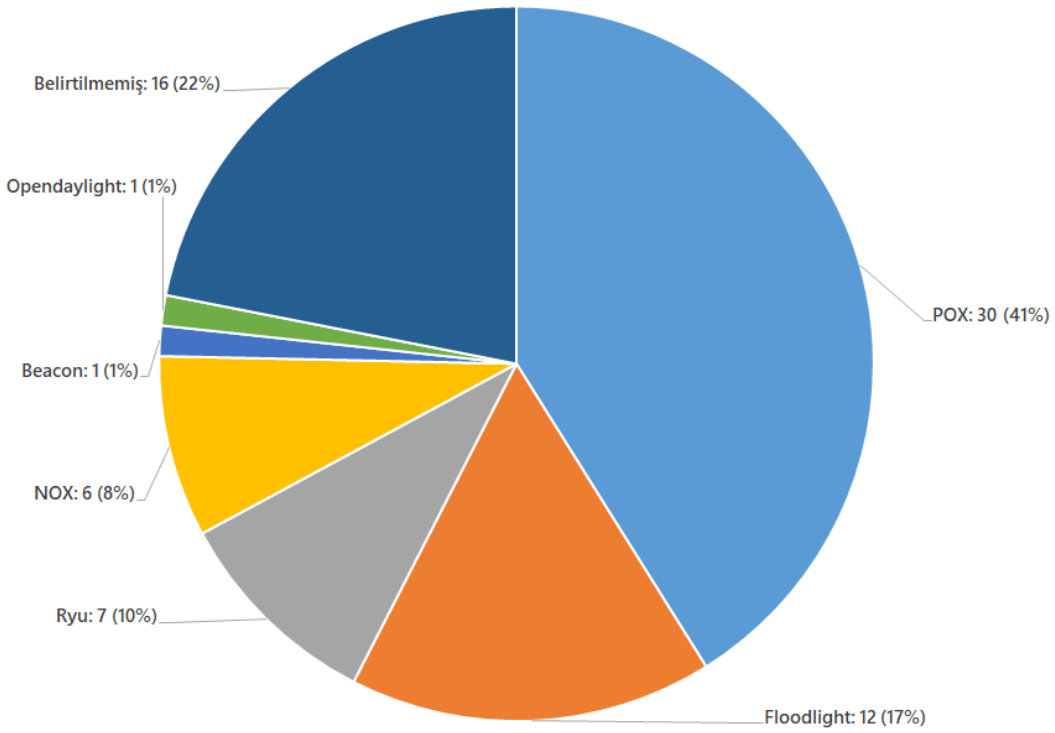
Şekil 3.6. YTA temelli savunma çözümlerinin kurulum taksonomisi

Literatürdeki savunma çözümleri birçok farklı YTA kontrolcüsü kullanılarak uygulanmıştır (Şekil 3.8). 16 çalışma kullanılan YTA kontrolcüsü hakkında hiçbir bilgi vermemiştir (%22). Buna rağmen, çalışmaların önemli bir kısmı (%41) önerilen yöntemleri uygulamak için POX

kontrolcüsünü [155] kullanmıştır. Ayrıca NOX [156], Ryu [157] ve Floodlight [127] en çok tercih edilen diğer YTA kontrolcüleridir.

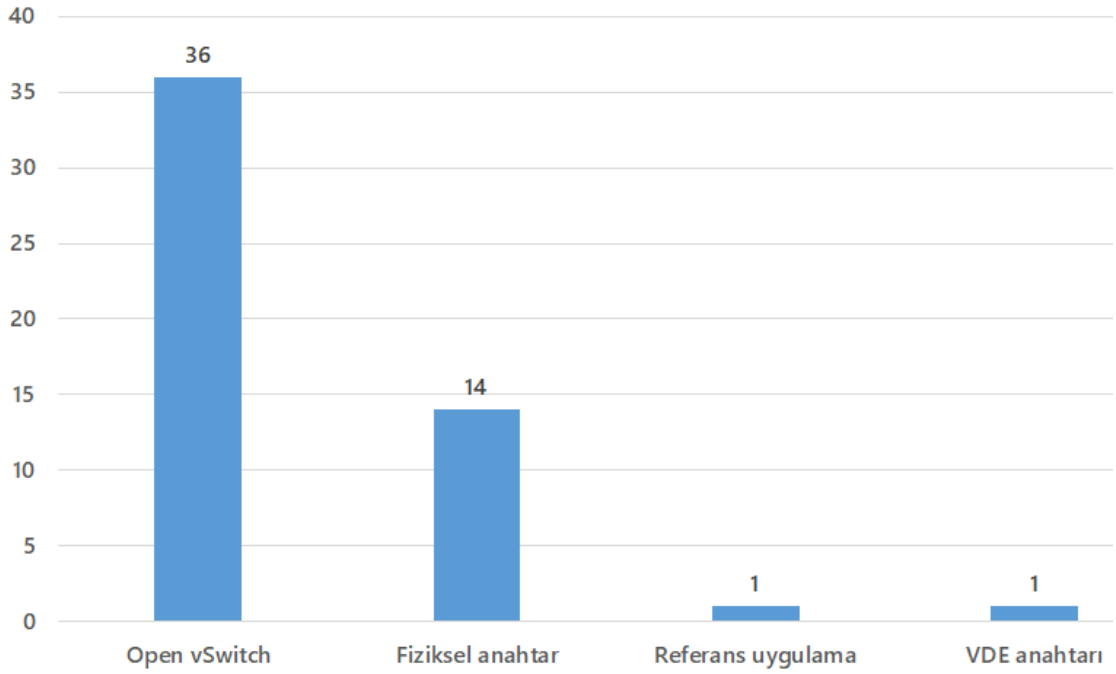


Şekil 3.7. YTA temelli savunma çözümlerinin kurulduğu katmanlar

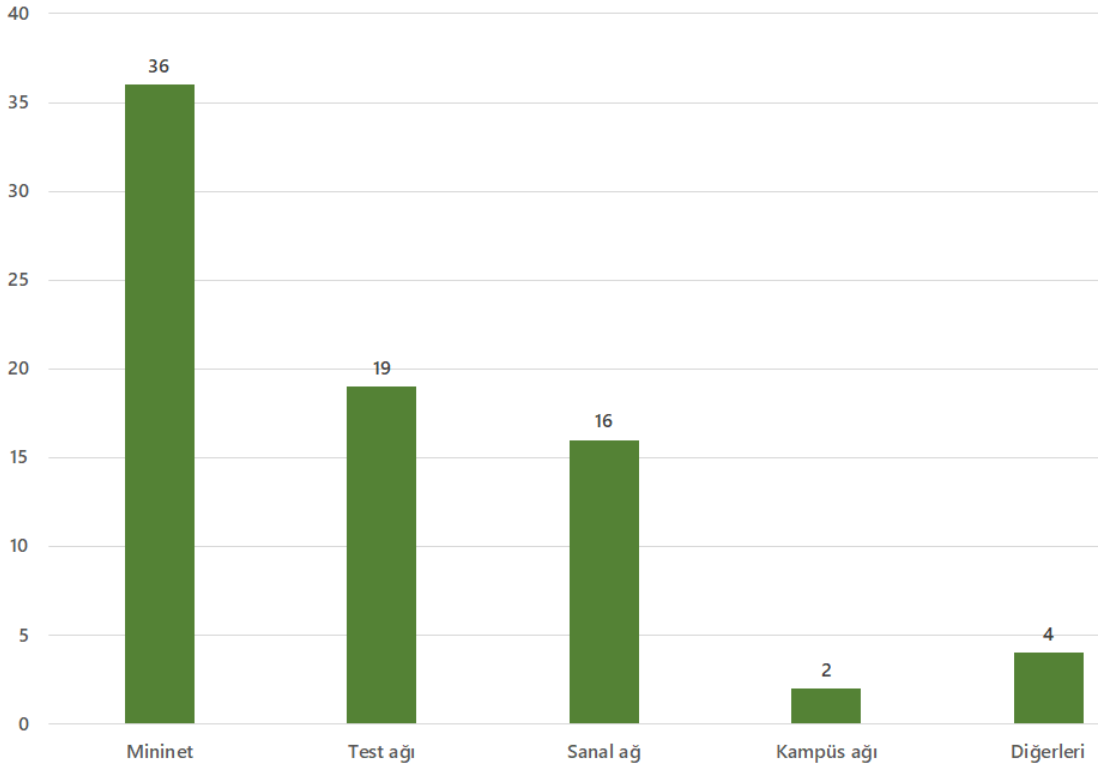


Şekil 3.8. YTA temelli savunma çözümlerinde kullanılan kontrolcülerin dağılımı

Önerilen savunma yaklaşımları genellikle YTA anahtarlı bir veya daha fazla test ağında değerlendirilmiştir. Çalışmalarda kullanılan anahtarların ve test ağlarının dağılımları sırasıyla Şekil 3.9 ve Şekil 3.10'da verilmiştir.



Şekil 3.9. YTA temelli savunma çözümlerinde kullanılan anahtarların dağılımı

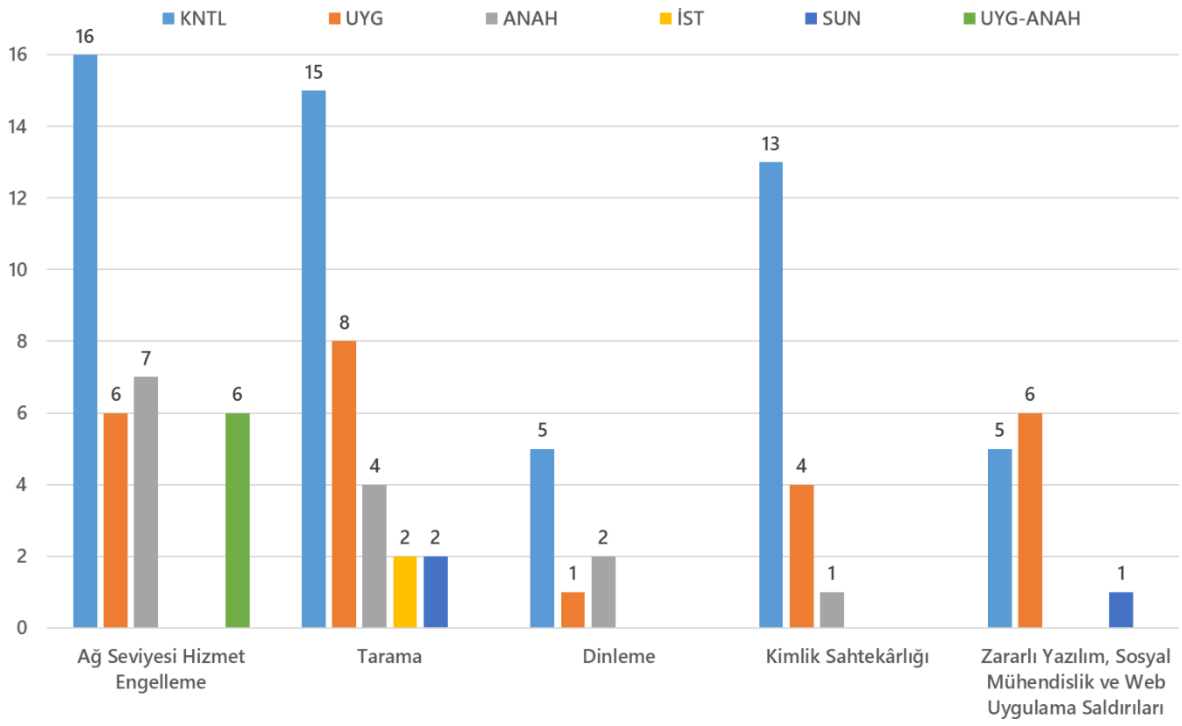


Şekil 3.10. YTA temelli savunma çözümlerinde kullanılan test ağlarının dağılımı

Çalışmaların yarısı (36 çalışma) önerilen savunma yaklaşımlarını test etmek için Mininet [158] altyapısı kullanmıştır. 19 çalışmada fiziksel makinelerle test ağları oluşturulmuş ve 16 çalışmada sanal makinelerle test laboratuvarları kurulmuştur. Sadece iki çalışma kampüs ağlarında yaklaşımlarını test etmiştir. Çalışmalarda YTA anahtarı olarak yaygın olarak Open vSwitch [159] (36 çalışmada) kullanılmıştır. 14 çalışmada önerilen savunma yaklaşımlarını değerlendirmek için OpenFlow destekli fiziksel anahtarlar kullanılmıştır.

YTA temelli siber savunma çözümlerinin kurulum konumlarının değerlendirmesi

İncelenen çalışmaların çoğu yeni bileşenler sunmuş ya da savunma yaklaşımları için mevcut YTA mimarisi bileşenlerini değiştirmeyi önermiştir. Bazı savunma yaklaşımlarını uygulamak için YTA mimarisine bir veya daha fazla bileşen yerleştirilmiştir. Her kurulum yeri için çalışmalar tehdit kategorisine göre sınıflandırılmış ve kurulum yerleri Şekil 3.11'de verilmiştir.



Şekil 3.11. YTA temelli savunma çözümlerinin tehdit türlerine göre kurulum yerlerinin dağılımı

Savunma yaklaşımına göre kurulum yerlerinin sayıları Çizelge 3.8'de verilmektedir. Önerilen çözümlerde *KNTL* ve *ANAH* kurulumlarının yaygın olduğu gözlenmiştir.

Çizelge 3.8. YTA temelli savunma yaklaşımlarının kurulum yerlerine göre dağılımı

Kurulum Yeri / Savunma Yaklaşımı	KNTL	UYG	ANAH	İST	SUN	UYG-ANAH	Toplam
DÜŞÜR	26	10	6	0	0	3	45
KARA-LİSTE	5	4	2	0	0	2	13
AĞI-AYIR	4	4	0	0	0	0	8
AEK	15	3	3	0	0	0	21
HIZI-SINIRLA	6	1	4	0	0	1	12
YÖNLENDİR	4	2	2	0	0	1	9
ÇOKLU-ROTA	4	1	1	0	0	0	6
RASTGELE-ALDATICI	4	3	0	0	0	0	7
YÜKÜ-ŞİFRELE	1	0	0	0	0	0	1
SANAL-GÖRÜNÜM	4	3	0	0	0	0	7
BAŞLIĞI-DEĞİŞTİR	3	3	1	0	0	0	7
ADRESİ-DEĞİŞTİR	8	4	2	2	3	1	20
YÜKÜ-DEĞİŞTİR	3	2	2	0		0	7
ALDATICIYA-GÖNDER	3	3	2	0	0	0	8
Toplam	90	43	25	2	3	8	-

Şekil 3.11 ve Çizelge 3.8 dikkate alınarak savunma yaklaşımlarının kurulumları ile ilgili inceleme sonuçları aşağıda listelenmiştir:

- *KNTL kurulumları*: Tarama, kimlik sahtekârlığı, dinleme ve DoS saldırılarıyla mücadele için en yaygın kurulum konumu YTA kontrolcüsü (*KNTL*) eklentisidir. *KNTL* kurulumları neredeyse tüm savunma yaklaşımlarında önerilmekte olup çoğunlukla *ADRESİ-DEĞİŞTİR, DÜŞÜR, SANAL-GÖRÜNÜM, YÖNLENDİR, ÇOKLU-ROTA* ve *RASTGELE-ALDATICI* yaklaşımları ile kullanılmaktadır.

- *UYG kurulumları*: Tüm siber tehdit türleri ve neredeyse tüm savunma yaklaşımları için özel YTA uygulama kurulumu (*UYG*) kullanılır.
- *İST ve SUN kurulumları*: Tarama ve web uygulaması saldırılarına karşı savunmak için bazı çalışmalarda ajan yüklemeleri gerekmektedir. *İST ve SUN* dağıtımları yalnızca *ADRESİ-DEĞİŞTİR* yaklaşımı için önerilmiştir.
- *ANAH kurulumları*: Bazı çalışmalarda standart YTA anahtar davranışının değiştirilmesi veya kuralların yönetimi önerilmiştir. YTA anahtar değişikliği kurulumu (*ANAH*) en çok tarama, dinleme ve DoS saldırıları için geçerlidir. Bu kurulum türü *DÜŞÜR* ve Ağ Erişim Kontrolü *(AEK)* savunma yaklaşımları için kullanılmaktadır.
- *UYG-ANAH kurulumları*: Bazı çalışmalar DoS saldırılarıyla başa çıkmak için YTA anahtarlarıyla doğrudan iletişim kuran bir YTA uygulaması önermektedir. Ayrıca, bu kurulum türü *DÜŞÜR* ve *AEK* savunma yaklaşımları için kullanılmıştır.

YTA temelli siber savunma çözümlerinde kullanılan savunma yaklaşımlarının değerlendirmesi

Siber tehdit kategorisine göre savunma yaklaşımlarının dağılımı Çizelge 3.9'da verilmiş ve sonrasında değerlendirilmiştir. Çoğu çalışma ağ seviyesinde hizmet engelleme, kimlik sahtekârlığı, tarama saldırıları üzerine odaklanmakta ve bu saldırılara karşı savunma için genellikle *DÜŞÜR, AEK, ADRESİ-DEĞİŞTİR* ve *YÖNLENDİR* yaklaşımlarını önermektedir. Farklı tehdit kategorileri için savunma yaklaşımlarının değerlendirme sonuçları aşağıda özetlenmiştir:

- Tarama saldırıları: Tarama saldırılarına karşı *ADRESİ-DEĞİŞTİR, SANAL-GÖRÜNÜM* ve *RASTGELE-ALDATICI* ana savunma yaklaşımlarıdır.
- Kimlik sahtekârlığı saldırıları: Kimlik sahtekârlığı saldırılarına karşı genellikle *DÜŞÜR* ve *AEK* savunma yaklaşımları uygulanmıştır. Bu saldırılarla mücadele için *YÖNLENDİR* ve *BAŞLIĞI-DEĞİŞTİR* savunma yaklaşımları da önerilmiştir.
- Ağ seviyesi DoS saldırıları: *DÜŞÜR* ve *YÖNLENDİR*, DoS saldırılarına karşı en yaygın kullanılan yaklaşımlardır.
- Dinleme saldırıları: *ÇOKLU-ROTA* yaklaşımının yanında *AEK, YÜKÜ-ŞİFRELE, ADRESİ-DEĞİŞTİR* ve *YÜKÜ-DEĞİŞTİR* bu saldırılara karşı kullanılmıştır.

Çizelge 3.9. YTA temelli savunma yaklaşımlarının tehdit türlerine göre dağılımı

Tehdit Türü / Savunma Yaklaşımı	Ağ Seviyesi Hizmet Engelleme	Tarama	Dinleme	Kimlik Sahtekârlığı	Zararlı Yazılım, Sosyal Mühendislik ve Web Uygulama Saldırıları	Toplam
DÜŞÜR	20	2	0	14	2	38
KARA-LİSTE	6	0	0	0	3	9
AĞI-AYIR	0	1	0	0	3	4
AEK (Ağ Erişim Kontrolü)	2	1	1	13	0	17
HIZI-SINIRLA	7	0	0	0	1	8
YÖNLENDİR	6	1	0	1	0	8
ÇOKLU-ROTA	0	0	5	0	0	5
RASTGELE-ALDATICI	0	4	0	0	0	4
YÜKÜ-ŞİFRELE	0	0	1	0	0	1
SANAL-GÖRÜNÜM	0	4	0	0	0	4
BAŞLIĞI-DEĞİŞTİR	3	1	0	1	0	5
ADRESİ-DEĞİŞTİR	1	6	1	0	2	10
YÜKÜ-DEĞİŞTİR	0	2	3	0	0	5
ALDATICIYA-GÖNDER	2	2	0	0	2	6
Toplam	47	24	11	29	13	-

- Zararlı yazılım ve sosyal mühendislik saldırıları: Zararlı yazılım ve sosyal mühendislik saldırıları için literatürde *DÜŞÜR*, *AĞI-AYIR*, *HIZI-SINIRLA* ve *ALDATICIYA-GÖNDER* savunma yaklaşımları kullanılmıştır.
- Web uygulaması saldırıları: Web uygulaması saldırılarına karşı *ADRESİ-DEĞİŞTİR* ve *ALDATICIYA-GÖNDER* savunma yaklaşımları kullanılmıştır.

DÜŞÜR savunma yaklaşımı, çoğunlukla kimlik sahtekârlığı ve DoS saldırılarına karşı kullanılmıştır. Bu yaklaşımın tarama, zararlı yazılım ve sosyal mühendislik saldırıları için de geçerli olduğu görülmüştür. Öte yandan bu yaklaşım daha kapsamlı bir savunma stratejisi için diğer yaklaşımlarla birlikte kullanılmalıdır, örneğin kimlik sahtekârlığı saldırılarına karşı *AEK* ve *DÜŞÜR* savunma yaklaşımları ile birlikte kullanılır. Ayrıca *YÖNLENDİR* ve *DÜŞÜR* yaklaşımlarının birlikte kullanılması genellikle DoS saldırıları için geçerlidir. *KARA-LİSTE* savunma yaklaşımı, beyaz liste yaklaşımının tamamlayıcısıdır. Bilinen siber tehdit istihbarat kaynaklarından bir siber tehdit listesi oluşturulabilir, YTA kontrolcüsü veya bir uygulama tarafından engellenmesi gereken bir listeye dönüştürülebilir. Kara listedeki tehditler, akış kuralları ile düşürme veya bal küpüne yönlendirme kurallarına dönüştürülür ve YTA anahtarlarına gönderilir. Bu yaklaşım hizmet engelleme, zararlı yazılım, sosyal mühendislik ve web uygulaması saldırıları için geçerlidir. Ancak, kara listenin birkaç binden fazla tehdit tanımını içerdiği durumlarda YTA anahtarlarındaki kısıtlar nedeniyle kara liste doğrudan uygulanamayabilir. Bunun en temel nedeni YTA anahtarlarının akış tablolarının sınırlı kapasiteye sahip olması ve genellikle akış tablolarına yalnızca binler mertebesinde akış kuralı yüklenebilmesidir. Öte yandan milyonlarca kara liste tehdit öğesi mevcut olabilir, bu nedenle YTA anahtarlarının akış tablolarında yüksek önceliğe sahip tehditler ele alınmalıdır.

AĞI-AYIR savunma yaklaşımı; tarama, zararlı yazılım ve sosyal mühendislik saldırılarına karşı uygulanabilir. Trafik kaynağı ile ilgili anormallikleri incelemek, ağın bir bölümünü izole etmek ve organizasyon politikalarını uygulamak için yaygın olarak kullanılır. Bu yaklaşım, istemci ağa bağlanmaya çalıştığında *AEK* yaklaşımıyla, trafiğinin davranışını analiz etmek için *ALDATICIYA-GÖNDER* ile birlikte uygulanır. Paketler bu yaklaşımla düşürülmez, bunun yerine ağ bileşenleri arasında bilgi paylaşmak için akışın her paketine bir etiket eklenir. Etiketli paketlere yalnızca YTA anahtarlarında ilgili kurallar yüklendikten sonra izin verilir. Bu sürecin işletilebilmesi için tüm ağ bileşenleri (anahtarlar,

yönlendiriciler vb.) etiketin nasıl çıkarılacağını ve etiketli bir paketin nasıl işleneceğini bilmesi gerekmektedir.

AEK savunma yaklaşımı, genellikle kimlik sahtekârlığı saldırılarına karşı kullanılır. Bu yaklaşımın tarama, hizmet engelleme ve dinleme saldırılarına karşı da uygulanıldığı görülmüştür. Ağ bileşenleri (istemciler, ağ cihazları vb.) ve erişim hakları, ilk isteği işleyen bir kimlik doğrulama ve yetkilendirme sunucusu olarak görev yapan YTA kontrolcüsü tarafından sağlanmalıdır. Geçerli ağ bileşenleri YTA ağ topolojisi kullanılarak doğrulanabilir veya bir kimlik doğrulama protokolü başlatılabilir. Bilinmeyen ağ bileşenlerinin ağa katılmasına izin verilmez ve ağda bilinmeyen herhangi bir trafik engellenir. Ağ erişim kontrolü farklı düzeylerde (kaynak MAC adresi, IP adresi, trafik rotası (kaynaktan hedefe) veya üst düzey protokolde (HTTP, SSH vb.) uygulanabilir. Bir kimlik doğrulama protokolü kullanılırken YTA kontrolcüsü veya bir uygulama, paketleri ara belleğe alır ve ara bellekte yalnızca kimliği doğrulanmış paketlerin yönlendirilmesine izin verilir. Bir kimlik doğrulama protokolü etkinleştirilirse her istemcinin kimlik doğrulama mekanizmasını bilmesi gerekmektedir. Yüksek performanslı ara bellek yönetim tekniklerine ihtiyaç duyulması nedeniyle bu yaklaşım istemcilerden bağımsız değildir.

HIZI-SINIRLA yaklaşımı; çoğunlukla hizmet engelleme, zararlı yazılım ve sosyal mühendislik saldırılarına karşı kullanılır. YTA kontrolcüsü, tıkanıklığı önlemek için trafiği veya akış iletim hızını sınırlar veya bulut bilişim özelliklerini kullanarak hizmet kalitesi gereksinimlerine uyacak şekilde kaynakları (örneğin, sanal makineler, sunucular) genişletir. Bu yaklaşımda trafik istatistikleri eşik değerlerine göre izlenmeli ve kontrol edilmelidir. Kaynaklar, rotalar, akışlar, hedefler için ayrı eşikler oluşturulmalıdır. *HIZI-SINIRLA* ile birlikte kötü niyetli trafiği önlemek için *YÖNLENDİR*, *KARA-LİSTE* ve *DÜŞÜR* yaklaşımları uygulanabilir. Kısa süreli dönemlerde sel saldırıları anahtar trafik istatistiklerinden tespit edilemediğinden, trafik istatistikleri YTA anahtarlarından periyodik olarak toplanmalıdır.

RASTGELE-ALDATICI savunma yaklaşımı, temel olarak tarama saldırılarına karşı kullanılır. Bir aldatıcı; bir bal küpü, ağ veya simüle edilmiş bir istemci olabilir. Herhangi bir tuzağa rastgele IP adresleri atanabilir ve tarama saldırıları için trafik izlenebilir. Bu yaklaşım, her istemcinin farklı sanal ağ görünümüne sahip olduğu *SANAL-GÖRÜNÜM* ile kullanılabilir. İstemciler ağa her bağlandığında ağ görünümleri değiştirilebilir. Zararlı ağ

taramaları, geniş aralıktaki alt ağ adres alanları kullanılarak önemli ölçüde geciktirilebilir. Ayrıca bu yaklaşımda değiştirilmiş DNS ve DHCP sunucuları kullanılabilir veya YTA kontrolcüsü, farklı ağ görünümlerini korumak için ağ geçidi, DNS ve DHCP sunucusu olarak çalışabilir. Sanal ağ görünümü veri tabanı korunmalı ve YTA kontrolcüleri arasında paylaşılmalıdır. Protokol davranışları (örneğin IP, ICMP, ARP) ve paket iletim gecikmesi, her ağ paketi için atanan sanal ağ görünümüne göre ele alınmalıdır. Ancak bu çözümler istemcide kullanılan statik ağ yapılandırmaları nedeniyle karma ağlar için tam olarak uygulanamaz. Saldırganlar ağ trafiğini dinleyerek gerçek ağ yapılandırmalarını öğrenilebilir. Ayrıca oluşturulan sanal ağ görünümleri arasındaki ilişkiler gerçek ağ yapılandırmasını keşfetmek için de kullanılabilir.

Paketi değiştirme yaklaşımları, yaygın olarak kullanılan savunma yaklaşımlarıdır. Tarama, dinleme ve web uygulaması saldırılarına karşı IP adresi değişikliği (*ADRESİ-DEĞİŞTİR*) savunma yaklaşımı kullanılır. YTA kontrolcüsü sanal IP adreslerini koruduğu ve düzenli aralıklarla değiştirdiği için bu yaklaşım istemcilerden bağımsızdır. Sanal IP adresi eşleme veri tabanı korunmalı ve YTA kontrolcüleri arasında paylaşılmalıdır. Özel bir DNS sunucusu, adların gerçek IP adresleri yerine sanal IP adreslerine dönüştürülmesini sağlar. Bu yaklaşımın dezavantajı, sanal IP elde etme işleminin ağda yüksek oranda ek yüke neden olabilmesidir. Ayrıca bu yaklaşım tarama trafiğinin kaynağını dinamik olarak algılayamayabilir. Diğer bir paket değiştirme yaklaşımı, tarama, kimlik sahtekârlığı ve DoS saldırılarına karşı kullanılan başlık güncellemedir (*BAŞLIĞI-DEĞİŞTİR*). Çoğunlukla parmak izi saldırılarını önlemek ve paketlere etiket atamak için kullanılır. Yükü şifreleme (*YÜKÜ-ŞİFRELE*) yaklaşımı ek bir yük oluşturmasına rağmen dinleme saldırıları için kullanılabilmektedir. Dinleme saldırılarına, özellikle parmak izi saldırılarına karşı bir başka savunma yaklaşımı *YÜKÜ-DEĞİŞTİR* yaklaşımıdır. *YÜKÜ-DEĞİŞTİR* veya *BAŞLIĞI-DEĞİŞTİR* savunma yaklaşımları ile ICMP, TCP ve uygulama düzeyinde protokollerin işlenmesi, işletim sistemi ve protokol davranışlarının analiz edilerek simüle edilmesi gerekmektedir.

3.3. Siber Tehdit İstihbaratı Verilerinin Üretilmesi, Paylaşılması ve Kullanılması

Siber tehdit istihbarat verileri çoğunlukla yapısal olmayan formattadır. Bu verileri yapısal formata dönüştürerek daha etkin güvenlik analizleri gerçekleştirilebilmektedir. Bu bağlamda literatürde çoğunlukla STİ verilerinden zararlı yazılımları tespit etmek ve bu yazılımların

özelliklerini çıkartabilmek için çalışmalar yapılmıştır. Jo ve diğerleri [160], yapısal olmayan formattaki açık kaynak STİ bilgilerini veri madenciliği yaklaşımları ile analiz eden, etiketleyen ve yapısal hale getiren GapFinder adında bir sistem önerisinde bulunmuştur. Sistem; veri çıkartma, ön işlemci, çizge oluşturucu, veri sorumlusu, veri analizcisi ve veri tabanlarından oluşan bileşenleri içermektedir. Çalışmada, 470 bin yapısal olmayan rapordan zararlı yazılımların dört ana özelliğini (ilk tespit tarihi, bulaşma yöntemleri, kullandığı zafiyetler, hedef platformlar) çıkartılmış ve yapısal formata dönüştürmüştür. Liao ve diğerleri [161] de benzer şekilde doğal dil işleme teknikleri kullanarak 71 bin açık kaynak teknik rapordan 900 bin OpenIoC [13] tehdit göstergesi oluşturmuştur. Kim ve diğerleri [162], STİ verilerinden zararlı yazılım, alan adı, IP adresi, özet ve zafiyet gibi bilgileri otomatik çıkaran bir yöntem sunmuş ve 498 bin etiketli veri üretmiştir. Zhu ve Dumitraş [163], 17 bin tehdit raporunu doğal dil işleme ve bilgi çıkarma teknikleri kullanarak analiz etmiş, tehditleri ve tehditlerin kullandıkları saldırı örüntülerini yapısal hale dönüştüren TTPDrill yöntemini önermiştir. Yazılımların kullandığı izinler, uygulama programlama ara yüzü çağrıları, işletilecek işlemlerin tanımları gibi Android zararlı yazılım özelliklerini yapısal olmayan verilerden çıkaran yöntem önermiştir.

Sabottle ve arkadaşları [164], bilinen zafiyetler için Twitter verileri kullanarak geliştirilen sömürüleri tespit eden bir analiz çalışması yürütmüştür. Zhao ve arkadaşları [165], günlük, bülten ve internet forumlarındaki verilerden STİ çıkaran ve analiz eden TIMiner adında bir çatı sunmuştur. Bu çalışmada elde edilen bilgiler evrimsel sinir ağları ile etiketlendirilerek sınıflandırılmıştır. Joshi ve diğerleri [166], çeşitli kaynaklardan aldığı yapısal olmayan zafiyet verilerinin arasındaki ilişkileri ve zafiyetin verilerinin yedi özelliğini (işletim sistemi, ağ bilgileri, saldırı yöntemi ve saldırı sonuçları, dosya ismi, donanım, isimli varlık tanımlayıcısı, teknik olmayan bilgiler) çıkartan çatı önermiştir. Jones ve diğerleri [167], yapısal olmayan zafiyet bilgilerini doğal dil işleme ve ön işleme algoritması kullanarak işleyen ve veriler arasında sekiz ilişki türünü çıkaran yöntem önermiştir.

Siber tehdit istihbaratı elde etmek veya bu verileri kullanmak için makine öğrenmesi ve yapay zeka teknolojilerinin kullanımı artmaktadır [168, 169]. Mittal ve diğerleri [170], güvenlik analistlerine yardımcı olmak ve güvenlik ile ilgili sorgular yapabilmek için yapay zeka temelli bir siber güvenlik bilgi sistemi önermiştir. Sistem çeşitli kaynaklardan (derin web, günlük, sosyal medya, rapor, gazete makalesi vb.) veri alarak vektör bilgi çizge formatında bilgi çıkartmaya çalışmaktadır. Koloveas ve arkadaşları [171] da benzer

kaynaklardan IoT ile ilgili STİ bilgileri çıkarmak için makine öğrenmesi destekli mimari önermiştir. Yapay zekanın siber güvenlik alanında kullanılması ayrıca güvenlik riskleri oluşturabilmektedir. Khurana ve diğerleri [172], yapay zeka temelli STİ sistemlerine karşı yapılan saldırıları önlemeye yönelik çalışma yürütmüştür.

STİ verilerinin kalitesi genellikle çok düşüktür [168, 173]. Siber tehdit istihbaratı bilgilerinin kalitesinin ölçülmesi ve problemlerin tespit edilmesi konusunda da çalışmalar yürütülmüştür. Bu kapsamda Schlette ve diğerleri [174], STIX STİ verilerinin kalitesini ölçmek için kalite boyutları belirlemiş ve bu boyutlara göre ölçüm yapabilen bir araç sunmuştur. Jo ve diğerleri [160], önerdikleri GapFinder sistemi ile siber tehdit bilgilerini çıkartırken aynı zamanda bu bilgiler arasındaki uyumsuzlukları tespit eden bir bileşen önermiştir. Iqbal ve Anwar [173], STIX formatındaki verileri analiz ederek verilerdeki eksiklikleri ve tam olmayan bilgileri raporlayan, STİ verisinin belirlenen kalite değerinin üstüne çıkarılmasına yardımcı olan bir çatı önermiştir.

Siber tehdit bilgileri elde edebilmek için ağların analizi ve tespit edilen zararlı trafiğin STİ verilerini destekleyecek şekilde oluşturulabilmesi için çalışmalar da yürütülmüştür. Al-Mohannadi ve diğerleri [175], zararlı trafiğin bal küpüne yönlendirilerek analiz edilmesini ve STİ bilgileri oluşturulmasını öneren bir sistem önermiştir. Bal küpündeki trafik analiz edilerek belirli eşik değerlerini aşan zararlı yazılımın davranışları çeşitli eylemlere göre (yetkili kullanıcı olarak bağlanma eylemi, yetkisiz bağlantı, yetkili kullanıcı ile bağlantı, kanal açılma hatası vb.) sınıflandırılmıştır. Arıkan [176], ağ trafiğinden veri madenciliği ile öğrenilen saldırı tiplerine göre zararlı trafiği sınıflandıran ve tespit edilen zararlı trafiği STIX formatında STİ verileri olarak paylaşan sistem önermiştir.

Siber tehdit istihbarat verilerinin güvenilir şekilde paylaşılması otomatik siber savunma için kritik hususların başında gelmektedir. Preuveneers ve diğerleri [177], STİ verilerini üreten ile kullanan arasında güvenli ve güvenilir dağıtık paylaşım çatısı önermiştir. Bu kapsamda siber tehdit istihbarat verilerinin güvenilirliğini sağlayacak blok zincir temelli çözüm önerileri [178–180] de sunulmaya başlanmıştır.

Literatürde siber tehdit istihbarat verilerinin kullanılması ve güvenlik politikalarının otomatikleştirilmesi hakkında çok az araştırma vardır. Amthor ve diğerleri [181], siber tehdit paylaşım platformları ile güvenlik politikası kontrol sistemleri arasında entegrasyon için

kavramsal bir tasarım önermiştir. Bu tasarımda tehditlere karşı savunma yaklaşımları, güvenlik mimarileri ve bilgi sunum gereksinimleri anlatılmıştır. Ayrıca yazarlar önerilerini zararlı yazılım ve DDoS saldırı senaryoları ile kavramsal olarak değerlendirmiştir. Çalışmada; otomatik savunma yapabilmek için gerekli veri yapılarının ve modellerinin olmadığı, tehdit türlerine ve risklere göre güvenlik çözümlerinin seçilebilmesi gerektiği ve güvenlik politikalarına uygun savunma stratejilerinin uygulanması gerektiği vurgulanmıştır.

Appala ve diğerleri [182], STİ verilerinin içinde otomatik savunma yapılabilmesi için STIX v1.2'ye eklenebilecek veri yapısı, STİ verilerinin güvenli paylaşımı için bir protokol ve savunma için alınabilecek ağ ve sistem seviyesi fonksiyonları önermiştir. Sung ve diğerleri [183], FS- OpenSecurity adlı güvenlik mimarisi modeli önermiştir. Model, yazılım tanımlı düzenleyici (YTD) ve SQUEAK adında iki katmandan oluşmaktadır. YTD, uygulama katmanında çalışmakta, tehdit istihbaratını kullanarak erişim kontrolü, veri koruma ve tehdit önleme görevlerini yürütmektedir. Yücel ve diğerleri [184], STİ verilerini kullanarak zararlı trafiğin bulut ortamında bulunan bal küplerine yönlendirilmesini sağlayan bir yöntem sunmuştur.

STİ verileri çoğunlukla mevcut ağ güvenlik çözümleri tarafından (güvenlik bilgisi ve olay yönetim sistemi, saldırı tespit sistemi) kullanılmaktadır [185]. Ancieta ve Rothenberg [186], STİ verilerini güvenilir kaynaklardan alarak göstergelerle STS alanlarına dönüştüren IntelFlow mimarisini önermiştir. Önerilen mimari ile saldırıların etkilerini azaltmak ve tehditleri önlemek için anahtarların akışları güncellenir. Bir saldırı tespit edilirse trafik engellenir ve bal küpüne yönlendirilir. Mimari, zararlı web sitelerinden gelen DoS saldırıları için değerlendirilmiştir. Garcia ve diğerleri [187], Kolektif İstihbarat Çerçevesindeki (Collective Intelligence Framework, CIF) [188] STİ verilerini kullanmak ve YTA anahtarlarını korumak için bir mimari önermiştir. Kolektif İstihbarat Çerçevesinde yer alan en yaygın tehdit istihbaratı türleri IP adresleri, etki alanları ve URL bilgileridir. Bu kapsamda bir YTA uygulaması geliştirilmiştir. Uygulama, alan adlarının ve URL'lerin IP adreslerini bulur ve YTA anahtarlarının akış tablolarını günceller.

3.4. Tez Çalışmasının Literatürle Karşılaştırması

Amthor ve diğerleri [181], gelecekte yapılması gereken çalışmalar olarak; otomatik savunma yapabilmek için gerekli veri yapılarının ve modellerinin belirlenmesi, tehdit türlerine göre

güvenlik çözümlerinin seçilebilmesi ve güvenlik politikalarına uygun savunma stratejilerinin uygulanması gerektiğini belirtmiştir. Bu tez kapsamında da belirtilen problemlerin çözümü için yenilikçi bir sistem önerisi sunulmuştur.

Appala ve diğerlerinin [182] sunduğu mimari çok genel olarak açıklanmış, mimaride geleneksel güvenlik cihazları kullanılarak gerçekleştirilecek fonksiyonlar belirtilmiştir. Küçük çapta uygulama ile kavramsal olarak değerlendirilmiştir. Bu tez kapsamında önerilen sistem, YTA'da ağ savunmasını otomatikleştirmek için STİ verilerine ve servis fonksiyon zincirleri gibi yeni nesil ağ teknolojilerine dayanmaktadır. Tez kapsamındaki çalışmaların ilk aşamalarında, STİ verilerini YTA ağ güvenliğinde kullanmak için bir kavramsal model [153] tasarlanmıştır. Çeşitli saldırılara karşı savunma yapmak için güvenlik fonksiyonlarının zincirler halinde eklenebildiği ve birbirine bağlanabildiği modüler bir yapıya sahip olacak şekilde model önerisi geliştirilmiştir. Tez çalışmasının sonunda da bu model geliştirilerek aşağıdaki kabiliyetlere sahip bir siber savunma sistemi mimarisi önerilmiştir:

- Saldırılar için ağ seviyesindeki eylemleri otomatikleştirmek için STİ verilerinde yer alması gereken bilgiler için veri modeli oluşturulmuştur.
- STİ verilerini, mevcut topolojiyi, organizasyonun varlıklarını ve mevcut savunma politikalarını dikkate alarak savunma servislerini seçebilmektedir.
- Siber tehditler için siber savunma servisleri ve sanal ağ fonksiyonlarını kullanarak SFZ oluşturmakta ve yazılım tanımlı ağlarda uygulayabilmektedir.

STİ verilerini kullanan çalışmaların bazılarında [186, 187] önerilen yöntemler saldırı tespit sistemlerinin yeteneklerine dayanmaktadır. Tez çalışmasında önerilen sistem içerisinde ek bir güvenlik önlemi olarak STS ile iletişim kuran ve anomalileri dinleyen bir bileşen yer almaktadır. Tez kapsamında önerilen sistem, bu özelliğinin yanında otomatik ağ savunması yapabilecek kabiliyete ve STS dışında birçok farklı servisi bünyesinde barındırabilecek modüler bir yapıya sahiptir.

Çizelge 3.10'da literatürdeki çalışmalar ile önerilen siber savunma sisteminin; ağ servisi oluşturma yöntemleri, ağ servisi çalışma yöntemleri, çalıştırmada gerçekleştirilen örnek servisler, belirlenen tehdit modelleri, STİ verisinden faydalanma durumu, SFZ desteği ve çalışmada kullanılan değerlendirme ölçütleri karşılaştırılarak özetlenmiştir.

Çizelge 3.10. Önerilen siber savunma sisteminin literatür ile karşılaştırması

Çalışma / Ölçüt	FRESCO [69]	Pyretic [70]	SPHINX [71]	SDNSOC [72]
Ağ servisi oluşturma yöntemi	Betik dili kullanılarak servisler oluşturulur. Paket düşürme, yönlendirme ve karantina eylemleri gerçekleştirilebilir.	Pyretic programla dili ile dinamik kurallar yazılır. Dilin desteklediği paket işlemleri gerçekleştirilir.	Karmaşık kural politikaları tanımlanarak işletilir. OpenFlow bir akış veya ölçüm değişikliği eylemi gerçekleştirilebilir.	Ağ fonksiyonlarını kullanarak SFZ oluşturulur. Güvenlik duvarı, web sunucu, yük dengeleyici gibi servisler ağ fonksiyonu olarak kullanılır.
Ağ servisi çalışma yöntemi	Olay odaklı çalışır.	Politikalar birleştirilerek çalıştırılabilir. Programlama dili kabiliyetlerine göre bir kerelik, periyodik veya olaya bağlı çalışabilir.	Paket geldiğinde, periyodik veya akış güncellemelerinde çalışır.	SFZ oluşturulduktan sonra zincirdeki tüm paketler ağ fonksiyonları tarafından işlenir.
Geliştirilen örnek servisler	Tarama algılama, ağ düzeyinde botnet analiz, zararlı yazılım algılama servisi	Tanımlanmamış	ARP zehirleme, Bağlantı Seviyesi Keşif Protokolü (Link Layer Discovery Protocol, LLDP) kimlik sahteciliği, kontrolcü DoS, ağ seviyesi DoS, tablo taşırma, anahtar kara deliği	Uygulama sunucusu, VPN sunucusu, güvenlik duvarı ve yük dengeleyiciden oluşan bir SFZ oluşturulmuştur.
Belirlenen tehdit modelleri	Zararlı tarama, botnet saldırganı, zararlı yazılım	Tanımlanmamış	Ağ topolojisi saldırganı, veri katmanı saldırganı	Tanımlanmamış
STİ verilerinden faydalanma	Yok	Yok	Yok	Yok
SFZ desteği	FRESCO içinde tanımlanan bileşenler programlama ile statik olarak birleştirilebilir.	Programlama ile paket adresleri değiştirilerek rotalar değiştirilmektedir.	Yok	Var
Servis Değerlendirme ölçütleri	Kaynak kod satır sayısı, kural oluşturma süresi, kural temizleme performansı	Tanımlanmamış	Saldırı tespiti, kullanılan algoritmadaki eşik değerinin hassasiyeti, akış tablosu taşması	SFZ için sanal ağ fonksiyonu çizgesi oluşturma süresi, çakışan kural sayıları, çakışan kuralları tespit süresi

Çizelge 3.10. (devam) Önerilen siber savunma sisteminin literatür ile karşılaştırması

Çalışma / Ölçüt	ANASTACIA [73]	Karmakar ve diğerleri [74]	Appala ve diğerleri [182]	Bu tez çalışması
Ağ servisi oluşturma yöntemi	AFS orkestrasyonu kapsamında tanımlanan ağ fonksiyonları kullanılabilmektedir. Otomatik veya bir kerelik güvenlik kuralları uygulanacak şekilde tanımlanabilmektedir.	Kural tanımlama politikaları çalıştırılarak uygulanmaktadır.	STIX verileri kullanılarak ağ ve uç nokta güvenlik servisleri oluşturulmaktadır.	Servisler sistem tarafından seçilir ve başlatılır. Servisler her hangi bir zamanda herhangi bir OpenFlow akış veya ölçüm değişikliği eylemi gerçekleştirilebilir.
Ağ servisi çalışma yöntemi	Ağ fonksiyonları kendisine gelen tüm paketler üzerinde işlem yapmaktadır.	Politikalar önerilen sistem tarafından uygulanmaktadır.	Saldırı tespit sistemi, alan engelleme / karantina, yetkilendirme, sanal yerel alan ağı oluşturma servislerinin sadece adları tanımlanmıştır.	Olay ve zaman odaklıdır.
Geliştirilen örnek servisler	ICMP sel saldırısı tespiti servisi	Zararlı yazılım tespit, kimlik sahtekârlığı tespit, topoloji zehirleme tespiti, imza	Genel tehdit önleme, engelleme ve savunma yaklaşımları anlatılmıştır. Prototip geliştirilmemiştir.	Karantina, kara liste, tuzak sisteme gönderme, çoklu rota
Belirlenen tehdit modelleri	Tekrarlama saldırısı, zararlı yazılım, ortadaki adam saldırısı, IoT botnet, IoT sıfırıncı gün saldırganı, SQL enjeksiyonu, dinleme saldıranı	Kontrolcüye hedef alan saldırılar, veri katmanı saldırıları, topoloji zehirleme	K&K sunucusu	DDoS saldırganı, oltalama saldırganı, kimlik sahtekârlığı saldırganı, dinleme saldırganı, zararlı yazılım
STİ verilerinden faydalanma	Var	Yok	Otomatik hareket tarzları kullanılması önerilmiştir. Muhtemel ağ fonksiyonları ve görevler belirtilmiştir. Savunma yöntemi detaylandırılmamış ve test edilmemiştir.	Önerilen veri modeline göre muhtemel hedefler için ağ savunma servisleri başlatılmaktadır.
SFZ desteği	Güvenlik politikaları AFS yapılandırmalarına ve görevlere dönüştürülür.	Yok	Yok	STİ verilerinden alınan soyut hareket tarzları ile uygun SFZ oluşturulur.
Servis Değerlendirme ölçütleri	Analiz ve tespit süresi, olay inceleme süresi, kural oluşturma süresi	Kural oluşturma süresi, CPU kullanımı, hafıza kullanımı, gecikme / seğirme	Tanımlanmamış	Ortalama kural oluşturma süresi, servisin kullandığı bant genişliği, dinlenen veri yüzdesi, paket kaybı yüzdesi

Literatürde belirli siber saldırı türleri için YTA temelli siber savunma önerileri yer almaktadır. Fakat STİ verilerini işleyerek ağ seviyesinde savunma servislerini otomatikleştiren ve yeni ağ teknolojilerini kullanan bütüncül güvenlik mimarisi önerisi bulunmamaktadır. Önerilen sistemin ağ savunma servisleri ile genişletilebilir olması ve birçok saldırı türüyle başa çıkma potansiyeli, literatürdeki diğer ağ savunma önerileriyle karşılaştırıldığında önemli yenilik sağlamaktadır.

Siber tehditlerin IP adresleri, bilinen STİ kaynaklarından alınabilir ve YTA kontrolcüsü veya bir uygulama tarafından kara listeye dönüştürülebilir. Kara listedeki tehditler, akış kurallarıyla düşürülebilir veya tuzak sisteme yönlendirilebilir. Miao ve diğerleri [115] çalışmalarında yönlendiricilerde kara liste yaklaşımını uygulamıştır.

Cabaj ve Mazurczyk [145], kurban ile komuta ve kontrol sunucusu arasındaki iletişimi engellemek için kullanılan fidye yazılımı vekil sunucularının dinamik kara listeye alınmasına dayanan yöntemler sunmuştur. Jin ve Wang [189], zararlı yazılım tespiti için bağlantı başarı oranı, bağlantı sınırlama ve toplama analizi ile IP kara listesini kullanmıştır. Kara listedeki bilgilerin güveni çok önemlidir. Zararlı İnternet adreslerinden ve alan adlarından korumak için blok zincir temelli mekanizmalar kullanılabilir. Bazı araştırmalar [132, 134, 135], tespit edilen zararlı IP adreslerini göndermek ve güvenilir kara listelere ulaşmak için Ethereum blok zinciri ve akıllı sözleşmeler kullanan mekanizmalar önermiştir.

Son zamanlarda görülen veya yüksek güven değerlerine sahip kara liste IP adreslerinin çoğu ağ seviyesinde düşürülebilir. Bu tez çalışmasında, kara liste öğeleri dinamik olarak yönetilmekte ve STİ verileri kullanılarak güncellenmektedir. Herhangi bir YTA anahtarı, güvenlik duvarı görevi görebilir ve derinlemesine savunmayı destekleyebilir. Kara liste servisi diğer savunma servisleri ile birlikte çalışabilir. Kara liste servis talebi ve tuzak sisteme gönderme servisi talebi aynı anda alındığında, YTA kontrolcüsü istekleri OpenFlow grup işlem demetleri ile birleştirir ve işlem demetinde yer alan işlemlerin paralel işletilebilmesi için anahtara akış kuralı eklenir. IP adresleri sürekli kara listede yer almayabilir. Bu nedenle kara liste öğeleri dinamik ve periyodik olarak güncellenmelidir. Tez çalışmasında önerilen kara liste servisi ile IP adresleri için düşürme kurallarının herhangi bir YTA anahtarına yüklenmesi ve belirli bir süre sonra silinerek tekrar güncellenmesi sağlanmaktadır.

Ağdaki kötü niyetli faaliyetleri önlemek için istemciler karantinaya alınabilir ve diğer istemcilerden izole edilebilir. Karantinaya alınan istemcilerden zararlı yazılımların temizlenmesi veya sistemin güncellenmesi için istemcilerin karantina yönetimi sunucularıyla iletişim kurmasına izin verilir. Cabaj ve Mazurczyk [145], çalışmalarında fidye yazılımı bulaşmış istemciler için karantina akış kuralları oluşturmuştur. Ceron ve diğerleri [146], ağ ortamını YTA kullanarak zararlı yazılım eylemlerine karşı dinamik olarak yeniden yapılandırmak için tuzak sistem içeren MARS adlı bir mimari sunmuştur. Tez kapsamında gerçekleştirilen çalışmada, karantina akışları Cabaj ve Mazurczyk'in çalışmasına [145] benzer şekilde oluşturulmuştur.

Ağ seviyesinde uygulanan aldatma sistemleri, kötü niyetli veya şüpheli trafiği analiz etmek için farklı tuzakları yönetebilir. Tuzak; bir bal küpü, ağ veya simüle edilmiş bilgisayar olabilir. Tuzak sisteme rastgele IP adresleri atanabilir ve atanan IP adresine ait tüm trafik izlenebilir. Shtern ve diğerleri [152] yaptıkları çalışmada, algılama sensörünü korunan uygulamaların önüne yerleştirmiştir. Sensör anormallik algıladığında, trafik bal küpüne yönlendirilir. Achleitner ve diğerleri [76] ile benzer bazı çalışmalar [77, 78], sanal görünümler oluşturmak ve rastgele bal küpleri dağıtmak için benzer ağ aldatma sistemleri önermiştir. YTA temelli solucanlara karşı kapalı devre bir savunma sistemi olan Worm-Hunter [147], YTA anahtarlarının akış tablolarını yönetir ve dinamik olarak farklı ağ topolojilerine sahip farklı bal küpü ağlarına dağıtır. Bu tez çalışmasında, zararlı trafik normal trafikten izole edilerek bal küpüne yönlendirilmektedir. Paketleri yeniden yönlendirmek için NSH protokolü kullanılmaktadır.

Jafarian ve diğerleri [107], ağın güvenlik, performans ve kalite gereksinimlerini karşılarken en uygun stratejiyi belirlemek için oyun teorisini ve kısıtlama memnuniyeti optimizasyonunu birleştiren RRM adı verilen çevik çoklu yönlendirme yaklaşımı önermiştir. Liu ve diğerleri [140] ayrıca bir 0-1 sırt çantası problemi olarak gelişmiş bir karınca kolonisi algoritması kullanılarak rastgele bir yönlendirme rotasının oluşturulabildiği bir mimari önermiştir. Zhao ve diğerleri [138], muhtemel rotaları oluşturan YTA tabanlı bir çift atlamalı yaklaşım önermiştir. Yaklaşımda trafik birden çok akışa dağıtılır ve trafik paketleri rastgele IP adresleriyle farklı rotalar üzerinden iletilir. Duan ve diğerleri [139], bir üst ağda SMT kullanarak rastgele rotaları hesaplayan ve değiştiren RRM adlı bir teknik sunmuştur. Germano Da Silva ve diğerleri [143], SCADA cihazları arasındaki iletişim rotalarını sık sık değiştirmek için dinamik ve statik çok rotalı yönlendirmeyi kullanan bir YTA uygulama

katmanı bileşeni sunmuştur. Tez kapsamında geliştirilen çoklu rota servisi, Zhao ve diğerlerinin [138] çalışmasında yer alan çok rotalı seçim yöntemi ile benzerdir. Çok rotalı savunma servisi yukarıdaki çalışmalarda belirtilen özelliklerden farklı olarak aşağıdaki kabiliyetlere sahiptir:

- Paketlerin IP adresleri, aynı IP adresine sahip dinlenmiş paketleri en aza indirmek için her anahtarda rastgele değiştirilebilir,
- Bir akışın gelen ve giden trafiği farklı rotaları izleyebilir. Saldırgan paketleri dinlediği zaman paketlerin cevaplarını her zaman izleyememektedir.

4. YTA İÇİN ÖNERİLEN SİBER SAVUNMA SİSTEMİ (Y-S3)

Bu bölümde, tez çalışması kapsamında önerilen YTA Temelli Siber Savunma Sistemi (Y-S3) detaylı olarak anlatılmaktadır.

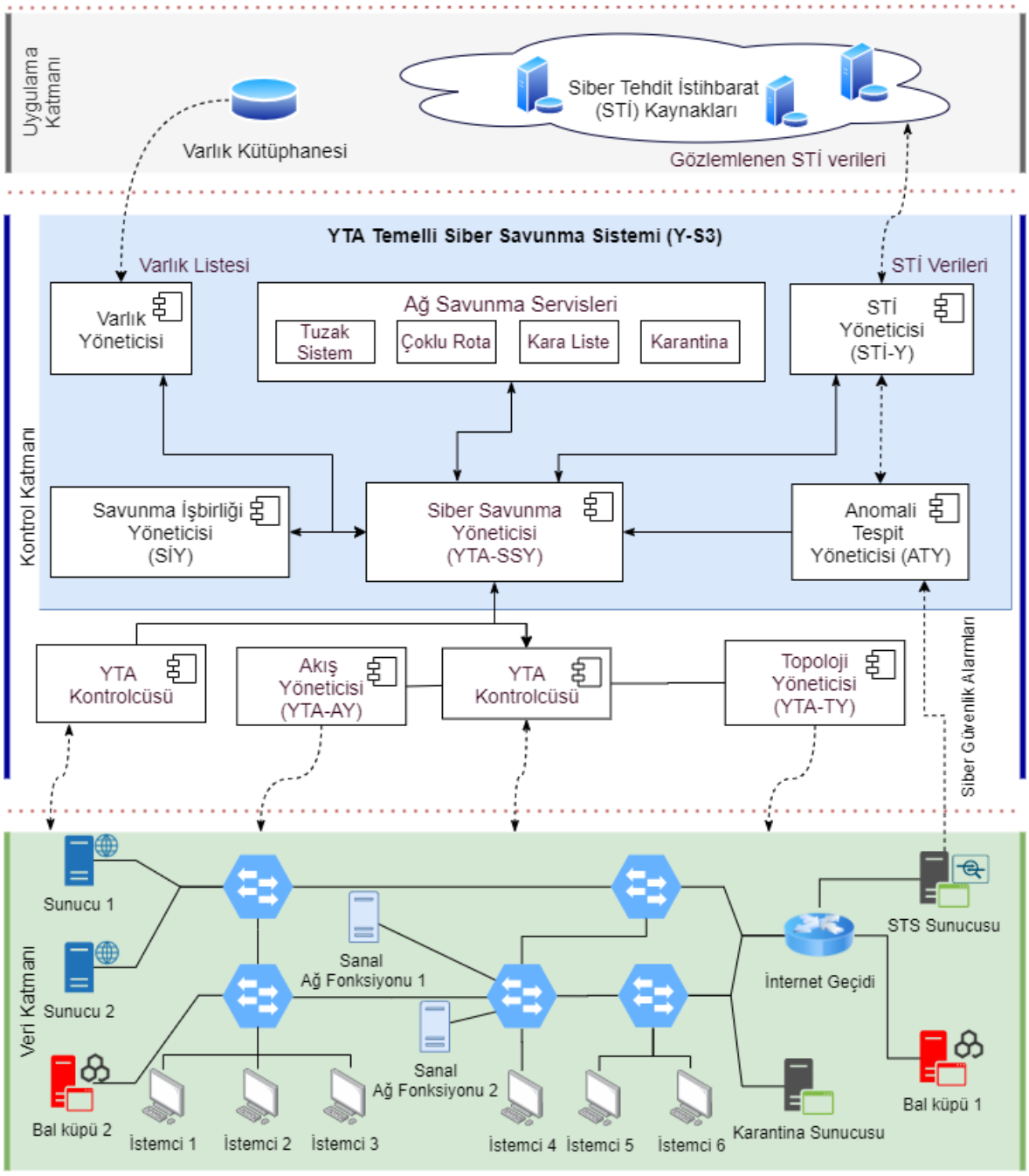
YTA Temelli Siber Savunma Sistemi (Y-S3); STİ verilerini, kurumsal varlıkları, ağ topolojisini ve mevcut güvenlik politikalarını dikkate alarak otomatik ağ seviyesi savunması gerçekleştirebilmek için önerilmiştir. Y-S3, aşağıda belirtilen on iki gereksinimi karşılayacak şekilde tasarlanmıştır:

1. Modüler ağ seviyesi savunma servisleri: Yeni bir siber savunma servisi eklemek için ilgili savunma servisinin kendi başına çalıştırılabilir veya test edilebilir olmasını sağlayacak şekilde sistem tasarımının yapılması gerekmektedir. Bu sayede farklı araştırmacıların geliştirdikleri servisler ile sistemin iletişim kurabilmesi için tasarımda gerekli ara yüz tanımlarının belirlenmiş olması gerekmektedir.
2. Genişletilebilir ağ seviyesi savunma servisleri: Ağ seviyesinde farklı siber savunma servislerine ihtiyaç duyulmaktadır. Tez çalışması kapsamında dört temel ağ savunma servisi geliştirilmiş olsa da sistemin farklı savunma servisleri ile birlikte çalışabilmesi gerekmektedir. Bu kapsamda sistemin ara yüzleri ve bileşenleri, savunma yaklaşımlarının (Bkz. Çizelge 3.2) ağ savunma servisi olarak eklenebilmesini sağlayacak şekilde olması gerekmektedir.
3. Farklı ağlarda kurulu sistemlerin iş birliği içinde çalışabilmesi: Siber savunma yaparken veya siber tehdit ile ilgili bilgiler güncellendiğinde farklı YTA ağlarında kurulu diğer sistem ile iş birliği yapılabilmesini sağlayacak altyapı sunulması ve gerekli ara yüzlerin tanımlanması gerekmektedir.
4. Farklı YTA ağlarında çalışabilmesi: Sistemin, çalıştığı ortamdaki ağ topolojisini ve servislerini otomatik olarak tespit ederek yönetebiliyor olması gerekmektedir. Sistemin herhangi bir YTA ağ topolojisine veya servislere doğrudan bağımlılığı olmaması, ağ ile iletişimde YTA kontrolcüsünün güney yönlü ara yüzünü kullanması gerekmektedir.
5. Ağ Savunması için akış kurallarının merkezi yönetimi: Birden fazla servis YTA anahtarlarına yüklenmesi için birden fazla kural üretebilmektedir. Savunma için üretilen tüm kuralların bir merkez üzerinden yönetilmesi ile servislerin diğer servisler tarafından üretilen kurallar hakkında bilgi edinebilmesi, anahtarlardan silinen kuralların her servise

iletilebilmesi, aynı akış için birden fazla kuralın birleştirilerek yüklenebilmesi ve kurallar arasındaki çakışmaların tespit edilebilmesi sağlanmalıdır.

6. Savunma servislerinin olay ve zaman temelli çalışabilmesi: Siber savunma servislerinin, belirli bir olay tetiklendiğinde veya başladıktan sonra belirli zaman aralıklarında otonom olarak çalışabilmeleri sağlanmalıdır.
7. Savunma akış kurallarının sürekli olarak kontrol edilmesi: YTA ağlarında oluşturulan siber güvenlik kurallarının zamana bağlı olarak güncellenebilmesi veya silinebilmesi gerekmektedir. Oluşturulan tüm kurallarda zaman aşımı değerlerinin belirlenmesi ve zaman aşımı değerleri ile YTA anahtarlarına yüklenmesi gerekmektedir.
8. Siber tehdit bilgilerinin proaktif yönetimi: Siber tehdit istihbaratı ile ilgili bilgiler işlenerek saldırı tespit edilmeden gerekli ağ savunma servislerinin başlatılabilmesi gerekmektedir. Siber tehdit bilgileri, siber tehdit istihbarat veri kaynaklarından alınabileceği gibi anlık olarak farklı ağların (organizasyonlar, çoklu ağ alanları vb.) bilgi paylaşmasıyla da elde edilebilmelidir.
9. Farklı kaynaklardan STİ verisi alınabilmesi: Farklı kaynaklardan farklı STİ veri formatını alarak işlem yapılabilmesini sağlayacak altyapı sunulmalıdır. Bu kapsamda STİ veri kaynaklarının; kurum içi kaynaklar, açık veya ticari kaynaklar veya iş birliği yapılan veri paylaşım ağları ile elde edilebileceği düşünülmelidir.
10. Tespit edilen saldırı ve tehditlerin STİ verisi olarak paylaşılabilmesi: Sistemin tespit ettiği tehditlerle STİ verisinin üretilebilmesini ve bunun paylaşılmasını sağlayacak bir altyapı sunulmalıdır.
11. Ağdaki siber saldırıların tespit edilmesi ve azaltılması: Ağ paketleri YTA kontrolcüsünde analiz edilirken paketlerin saldırı olup olmadığının anlaşılabilmesi için savunma servisleri devreye girebilmelidir. Bu işlem sonunda saldırı olarak tespit edilen akışlar için ilgili ağ savunması devreye alınabilmelidir. Ayrıca sistem, ağda bulunan saldırı tespit sistemleri ile entegre çalışabilmelidir. Bu kapsamda saldırı tespit sistemlerinin üreteceği tüm uyarılar anlık olarak izlenebilmelidir.
12. Saldırıların hedeflerinin tespit edilmesi ve uygun ağ savunmasının yapılması: Gerçekleştirilen tüm saldırılar farklı kaynakları hedef almaktadır. Ağın tümüne hizmet eden kaynaklar için tüm ağda, bir alt ağı tehdit eden saldırılar için o alt ağda, bir uygulama servisi veya istemciyi tehdit eden saldırılar için o istemcinin trafiğinde savunma yapılması gerekmektedir. Tehditler için muhtemel hedeflerin ve bu hedeflerin görevlerinin, zafiyetlerinin ve erişim bilgilerinin yönetilebilmesi sağlanmalıdır.

Y-S3'ün yazılım mimarisi ve bileşenleri belirtilen bu gereksinimler dikkate alınarak tasarlanmıştır. Şekil 4.1'de Y-S3'ün bileşenleri, iletişim kurulan dış sistemler ve YTA katmanları dikkate alınarak gösterilmektedir. Ağ katmanında STS, bal küpü ve farklı amaçla kullanılan sanal ağ fonksiyonlarının yanı sıra istemciler ve sunucular ağa bağlı olarak gösterilmiştir. Uygulama katmanında kurumsal veri kütüphanesi ve STİ veri kaynakları etkileşimde bulunulan sistemler olarak belirlenmiştir.



Şekil 4.1. Önerilen YTA Temelli Siber Savunma Sisteminin (Y-S3) genel görünümü

Şekil 4.1'de gösterilen sistemin bileşenleri ve temel görevleri aşağıda özetlenmiştir:

1. YTA Kontrolcüsü: OpenFlow protokolünü kullanarak YTA anahtarları ile iletişim kurar ve asenkron mesajları dinler.
2. Topoloji Yöneticisi (YTA-TY): Ağ topolojisini öğrenir ve topolojiyle ilgili OpenFlow mesajları gönderir.
3. Akış Yönetici (YTA-AY): Akışların durumu hakkında kontrol katmanı bileşenlerini bilgilendirir, akışları ve akış istatistiklerini yönetmek için akışla ilgili OpenFlow mesajları gönderir.
4. Anomali Tespit Yöneticisi (ATY): Ağda kurulu STS alarmlarını dinler.
5. Varlık Yöneticisi: Varlık kütüphanesinden ağ katmanında korunabilecek varlıkları sorgular ve yönetir.
6. STİ Yöneticisi (STİ-Y): Siber tehdit istihbaratı verilerini güvenilir kaynaklardan alır ve ağ seviyesinde uygulanabilir olanları filtreler.
7. Siber Savunma Yöneticisi (YTA-SSY): STİ verilerini, varlıkları, topolojiyi ve mevcut güvenlik mimarisini kullanarak savunma servislerini seçer.
8. Savunma İş Birliği Yöneticisi (SİY): Başka ağlarda kurulu sistem ile iletişim kurarak tehdit bilgisi paylaşımı yapar ve ağ savunması için savunma servisi başlatma talepleri iletir. Bu şekilde gelen taleplerin gereğini yerine getirir.
9. Siber Savunma Servisleri: Birden çok servisten oluşur. Savunma servisleri, savunma eylemlerini zamana bağlı olarak veya yeni akış algılandığında gerçekleştirirler.

Y-S3'teki bileşenlerin, belirlenen gereksinimlerinin hangilerini ne oranda karşıladığı Çizelge 4.1'de gösterilmiştir. Bir bileşen bir gereksinimi karşılaması için geliştirildiyse bu durum çizelgede (**) işareti ile gösterilmiştir. Bir gereksinimin karşılanması için bir bileşenden yararlanılmaktaysa bu durum da çizelgede (*) işareti ile ifade edilmiştir.

4.1. Y-S3 Bileşenleri

Bu bölümde, Y-S3'ün bileşenleri sırası ile detaylandırılmıştır. Her bir bileşen için tanımlanan her bir alt bölümde; bileşenlerin görevleri, bileşenlerin tasarımda dikkate alınan hususlar, görevlerini yerine getirirken gerçekleştirdiği işlemler ve diğer bileşenlerle etkileşimleri açıklanarak Y-S3'ün kabiliyetleri özetlenmektedir.

Çizelge 4.1. Y-S3 bileşenlerinin karşıladığı gereksinimler

Gereksinim No / Bileşen	1	2	3	4	5	6	7	8	9	10	11	12
YTA Kontrolcüsü			*	**	*	*		*			**	
Topoloji Yöneticisi (YTA-TY)			**	**								
Akış Yöneticisi (YTA-AY)			**		**	**	*	*				
Anomali Tespit Yöneticisi (ATY)										*	**	
Varlık Yöneticisi								*				**
STİ Yöneticisi (STİ-Y)								**	**	**		
Siber Savunma Yöneticisi (YTA-SSY)	**	**			*			**			**	**
Savunma İş Birliği Yöneticisi (SİY)			**					**				
Siber Savunma Servisleri	**	**	**		*	**	**	**			**	**

4.1.1. YTA Kontrolcüsü

YTA Kontrolcüsü, veri katmanında yer alan YTA anahtarlarını dinleyerek ve onlara gerektiğinde kurallar ekleyerek ağları yönetmekle görevlidir. YTA Kontrolcüsü, kontrol katmanında üç ana bileşenle haberleşir: Topoloji Yöneticisi (YTA-TY), Akış Yöneticisi (YTA-AY) ve Siber Savunma Yöneticisi (YTA-SSY). YTA kontrolcüsü, YTA anahtarlarından gönderilen anahtar, bağlantı noktası ve bağlantı yapılandırma olaylarını dinler ve bu olayları ağ topolojisini keşfetmek için Topoloji Yöneticisine (YTA-TY) aktarır. Ayrıca akış ile ilgili olaylar da YTA Akış Yöneticisine (YTA-AY) devredilir. YTA kontrolcüsü, ağ seviyesinde siber savunma operasyonlarını yönetmek için her gelen paket mesajını YTA-SSY ile paylaşır. YTA-SSY paketi işledikten sonra paketin düşürülmesine ve akışın engellenmesine karar verdiği durumda YTA kontrolcüsü paket için paket çıkışı mesajları göndermeyerek akışı engeller.

4.1.2. Topoloji Yöneticisi (YTA-TY)

Topoloji Yöneticisi (YTA-TY), ağ için anahtarları ve bağlantılarını içeren topoloji çizgesi oluşturur ve yönetir. YTA-TY, çoklu rota, yük devretme vb. gelişmiş işlemleri desteklemek için ağdaki döngüsel bağlantıları çözer ve yönetir. Topoloji bilgisi değiştiğinde, güncellemeleri çizgeye aktarır ve ağdaki döngüsel bağlantıları kontrol eder. YTA-TY temel olarak yeni anahtarı bağla, anahtarı iptal etme, bağlantı noktası açıklama durumu, yeni bağlantı ve bağlantı silme olaylarını dinler ve bu olaylara cevap verir.

YTA-TY, topoloji çizgesini her yeni anahtar bağlandığında günceller, bağlantı noktalarının özelliklerini öğrenmek için bağlantı noktası açıklama durumu isteği gönderir. Bağlantı noktası açıklama durumu yanıt mesajı alındıktan ve işlendikten sonra bağlantı noktası özelliklerini (bağlantı noktası numarası, tür, durum, desteklenen hız, mevcut hız vb.) topoloji çizgesine ekler. Topoloji döngüsel bağlantılar içeriyorsa, döngülerden uzak durulması için kapsama ağacını hesaplar. Her bağlantı ve bağlantı noktası mesajı alındığında, kapsama ağacını kontrol eder ve günceller. YTA kontrolcüsü, paketi tüm bağlantı noktalarına göndermek yerine bu kapsama ağacındaki bağlantı noktalarını kullanır.

4.1.3. Akış Yöneticisi (YTA-AY)

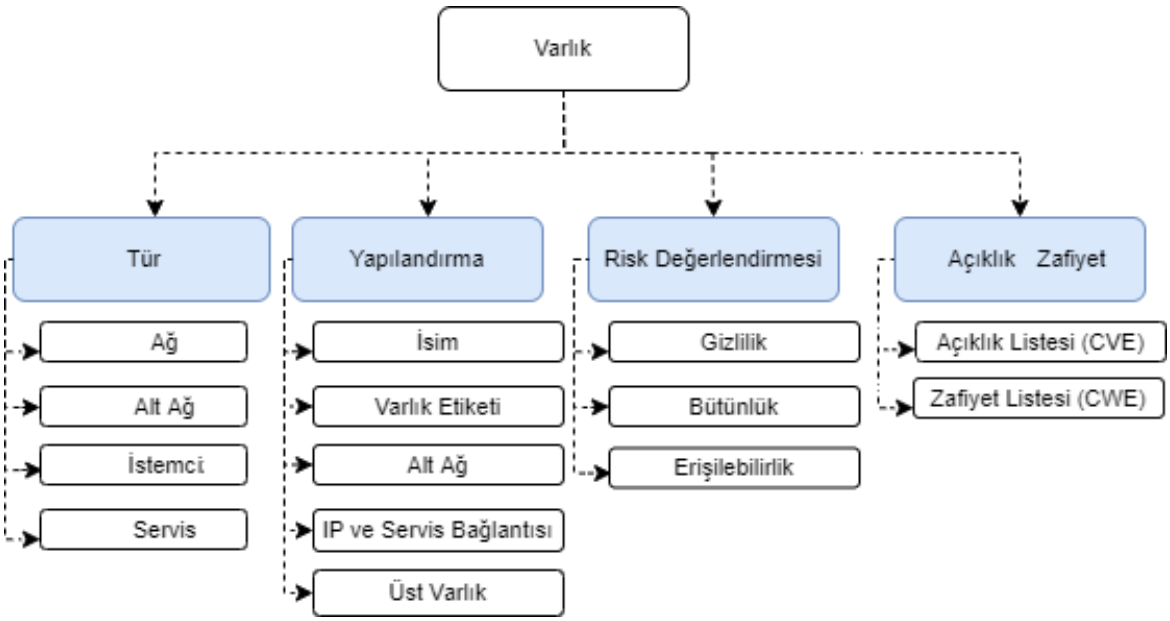
Akış Yöneticisi (YTA-AY), aktif akışları, akış değiştirme isteklerini (akış ekleme, silme ve değiştirme) ve akışla ilgili olayları yönetir. Sistemin her bileşeni istediği zaman akış tablolarını değiştirme talebinde bulunabilir. YTA-AY, sistem bileşenleri tarafından oluşturulan tüm akış mesajlarını iletir ve anahtarlar tarafından gönderilen asenkron bilgi mesajlarını alabilir. Aldığı bilgileri ilgili bileşenlere alarak aktarır. Anahtarlara akış değişikliği mesajları sadece YTA-AY tarafından gönderilebilir. YTA-AY, akış değiştirme mesajı gönderirken akış için benzersiz kimlik oluşturur ve bu bilgiyi akış kuralının çerez alanına ekler. Mesaj başarıyla gönderildiğinde, akışı aktif akış listesine ekler ve akışı izlemek için ilgili yöneticiye akış kimliğini döner. Y-S3 içinde yer alan savunma servisi bileşenleri ve YTA-SSY, akış kimliğine sahip ise herhangi bir anahtardaki akışı silme talebinde de bulunabilir. Talep edilen akışı siler ve durumu diğer bileşenlere bildirir. Akışlar ve tablolarla ilgili herhangi bir olay meydana gelirse, akış değişikliği talebi sahiplerine durumu bildirir. YTA-AY ayrıca, bir akış silindikten sonra kullanılmayan ve akış ile birlikte oluşturulmuş olan OpenFlow gruplarını siler.

4.1.4. Anomali Tespit Yöneticisi (ATY)

Tüm ağlarda güvenlik çözümleri ve anomali algılama servisleri vardır. Anomali tespit servisleri, paketleri analiz eder ve uyarılar oluşturur. Derin paket incelemeleri ve ağ trafiği anormallikleri, STS ve derin paket inceleme sistemleri kullanılarak tespit edilebilir. Siber Savunma Yöneticisi (YTA-SSY), STS veya derin paket inceleme sistemi ile haberleştiğinde, ağ seviyesinde bazı tehditler azaltılabilir. Anomali Tespit Yöneticisi (ATY), ağdaki STS sunucusunun konumunu tespit eder, STS'ye paket kopyalarının gönderilmesini sağlar ve STS alarmlarını dinler. ATY, bir alarm oluştuğunda alarmın türünü ve ayrıntı bilgilerini YTA-SSY'ye gönderir.

4.1.5. Varlık Yöneticisi

Varlık yöneticisi; bilgisayarları, alt ağları ve servisleri içeren tüm kurumsal varlıkları yönetir. Her varlığın; tür, yapılandırma, risk değerlendirme değerleri (gizlilik, bütünlük ve erişilebilirlik), olası güvenlik açıkları ve zayıflıklar gibi ayrıntılı bilgileri (Şekil 4.2) vardır. YTA-SSY, saldırılara karşı hedef varlıkları seçer ve saldırıları önlemek için ayrıntılı parametrelerle varlıkları sorgular.



Şekil 4.2. Kurumsal varlık taksonomisi

4.1.6. Siber Tehdit İstihbaratı Yöneticisi (STİ-Y)

STİ-Y, STİ verilerini sorgular ve depolar. Yeni STİ verilerini üç kritere göre değerlendirir ve filtreler. Filtrelemede kullanılan kriterler aşağıda listelenmiştir:

- Makine tarafından okunabilir tehdit tanımı olması
- Ağ seviyesinde uygulanabilir olması
- Güvenilirlik değerinin olması

STİ-Y, veriler güncellendiğinde YTA-SSY'i bilgilendirir. STİ-Y, her bir STİ verisini işleyerek ağ göstergelerini çıkarır. Yalnızca üst seviye göstergeleri (özet, e-posta adresi, dosya adı vb.) olan veya ağ göstergesi olmayan STİ verilerini dikkate almaz. Çizelge 4.2'de örnek STİ verileri ve bunlarla ilişkili ağ göstergeleri verilmiştir.

Çizelge 4.2. Siber tehdit istihbaratı ile paylaşılan ağ seviyesi gösterge örnekleri

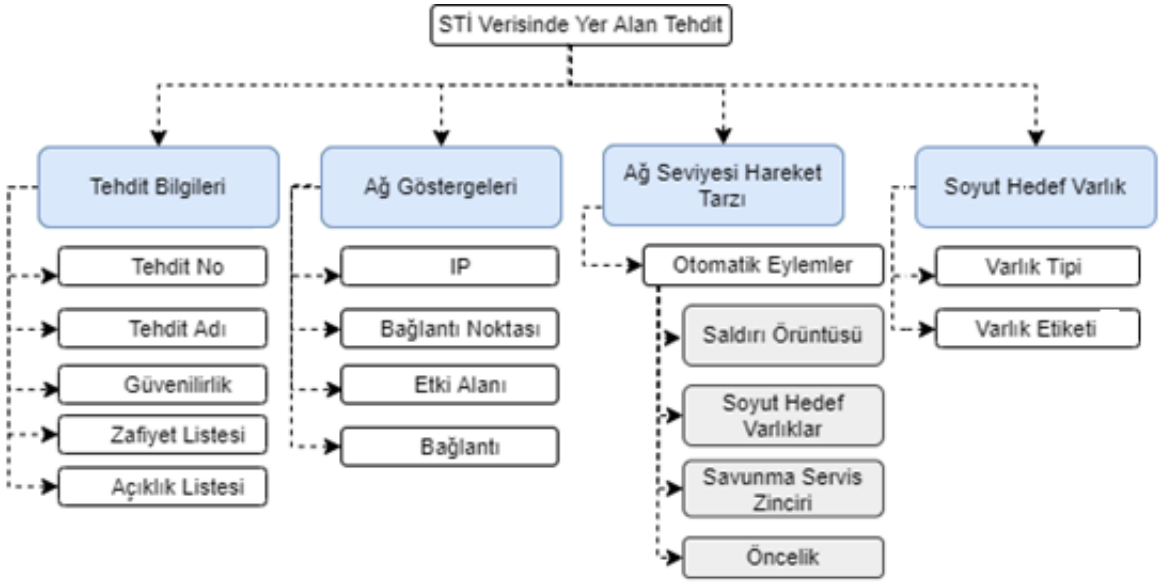
Tehdit	Açıklama	Ağ Göstergeleri
HAWKBALL Arka Kapısı	Yeni HAWKBALL arka kapısı, Orta Asya'da Hükümet Sektörünü hedef almıştır.	149.28.182.78:443, 149.28.182.78:80
Backdoor.Win32.Delf.DU - Zararlı Yazılım Etki Alanı Beslemesi v2	Backdoor.Win32.Delf.DU için Komut ve Kontrol etki alanlarıdır.	etki alanı: xexxds03.top
WannaCry - Zararlı Yazılım Alan Yayını v2	WannaCry için Komut ve Kontrol etki alanlarıdır.	www.ayylmaotjhsstasdfasdf asdfasdfasdfasdfasdf .com, fh724.pikatchuworld.club
OpenSSH truva atı haline getirilmiş araç seti	Bir araç seti; truva atı, rootkit, ters adli analiz, dinleme ve komuta ve kontrol yeteneklerine sahiptir.	gopremium.mooo.com, 5.189.136.43
APT28	Konaklama sektörünü hedefleyen bir saldırı dizisidir.	mvband.net, mvtband.net

Her STİ verisi, ilgili tehditleri önlemek veya saldırıların etkisini azaltmak için bir dizi eylem içerebilir. Ancak STİ verilerinin çoğu herhangi bir etki azaltma eylemi içermemekte veya yalnızca metin olarak (yapısal olmayan) yazılmaktadır. Aslında, STİ verilerinde otomatik

hareket tarzı belirlemek ve uygulamak gerçekten zordur. Bunun nedenleri genel olarak aşağıda özetlenmiştir:

- Güvenilirlik değerinin eksikliği: STİ verilerinin çoğunun güvenilirlik seviyesi genellikle bulunmamakta veya derecelendirilmemektedir. STİ verisinin güvenilirlik seviyesi, eşik değerin altında olması durumunda STİ verisi içerisinde yer alan saldırıya karşı hareket tarzı verilerine de güvenilmez.
- Topoloji farkları: Her organizasyon farklı bir ağ topolojisine sahip olduğundan, savunma mekanizmalarını kendi topolojisine göre uyarlaması veya savunma için topolojiyi yeniden yapılandırması gerekmektedir.
- Kurumsal dijital varlıkların farklılıkları: Dijital varlıklar; ağ, alt ağlar, bilgisayarlar, uygulamalar, servisler olabilir. Bir organizasyon diğer organizasyonlarla ortak ağ topolojisine sahip olsa bile, her kuruluş farklı servisleri, uygulamaları ve verileri yönetir. Bu varlıkların yapılandırmaları dikkate alınarak savunma mekanizmalarının bütüncül olarak yeniden tasarlanması gerekir.
- Mevcut savunma mimarileri ile uyumluluk: Bir savunma eylemi, organizasyonun siber güvenliğinin sürdürülebilirliğini desteklemelidir. Bir savunma mekanizması seçerken veya uygularken, siber savunma yöneticisi, bu savunma mekanizmasının başka güvenlik sorunlarına veya çakışmalara neden olmadığına emin olmalıdır.
- Alternatif savunma yöntemleri: Bir tehditle mücadele etmek için farklı savunma mekanizmaları belirlenmiş olabilir. Her savunma tüm ağlarda uygulanabilir bir çözüm olamayabilir. STİ verilerinde bir savunma mekanizması belirlendiğinde, bu mekanizmanın uygun olup olmadığı değerlendirilmelidir.

Şekil 4.3'te otomatik ağ savunması için STİ verilerinde olması gereken bilgileri içeren veri modeli önerisi sunulmaktadır. Mevcut STİ veri formatları, Şekil 4.3'te belirtilen veri yapısında tanımlanan tüm verileri içermemektedir. STIX belirtim dokümanı [11], en genel STİ veri paylaşım veri modelini tanımlamaktadır. Bu veri modelinde ağ seviyesinde otomatik savunma için hareket tarzları tanımlanmamaktadır. Ayrıca, STIX 2.1 öncesinde STİ verileri için hiç otomatik savunma tanımlanamamaktaydı. İlgili sürüm sonrasında ise sadece sistem seviyesinde otomatik işlemlerin tanımlanabilmesi için veri modeli oluşturulmuştur.



Şekil 4.3. Otomatik ağ savunması için STİ bilgisinde yer alması önerilen veri yapısı

Oluşturulan veri modelinde belirlenen veri kategorileri ve veri alanları aşağıda açıklanmıştır:

- Tehdit bilgileri: Otomatik ağ savunması için tehdidi tanımlanan bilgileri içeren veri alanları bu kategoride yer almaktadır. Bu kategoride tehdit no, tehdit adı, güvenilirlik, hedef alınan / kullanılan zafiyet ve açıklık listeleri tanımlanmaktadır. Bu veri alanları mevcut tüm STİ veri standartlarında farklı formatlarda yer almaktadır. Tehdit no veri alanı tehdidin tekil olarak oluşturulabilmesi ve mevcut tehditler arasından sorgulama yapılabilmesi için kullanılmaktadır. Tehdit adı bilgisi, tehdit için kullanılan alternatif isimleri tanımlamaktadır. Güvenilirlik bilgisi, STİ ile paylaşılan bilginin doğruluk düzeyini ifade etmek için kullanılmaktadır. Bu alanın yüksek değere sahip olması otomatik savunma yapılabilmesi için kriter olarak kullanılmaktadır. Tehdidin kullandığı veya hedef aldığı zafiyet ve açıklıklar listeler halinde belirtilmektedir. Bu listedeki zafiyet ve açıklıklar, kurumsal varlık listeleri taranarak tehdidin potansiyel hedeflerinin belirlenmesi sürecinde kullanılmaktadır.
- Ağ göstergeleri: Tehdidin ağ seviyesinde tespit edilebilmesi için kullanılabilecek IP, bağlantı noktası, etki alanı ve bağlantı verileri ağ göstergeleri olarak saklanmaktadır. Tüm STİ verileri bu bilgileri farklı formatlarda saklamaktadır. Organizasyonlar STİ verilerini çoğunlukla bu bilgileri alabilmek için kullanmaktadır.
- Ağ seviyesi hareket tarzı: Çoğu STI veri formatında hareket tarzları düz metin olarak yer almaktadır. STIX v2.1 ile sistem seviyesinde otomatik hareket tarzları tanımlanabiliyor olsa da otomatik savunma içeren hareket tarzları tanımlanamamaktadır. Bu kategorideki

veri alanları bu tez kapsamında belirlenmiştir. Ağ seviyesi hareket tarzlarının STİ verilerinde düz metin olarak ifade edilmesi, tehdit önleme işlemlerinin otomatik gerçekleştirilmesini sağlamadığı için otomatik eylem listeleri tanımlanması gerekmektedir. Bu eylem listelerinin her birinde tehdidin saldırı örüntüsü veya hedef aldığı varlıklar için alınabilecek savunma servis zinciri öncelik değeri tanımlanarak belirtilmektedir. Saldırı örüntüsü, hedefin saldırılarda kullandığı teknik, araç ve işlemleri ifade etmektedir. Bu alan kullanılarak tehdidin belirtilen farklı saldırı örüntüleri için farklı savunma yöntemleri kullanılabilmektedir. Ayrıca tehdidin önlenmesi için belirli farklı varlıklarda farklı ağ savunması sağlanabilmektedir. Savunmada kullanılacak ağ savunma servisleri ve sıralaması, savunma servis zinciri şeklinde tanımlanmaktadır. Savunma servis zincirinde tez kapsamında belirlenen temel ağ savunma servislerinin kullanılması sağlanmıştır.

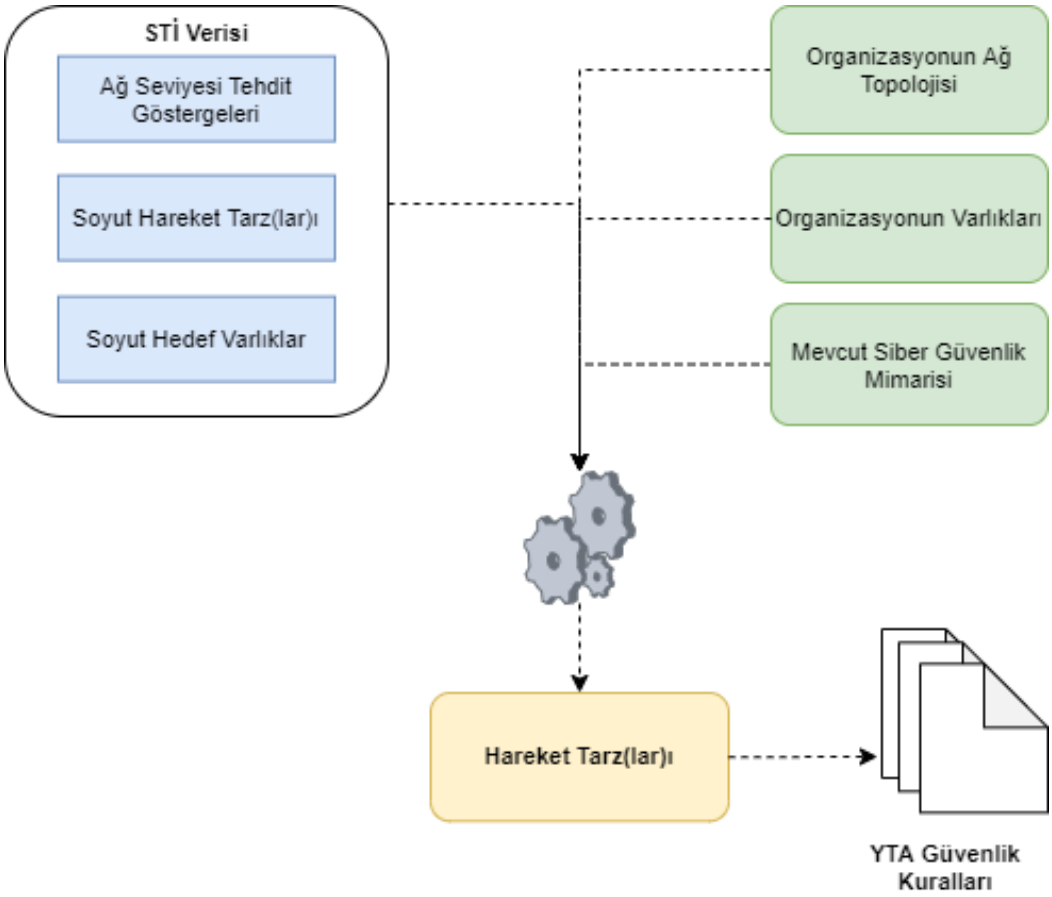
- Soyut hedef varlık: Tehdit bilinen bir zafiyeti veya açıklığı kullanmadan doğrudan belirli bir varlık grubunu hedef aldığında (MySQL veritabanı, e-Devlet uygulaması) varlığın belirtilebilmesi için kullanılabilecek alanlardan oluşmaktadır. Bu kategorideki bilgiler, otomatik ağ savunması için Şekil 4.2'de belirtilen kurumsal varlık veri modeli ile eşleştirilmesinde kullanılmaktadır. Zafiyet veya açıklık listesi ile gösterilmeyen hedef varlıkların varlık tipleri ve varlıklar için kullanılabilecek muhtemel etiketler bu kategoride tanımlanmaktadır. Varlık tipi olarak; ağ, alt ağ, istemci veya bir istemcide çalışan bir servis tanımlanabilmektedir.

Şekil 4.3'te gösterilen veri modelinin kullanımına örnek olarak tehditlere ait soyut hedef varlıkları ile savunma servisi önerileri Çizelge 4.3'te gösterilmiştir.

STİ verilerinde soyut savunma yöntemlerinin ve hedef varlıkların yer alması durumunda, organizasyonlar Şekil 4.4'te gösterildiği gibi, ağ üzerinde savunma kuralları oluşturabilir ve oluşturulan kurallar ile mevcut güvenlik mekanizmalarını yapılandırabilirler.

4.1.7. Siber Savunma Yöneticisi (YTA-SSY)

Siber Savunma Yöneticisi (YTA-SSY); ağ topolojisi, kurumsal varlıklar, siber güvenlik mimarisi ve STİ verileri kullanılarak oluşturulan eylemlerin akışına dayalı YTA güvenlik kuralları oluşturmak için savunma servislerini yönetir (Bkz. Şekil 4.4).



Şekil 4.4. STİ verisi kullanılarak ağ savunma kurallarının otomatik oluşturulması

Çizelge 4.3. Otomatik ağ savunması için önerilen veri modelinde yer alan bilgileri içeren STİ verisi örnekleri

Tehdit Tanımı	Ağ Göstergeleri	Soyut Hedef Varlıklar	Savunma Servis Zinciri Önerileri
HAWKBALL Arka Kapısı	149.28.182.78:443, 149.28.182.78:80	Varlık Tipi: E-devlet uygulaması, Ofis uygulaması Etiket: Microsoft Office CVE: CVE-2018-0802, CVE-2017-11882	1- kara liste 2- derin paket inceleme, bal küpü 3- saldırı tespit sistemi, derin paket inceleme
Backdoor.Win32.Delf.DU - Zararlı Yazılım Etki Alanı Beslemesi v2	etki alanı: xexxds03.top	Etiket: İstemci, Windows 7, Windows 10	1- kara liste
WannaCry - Zararlı Yazılım Alan Yayını v2	www.ayylmaotjhsstasdfasdfasdfasdfasdfasdfasdf .com, fh724.pikatchuworld.club	Varlık Tipi: Uygulama Sunucusu, İstemci Etiket: SMB Sunucu CVE: CVE-2017-0143, CVE-2017-0144, CVE-2017-0145, CVE-2017-0146, CVE-2017-0148	1- kara liste, karantina 2- kara liste, karantina, paket düşür (port 445) 3- derin paket inceleme, karantina, bal küpü

Şekil 4.5'te YTA-SSY tarafından hareket tarzı belirleme sürecinin ana adımları gösterilmiştir. YTA-SSY, hareket tarzı belirlenirken ilk olarak STİ verilerinin güvenilirlik değerini kontrol eder. Sürecin devam etmesi için STİ güvenilirlik değeri eşikten büyük olması gerekmektedir (satır 2 - 3). STİ verisindeki soyut hedef varlıkları, ağ göstergeleri ve tehdit bilgisi verileri kullanılarak varlık kütüphanesindeki tüm ilgili varlıklar bulunur. STİ verisinde yer alan varlık türü, varlık etiketleri, açıklıkları (CVE) ve zafiyetleri (Common Weakness Enumeration (Genel Zayıflık Sayımı, CWE) ile eşleşen hedef varlıklar seçilir (satır 4). Her varlık için hizmet zincirleri oluşturulmaya çalışılır (satır 6 - 9). Varlıklar; ağ, alt ağ, ana bilgisayar veya servis olabilir. Ağ için servis zinciri oluşturulmuşsa (kurumsal seviye) diğer servis zincirleri göz ardı edilebilir.

```
1  Fonksiyon hareket_tarzi_belirle(sti_verisi, topoloji, varlik_kutuphanesi,
       guvenlik_servisleri, guvenilirlik_min):
2      if sti_verisi.guvenilirlik < guvenilirlik_min then
3          return
4      varliklar_tumu ← varliklari_bul(varlik_kutuphanesi, sti_verisi)
5      guvenlik_zincirleri ← ∅
6      for varlik ∈ varliklar_tumu do
7          guvenlik_zinciri ← zincir_olustur(varlik, sti_verisi, topoloji, guvenlik_servisleri)
8          if guvenlik_zinciri then
9              guvenlik_zincirleri[varlik] ← guvenlik_zinciri
10     return guvenlik_zincirleri
11 Fonksiyon zincir_olustur(varlik, sti_verisi, topoloji, guvenlik_servisleri):
12     hareket_tarzi_tumu ← sti_verisi.hareekt_tarzi.otomatik_eylemler
13     hareket_tarzi ← uygun_hareket_tarzi(hareket_tarzi_tumu, varlik.etiketler)
14     if hareket_tarzi then
15         savunma_zinciri ← hareket_tarzi.savunma_zinciri
16         servisler ← servisleri_bul(guvenlik_servisleri, hareket_tarzi.savunma_zinciri)
17         guvenlik_zinciri ← zincir_olustur(varlik, topoloji, servisler, savunma_zinciri)
18         if cakisma_var_mi(guvenlik_zinciri, servisler, guvenlik_servisleri) = 0 then
19             return guvenlik_zinciri
20     return
```

Şekil 4.5. STİ verileri kullanılarak otomatik savunma hareket tarzı belirleme adımları

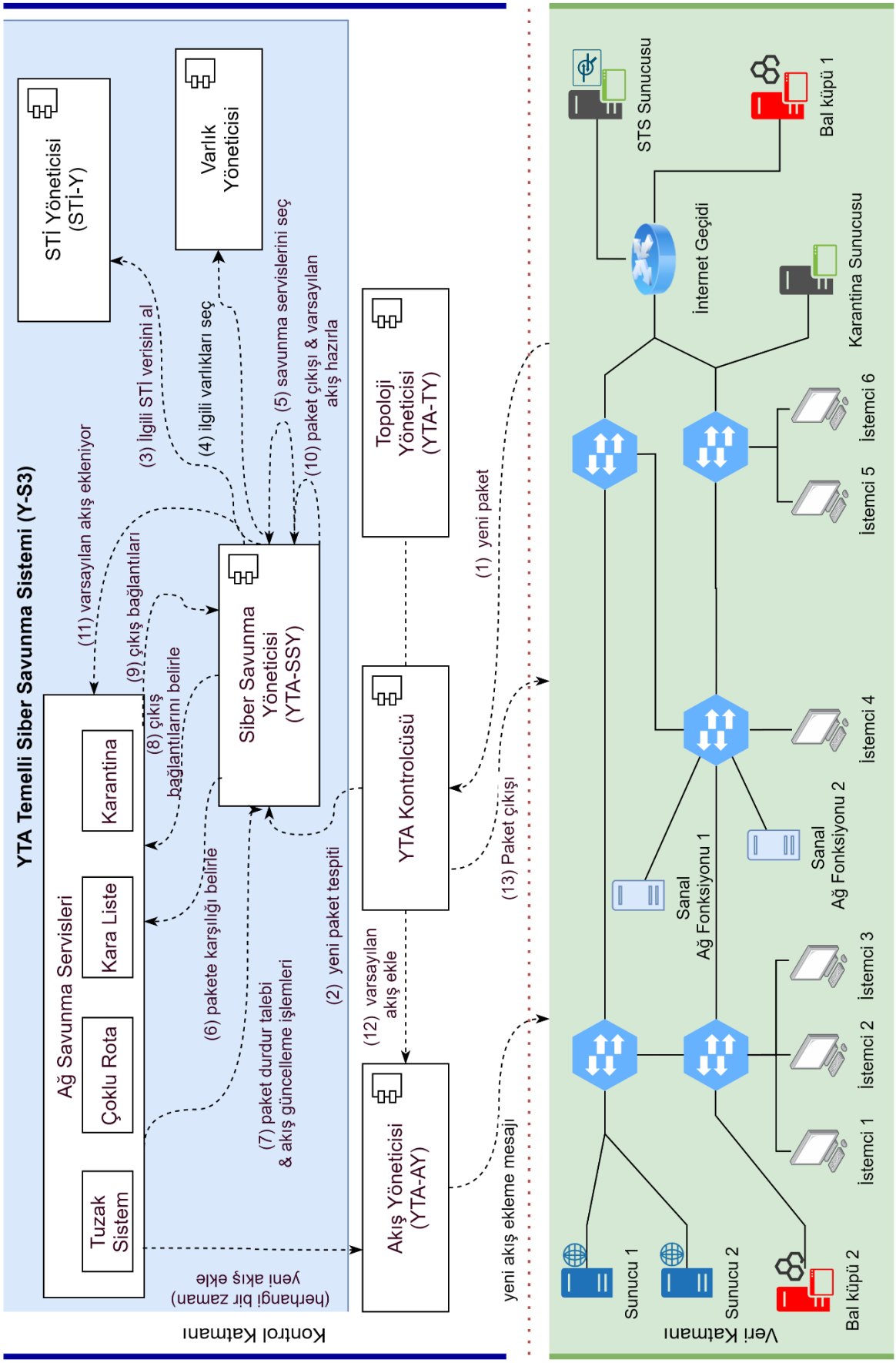
Servis zinciri oluştururken otomatikleştirilmiş eylemler bulunur, kriterlere uyan ve daha yüksek önceliğe sahip olan seçilir (satır 12 - 13). YTA-SSY, STİ verilerindeki savunma zinciri tanımını uyan ağ savunma servislerini bulur. Ağ savunma servisleri, çalışmamızda önerilen savunma servisleri veya ağda dağıtılan ağ sanallaştırma fonksiyonları olabilir. Servis zinciri, topoloji ve varlıklar kullanılarak belirlenir (satır 15 - 17). Akış kurallarının tutarlılığı kontrol edilir (satır 18 - 19). NSH kullanılarak, her servis zincirinin benzersiz bir

kimliği oluşturulur. Böylece servis zinciri için oluşturulacak akış kurallarında bu bilgi kullanılarak akış çatışmalarına neden olunmaması sağlanır.

Genel olarak YTA kontrolcüsü veya bileşenler, gerçekleşen olayları YTA-SSY'ye gönderir. YTA-SSY, yeni paket algılandığında veya diğer bileşenler tarafından STS uyarıları veya yeni STİ verileri gibi yeni olaylar tetiklendiğinde paketi işlemeye başlar. Örnek bir iş akışı olarak Şekil 4.6'da YTA kontrolcüsü tarafından yeni bir paket algılandığında YTA-SSY tarafından gerçekleştirilen işlem adımları özetlenmiştir.

Yeni paket algılandığında (1), YTA kontrolcüsü siber savunmayı başlatmak için YTA-SSY'yi (2) çağırır. YTA-SSY, STİ verilerini (3) ve ilgili varlıkları (4) seçer. Topoloji, STİ verileri, varlıklar ve mevcut güvenlik kurallarını dikkate alarak paketleri değerlendirir. Siber savunma çözümlerini kullanarak genel eylemler üretir (5). Pakete karşı yanıtlara karar vermek için seçilmiş servislere paketi gönderir (6). Siber savunma servisleri, akış değiştirme eylemlerini ve süreç sonlandırma durumunu hazırlar ve YTA-SSY'ye gönderir (7). Paket işlemi sonlandırılmazsa, YTA-SSY, servislere paket çıkış portunu sorar (8). Servisler, yeniden yönlendirme, tuzağa gönderme gibi paketin varsayılan davranışını değiştirirlerse seçilen paket için çıkış bağlantı noktasını dönerler (9). YTA-SSY, paket ve varsayılan akış kuralı için paket çıkışı mesajı hazırlar (10). Birden fazla servis aynı akış eşleşmesi için eylemler üretirse, tüm eylemleri uygulamak için OpenFlow grubu oluşturur ve ilgili gruba yönlendirmek için bir akış kuralı oluşturulur. Servisleri eklenecek olan varsayılan akış hakkında bilgilendirir (11) ve YTA kontrolcüsü tarafından yönetilen YTA-AY kullanarak akış değiştirme mesajı gönderir (12). Son olarak, YTA kontrolcüsü paket için hazırladığı paket çıkışı mesajını anahtarlara gönderir. Bir siber savunma servisi başladıktan sonra YTA-AY üzerinden her zaman akış değişiklik istekleri gönderebilir. YTA-AY istekler için anahtarlara mesaj gönderir ve istekte bulunanlara akış kimliğini geri döner.

STS uyarılarını alan ATY, STİ verileri güncellendiğinde STİ-Y ve farklı ağlardan iş birliği mesajlarını alan SİY tarafından ağ savunması için YTA-SSY tetiklenebilmektedir.



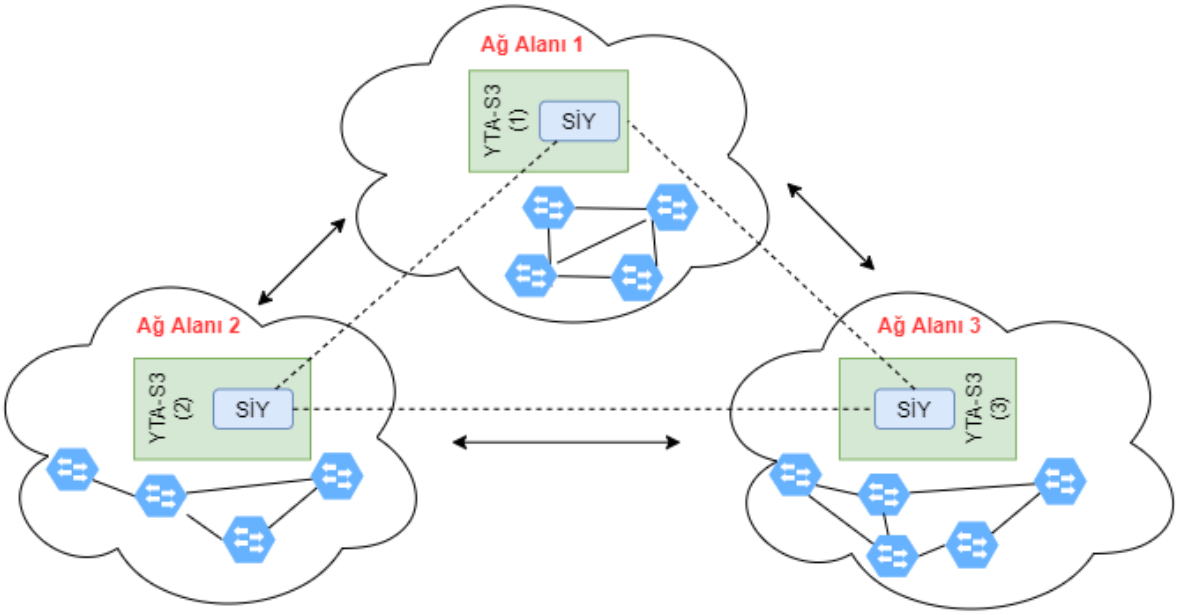
Şekil 4.6. Yeni bir paket alındığında ağ seviyesi savunma servisi oluşturma adımları

4.1.8. Savunma İş Birliği Yöneticisi (SİY)

Savunma iş birliği yöneticisi, aynı sistemin farklı ağlarda kurulan altyapılarla iletişim kurması için ajan görevini üstlenmektedir. Savunma İş Birliği Yöneticisi, kuzey yönlü ara yüz sunarak diğer SİY ve uygulamaların iletişim kurabilmesini sağlamaktadır.

Diğer SİY tarafından paylaşılan bilgiler ve savunma talepleri için SİY, YTA-SSY ile etkileşimde bulunur. İki veya daha fazla SİY Şekil 4.7'de gösterildiği gibi birbirleri ile aşağıdaki durumlarda iletişim kurmaktadır:

- Siber tehdit istihbarat bilgisini paylaşmak veya almak
- Saldırı anında saldırı bilgilerini paylaşmak veya almak
- Saldırıları önlemek için ağ savunması talebinde (kendisine gelen trafiğin düşürülmesi, belirli bir kaynak IP adresinin karantinaya alınması vb.) bulunmak



Şekil 4.7. Farklı ağ alanları arası ağ savunma iş birliği

4.1.9. Ağ seviyesi savunma servisleri

YTA-SSY ile iletişime girerek ağ seviyesinde gerçekleştirilecek servisler modüler şekilde geliştirilerek sisteme eklenebilmektedir. Bu kapsamda Bölüm 3'te yer alan Çizelge 3.2'deki ağ seviyesi savunma servislerinin geliştirilebileceği gibi, belirli türdeki saldırılara özel savunma servislerinin de geliştirilebilmesi mümkündür. Tez kapsamında Y-S3'ün kavram

ispatının yapılması ve performansının test edilmesi için bazı ağ seviyesi savunma servisleri tasarlanmış ve geliştirilerek sisteme entegre edilmiştir. Bu servisler Bölüm 4.2'de detaylı olarak açıklanmıştır.

4.2. Y-S3'e Entegre Edilen Ağ Savunma Servisleri

Bu tez çalışmasında, birden fazla saldırgan modeline karşı etkili, ağ açısından düşük maliyetli ve ele alınan saldırı modellerinin hepsine karşı çözüm önerebilecek kapsayıcılıkta bir sistem oluşturmak hedeflenmiştir. Bu hedef doğrultusunda, temel siber savunma servisleri arasından kara liste, karantina, tuzağa gönderme ve çoklu rota servisleri seçilmiş ve seçilen bu servisler geliştirilerek Y-S3'e entegre edilmiştir. İhtiyaca göre geliştirilecek yeni siber savunma servisleri Y-S3'e entegre edilebilmektedir. Çizelge 4.4'te, tez çalışması kapsamında belirlenen tehdit modellerine karşı önlem almak için kullanılabilecek örnek ağ savunma servisleri gösterilmektedir.

Çizelge 4.4. Saldırgan modelleri ve kullanılabilecek siber savunma servisleri

Ağ Savunma Servisi / Saldırgan Modeli	Kara Liste Servisi	Karantina Servisi	Tuzak Sisteme Gönderme Servisi	Çoklu Rota Servisi
DDoS saldırganı	*	*	*	*
İndirme saldırganı	*	*	*	
Zararlı yazılım kontrolcüsü	*	*	*	
Oltama saldırganı	*	*	*	
Dinleme saldırganı		*		*
IP sahtekârlığı saldırganı		*	*	

Saldırıları önlemek için yalnızca bir savunma servisi kullanılabileceği gibi, gelişmiş çözümler için savunma servisleri kombinasyonu (sıralı veya paralel) oluşturulabilmektedir. Bir saldırgan, saldırının her aşaması için farklı rollere sahip olabilir. Örneğin, bir saldırgan ilk aşamada bir oltalama saldırganı olarak hareket ettikten sonra ileriki aşamalarda zararlı yazılım kontrolcüsü rolüne sahip olabilir.

Savunma senaryosu hareket tarzı örnekleri Şekil 4.8'de gösterilmiştir. Bu hareket tarzlarını kullanabilmek için örnek bir STİ verisi olması gerekmektedir. Bu STİ verisi içinde DDoS komuta ve kontrol sunucusu adresleri, soyut hedef varlıkları ve soyut hareket tarzları yer almaktadır.

Ağ seviyesi hareket tarzı olarak K&K sunucularına gelen ve sunuculardan gelen tüm trafiğin kesilmesi, bu sunucularla iletişim kurmaya çalışan istemcinin karantinaya alınması için Şekil 4.8a'da belirtilen savunma senaryosu oluşturulmuştur. Belirtilen şekilde gösterilen hareket tarzı, zararlı trafiğin tuzak sistemlerde analiz edilmesine gerek olmadığı durumlarda; ağda kara liste ve karantina hizmetlerini kullanarak indirme, zararlı yazılım ya da oltalama saldırısı için kullanılabilir.

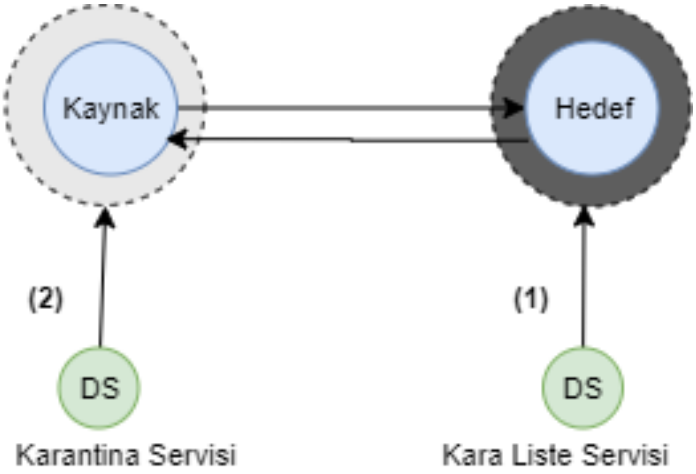
Zararlı veya şüpheli trafik analiz edilmek istendiğinde, hareket tarzı Şekil 4.8b'de gösterilen şekilde değiştirilebilir. Bir istemci herhangi bir zararlı IP adresine (fidye yazılımı K&K sunucusu, zararlı yazılım K&K sunucusu vb.) erişmeye çalıştığında, trafik bal küpüne gönderilmelidir. Bal küpü K&K sunucusuna / sunucusundan gelen tüm trafiği yönetir. Zararlı yazılım bulaşan istemci, karantina servisi ile ağdan izole edilir. Tuzağa gönderme servisi, zararlı trafiği izole etmek ve yönetmek için servis fonksiyon zincirleri oluşturulur.

4.2.1. Kara liste servisi

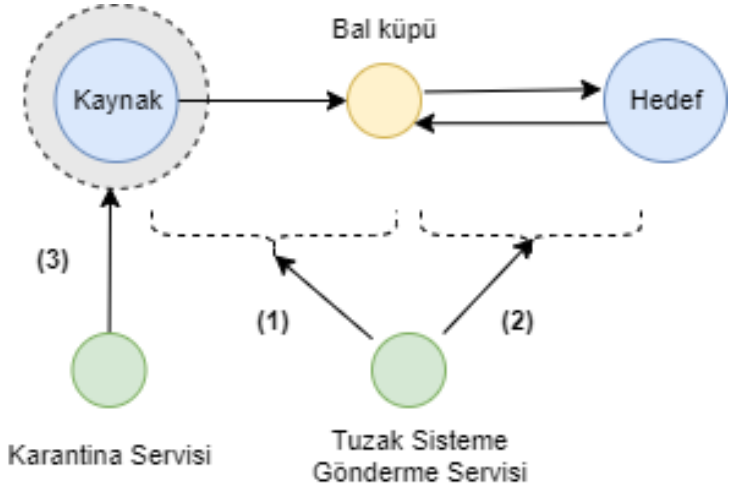
Kara liste yönetimi, temel siber savunma servislerinden biridir. Geleneksel ağlarda kara listeler çoğunlukla güvenlik duvarları üzerinde uygulanır. Ancak bu çözümün çalışabilmesi için tüm trafiğin güvenlik duvarlarından geçmesi gerekir. YTA mimarisinde ise kara liste yönetimi ağ kontrolcüsü tarafından yapılabilir. Tez kapsamında tanımlanan kara liste servisi, ağdaki herhangi bir anahtarda paketleri engelleyebilmektedir. Bu özellik sayesinde, kaynak anahtarda, yol üzerindeki orta anahtarlarda veya tüm anahtarlarda engelleme uygulanabilmektedir.

Tüm kara liste öğelerinin sona erme süresi vardır. Bu bilgi sürekli ve dinamik olarak güncellenmelidir. Tüm kara liste öğelerinin sayısı bir YTA anahtarının akış tablosunun maksimum kapasitesinden daha büyükse, kara liste öğeleri anahtarlar arasında iş birliği içinde yönetilebilir. YTA ile kara liste öğelerinin yönetilebileceği diğer bir yöntem de sadece

en son görülen veya güvenilirliği en yüksek olan kara liste öğelerinin ağ performansını artırmak için anahtarlara yüklenmesidir.



a)



b)

Şekil 4.8. a) Karantina ve kara liste servisleri ile ağ savunması b) Karantina ve tuzak sisteme gönderme servisleri ile ağ savunması

Kara liste servisi, algılanan yeni paket veya STİ verisi alındığında çağrılarak aktive edilebilir ve Şekil 4.9'da gösterilen adımlar işletilir. Kara liste servisi etkinleştirildiğinde, servis paket içindeki kaynak ve hedef IP adreslerini kara liste ile karşılaştırır. Adreslerden biri kara listedeyse düşürme eylemi için akış değiştirme mesajı oluşturur ve mesaj göndermek için talepte bulunur (satır 2 - 11). Son olarak kara liste akışı oluşturulursa paketin işlenmesini sonlandırmayı ve paket çıkışı mesajı gönderilmemesini talep eder (satır 12 - 15). IP adresleri için kurallar anahtarlara eklendiğinde IP adresleri uygulanan kara listeye kaydedilir. Eklenen

kural, belirlenen süre sonunda silindiğinde uygulanan kara listeden ilgili IP adresleri kaldırılır.

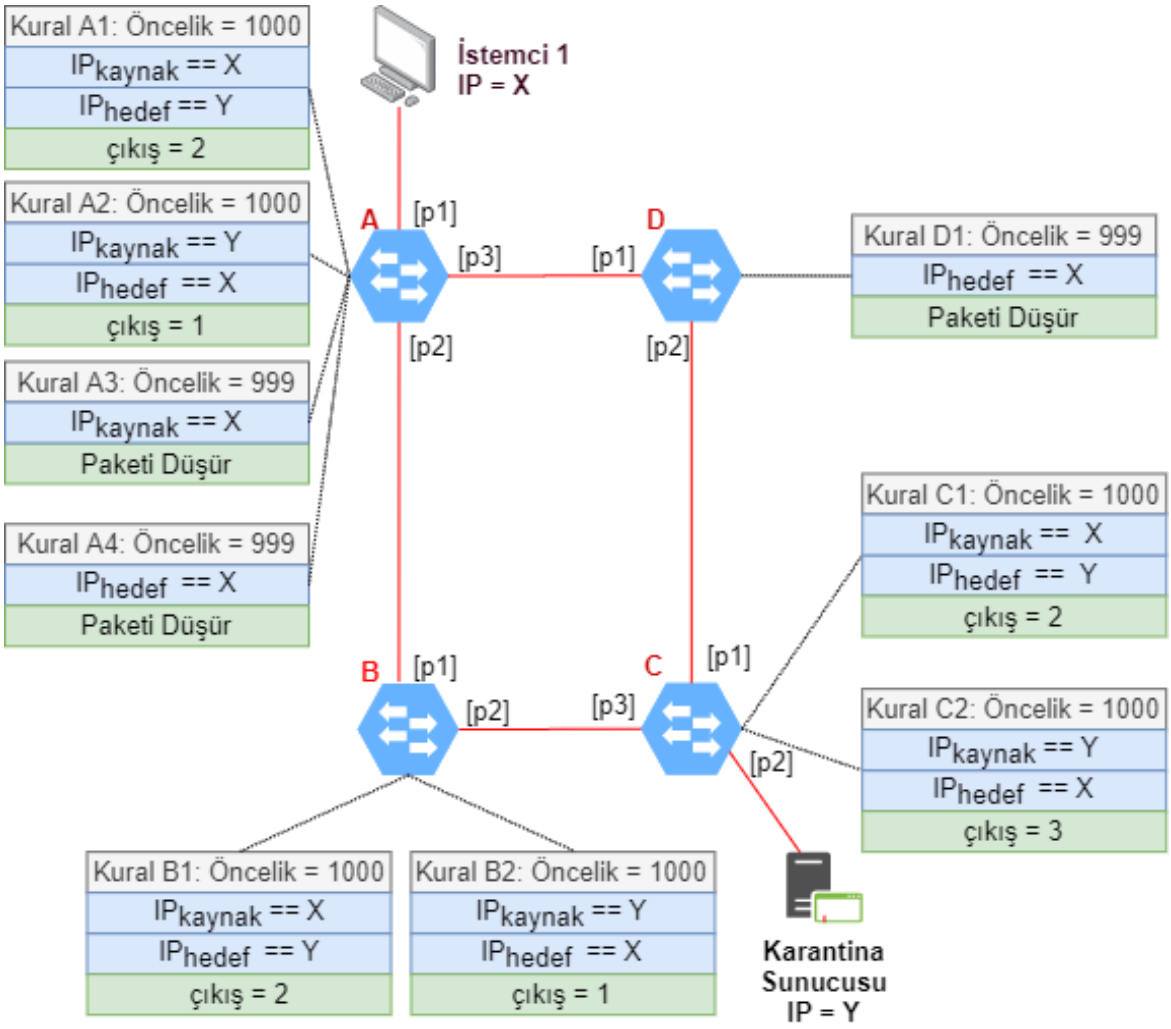
```
Girdi : karaliste, karaliste_aktif, src_dp,p,ip, dst_dp,p,ip, öncelik, bekleme_süresi
eşlemeler ← []
if kaynak_ip ∈ karaliste and kaynak_ip ∉ karaliste_aktif then
    karaliste_aktif.ekle(kaynak_dp,p,ip)
    eşlemeler.ekle({ipv4_kaynak = kaynak_ip})
if hedef_ip ∈ karaliste and hedef_ip ∉ karaliste_aktif then
    karaliste_aktif.ekle(hedef_dp,p,ip)
    eşlemeler.ekle({ipv4_hedef = hedef_ip})
for esleme ∈ eslemeler do
    eylemler ← düşürme_eylemi
    akış_mesajı ← akışı_oluştur(eşleme, eylemler, öncelik, bekleme_süresi)
    gönderme_talep_et(akış_mesajı)
if kaynak_ip ∈ karaliste or hedef_ip ∈ karaliste then
    return BİTİR
else
    return DEVAM
```

Şekil 4.9. Kara liste servisi tarafından anahtarlara kara liste eklenmesi

4.2.2. Karantina servisi

Ağdaki kötü niyetli etkinlikleri önlemek için istemcilerin karantinaya alınarak diğer ana bilgisayarlardan izole edilmesi yaygın kullanılan bir savunma yaklaşımıdır. Karantina; akış kesintisi, erişim kontrol listeleri, sanal yerel alan ağı (virtual local area network, VLAN) anahtarlama, anahtar bağlantı noktası kapatma ve adres çözümleme gibi farklı teknikler kullanarak uygulanabilir. Tez çalışmasında akış kesinti yöntemi kullanılmıştır. Karantinaya alınan istemcilerdeki zararlı yazılımları temizlemek veya saldırının etkilerini azaltmak için ağa bir sunucu yüklenir.

Karantina servisi, karantina işlemleri için veri katmanında gerekli yapılandırmaları gerçekleştirir. Şekil 4.10'da İstemci 1'in karantinaya alınması için oluşturulan ve anahtarların kural tablolarına eklenen girdiler örnek olarak gösterilmektedir. Şekilde kuralların mavi ile gösterilen satırları akış eşleşme şartlarını, yeşil ile gösterilen satırlar da anahtarda gerçekleştirilecek işlemleri ifade etmektedir.



Şekil 4.10. Karantina servisi tarafından gerçekleştirilen örnek ağ yapılandırması

Bir istemci karantinaya alınmak istendiğinde istemcinin bulunduğu bağlantı noktasından gelen paketlerin incelenmesi ve sadece karantina sunucusuna giden paketlere izin verilmesi gerekmektedir. YTA anahtarları tarafından bir paket için birden fazla kural ile eşleşme sağlandığında önceliği yüksek kurallar işletilmektedir. Bu sebeple şekildeki örnekte karantina sunucusuna giden paketlerin düşürülmemesi ve önceliklendirilmesi için anahtarlarda oluşturulan kuralların önceliği 1000, diğer paketleri işlemek için oluşturulan kuralların önceliği de 999 olarak belirlenmiştir.

Şekil 4.10'da gösterildiği gibi, istemcinin bağlı olduğu anahtardan gelerek karantina sunucusuna giden trafik için yönlendirme yapılmaktadır (Kural A1). Hedef IP adresi karantina sunucusu olmadığı durumda da paketler düşürülmektedir (Kural A3). Benzer şekilde hedef IP adresi olarak İstemci IP adresine sahip paketler incelenmekte ve karantina sunucusundan gelenlere izin verilmekte (Kural A2), diğer paketler düşürülmektedir (Kural

A4). İstemciden karantina sunucusuna giden ve karantina sunucusundan gelen trafiğin yönlendirilebilmesi için karantina servisi tarafından rota hesaplanarak rotada yer alan anahtarlara da yüksek önceliğe sahip yönlendirme kuralları eklenmektedir. Verilen örnekte en kısa rota A - B - C anahtarları olarak hesaplanmıştır. Bu rota üzerindeki B ve C anahtarlarına da paketlerin karantina sunucusuna yönlendirilmesi için sırası ile Kural B1 ve Kural C1 tanımlanmıştır. Geri dönen trafiğin belirlenen rota üzerinden iletilebilmesi için yine C - B - A anahtarlarında sırası ile Kural C2, Kural B2 ve Kural A2 oluşturulmaktadır. Ayrıca istemci ile karantina sunucusu arasında yer almayan anahtarlarda hedef IP adresi olarak istemci IP adresi tespit edildiğinde karantina servisi ilgili anahtarlara paketin düşürülmesi için Kural D1'de belirtilen şekilde kurallar belirlemektedir. Bu işlemler karantina servisi tarafından kontrol katmanında hesaplanarak tüm kuralların anlık olarak oluşturulması için gerekli OpenFlow mesajları YTA Akış Yöneticisine gönderilmektedir.

Karantina servisi, Şekil 4.11'de belirtilen adımları takip ederek karantina işlemi için gerekli ilgili kuralları oluşturmaktadır.

Girdi : $karantina_{aktif}$, $kaynak_{dp,p,ip}$, $karantina\_sun_{dp,p,ip}$, $öncelik$, $bekleme\_süresi$

1 $eşlemeler \leftarrow []$
2 **if** $kaynak_{ip} \notin karantina_{aktif}$ **then**
3 $\quad karantina_{aktif}.ekle(kaynak_{dp,p,ip})$
4 $\quad eşleme[0] \leftarrow \{ipv4\_kaynak = kaynak_{ip}, girdi\_p = kaynak_p\}, eylemler[0] \leftarrow drop\_eylemi$
5 $\quad eşleme[1] \leftarrow \{ipv4\_hedef = kaynak_{ip}\}, eylemler[1] \leftarrow düşür\_eylemi$
6 $\quad rota \leftarrow en\_kısa\_yol(kaynak_{dp}, kaynak_p, karantina\_sun_{dp}, karantina\_sun_p)$
7 $\quad eşleme[2] \leftarrow \{ipv4\_kaynak = kaynak_{ip}, ipv4\_hedef = karantina\_sun_{ip}, girdi\_p = kaynak_p\}$
8 $\quad eylemler[2] = [output(rota[kaynak_{dp}][output])]$
9 $\quad eşleme[3] \leftarrow \{ipv4\_hedef = kaynak_{ip}, ipv4\_kaynak = karantina\_sun_{ip}\},$
$\qquad\qquad eylemler[3] = [çıkış(kaynak_p)]$
10 $\quad$ **for** $i = 0; i < 4$ **do**
11 $\quad\quad akış\_mesajı \leftarrow akışı\_oluştur(eşleme[i], eylemler[i], öncelik, bekleme\_süresi)$
12 $\quad\quad gönderme\_talep\_et(akış\_mesajı)$
13 $\quad\quad i = i + 1$
14 **if** $kaynak_{ip} = karantina\_sun_{ip}$ **or** $hedef_{ip} = karantina\_sun_{ip}$ **then**
15 $\quad$ **return** $DEVAM$
16 **return** $BİTİR$

Şekil 4.11. Karantina servisi tarafından istemcilere karantina uygulanması

Karantina servisi, kaynak IP adresini anahtara dört akış ekleyerek izole eder. Bunlardan ikisi karantinaya alınacak IP adreslerinden gelen ya da bu adrese giden paketleri düşürmek içindir (satır 4 - 5). Diğer iki akış, paketlerin karantina sunucusuyla iletişim kurmasına izin vermek

içindir (satır 6 - 9). Karantina servisi, gerekli mesajları hazırlar ve her mesaj için akış değişikliği talebinde bulunur (satır 10 - 13). Paketteki kaynak veya hedef IP adresi karantina sunucusu değilse paketin düşürülmesi ve işlemesinin sonlandırması için YTA-SSY'yi bilgilendirir.

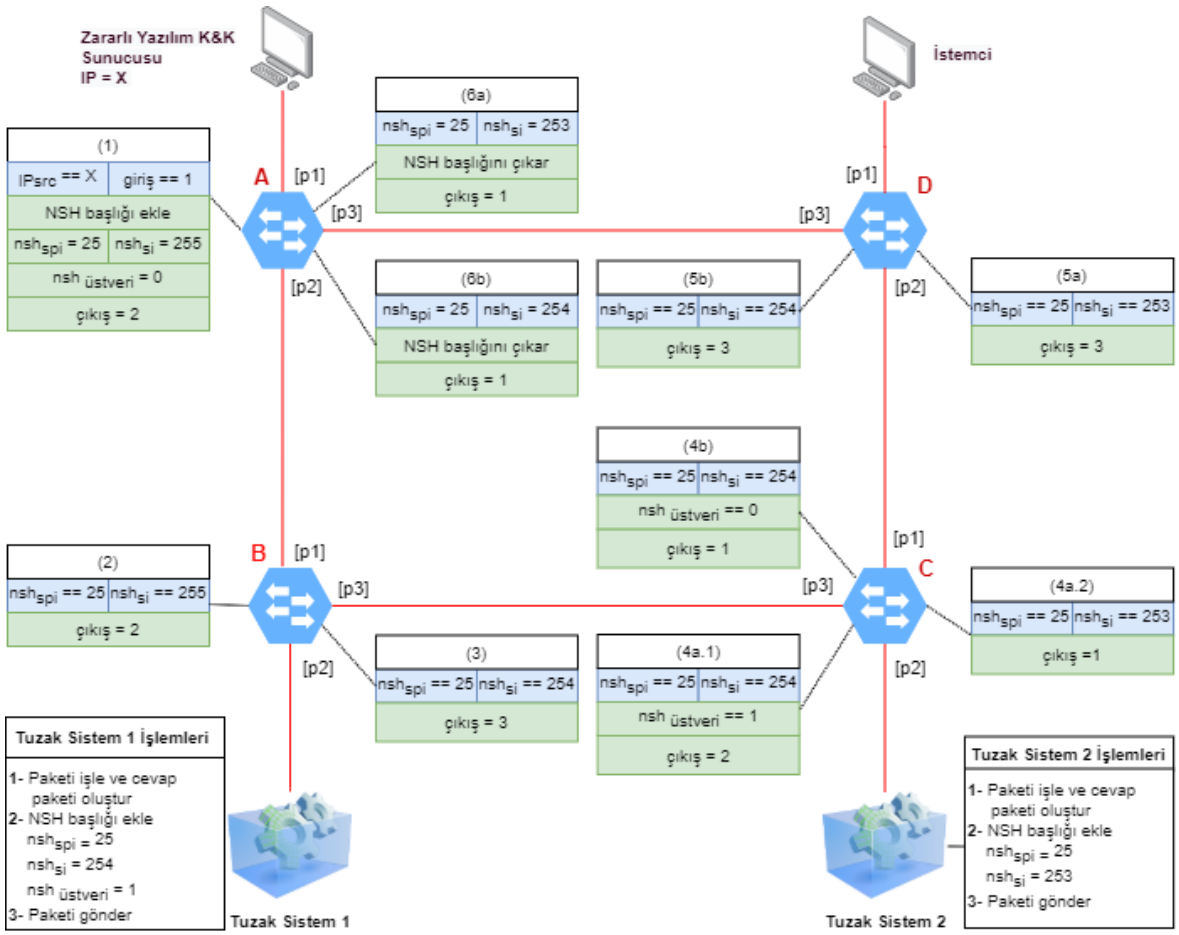
4.2.3. Tuzak sisteme gönderme servisi

Tez çalışmasında, tuzak terimi ağlara kurulan herhangi bir aldatma sistemi (bal küpü, bal küpü ağı, STS vb.) için kullanılmıştır. Bal küpü, ağlar için yaygın bir tuzak sistemidir. Bal küpleri sıfırıncı gün saldırılarını algılayabilir ve saldırganların ayrıntılı davranışlarını elde etmek için destek sağlayabilir. Klasik ağlara yeni bal küpleri yüklendiğinde, ağ yapılandırmaların ve güvenlik politikalarının revize edilmesi gerekmektedir. YTA mimarisinde, sistemleri dinamik olarak tuzağa düşürmek için trafik izole edebilmektedir. Tuzak sisteme gönderme servisi, zararlı trafiği normal trafikten ayırır ve tuzak sistemlere yönlendirir.

Şekil 4.12'de zararlı yazılım K&K sunucusundan gelen bir trafik tespit edildiğinde trafiğin tuzak sistemlere yönlendirilmesi için veri katmanındaki yapılandırma gösterilmektedir. Şekil 4.12'de gösterilen kuralların satırlarının renkleri Şekil 4.11'deki ile aynıdır. Mavi ile gösterilen satırlar akış eşleşme şartlarını, yeşil ile gösterilen satırlar anahtarda gerçekleştirilecek işlemleri ifade etmektedir. Tuzak sistemlerin gerçekleştirdiği işlemler de şekilde kutu içinde açıklanmıştır. Zararlı trafik anahtar A tarafından algılandığında paketine NSH başlığı eklenerek bu kaynak IP adresi için 25 numaralı servis zincirinin işletilmesi süreci başlatılmaktadır. Bu zincirde iki sanal ağ fonksiyonu yer almaktadır. Bu zincirde trafik, 255 nolu servis indeksi ile tuzak sistem 1'e, 254 nolu servis indeksi ile de tuzak sistem 2'ye yönlendirilmektedir (1). Sınıflandırıcı olarak görev yapan anahtar A, zincirdeki 255 nolu indeksteki tuzak sisteme ulaşması için paketi ilgili porta yönlendirir. Anahtar B, paketin NSH başlığına bakarak trafiğin nasıl yönlendirileceğine karar verir. Anahtar, NSH servis numarası 25 ve servis indeksi 255 olan paketleri tuzak sisteme yönlendirir (2).

Tuzak sistem ilgili paketi işleyerek nasıl bir cevap döneceğine karar verir ve ilgili cevap paketini oluşturur. Tuzak sistem, pakete uygun NSH servis yolu indeksi, servis indeksi ve NSH üst veri bilgilerini içeren NSH başlığı ekler. NSH üst verileri anahtarlarda kural tanımlamak veya diğer ağ fonksiyon servisleri ile bilgi paylaşmak için kullanılabilmektedir.

Bu örnekte üst verinin değeri 1 olduğu zaman tuzak sistem 2'nin işletilmesi, 0 olduğu zaman da zincirin çıkışa yönlendirilmesi planlanmıştır. Tuzak sistem 1, bir sonraki servis fonksiyonunun işletilmesi için servis indeksini bir azaltır ve işlem sonucu paketi tekrar anahtar B'ye gönderir. Anahtar B, paketin işlenme sürecine devam edilmesi için paketi bir sonraki servise yönlendirir (3). Anahtar C, NSH üst verisine bakarak paketin 254 numaralı servise yönlendirilip yönlendirilmeyeceğine karar verir (4a.1). NSH üst verisinin değeri 1 olduğu durumda paket tuzak sistem 2'ye yönlendirilir. Tuzak sistem 2, paketi işler ve NSH servis indeksini bir azaltarak tekrar anahtar C'ye gönderir. Anahtar C, paketi servis zincirinden çıkacağı anahtara yönlendirir (5a.2).



Şekil 4.12. Tuzak sisteme gönderme servisi tarafından gerçekleştirilen örnek ağ yapılandırması

NSH üst verisinin değeri 0 olduğu durumda anahtar C, paketi tuzak sisteme göndermeden doğrudan servisin çıkış yapacağı anahtara yönlendirir (4b). Sonraki anahtar, paketi çıkış yapacağı anahtara gidebilmesi için NSH paket bilgilerini kullanarak yönlendirmeye devam

eder (5a, 5b). Paket, Anahtar A'dan çıkış yapmadan önce NSH başlığı çıkarılır ve ilgili çıkış bağlantısına yönlendirilir (6a, 6b).

Zararlı trafiğin tuzak sisteme gönderilmesi için veri katmanındaki yapılandırmaların belirlenmesi amacıyla gerçekleştirilen adımlar Şekil 4.13'te açıklanmıştır.

```
   Girdi : kaynak_dp,p,ip, hedef_ip, aldatıcı_dp,p,ip, öncelik, nsh_spi, nsh_si
1  rota ← en_kısa_yol(kaynak_dp,p, aldatıcı_dp,p)
2  eşleme_eylemler ← {}
3  for each düğüm ∈ rota do
4      girdi_p ← düğüm[girdi], çıkış_p ← düğüm[çıkış]
5      if uzunluk(rota) > 1 then
6          if düğüm is first then
7              eşleme ← {ipv4_kaynak = kaynak_ip, ipv4_hedef = hedef_ip, in_p = girdi_p}
8              eylemler =
                 [sarmala_nsh(), alanı_güncelle(nsh_spi), alanı_güncelle(nsh_si),
                  sarmala_eth(), çıkış(çıkış_p)]
9          else if düğüm is orta then
10             eşleme ← {nsh_spi = nsh_spi, nsh_si = nsh_si}
11             eylemler = [çıkış(çıkış_p)]
12         else
13             eşleme ← {nsh_spi = nsh_spi, nsh_si = nsh_si}
14             eylemler = [çıkar_eth(), çıkar_nsh(), çıkış(çıkış_p)]
15     else
16         eşleme ← {ipv4_kaynak = kaynak_ip, ipv4_hedef = hedef_ip, in_p = girdi_p}
17         eylemler = [çıkış(çıkış_p)]
18     eşleme_eylemler[düğüm] ← (eşleme, eylemler)
19 kurulum_sırası ← rota, kurulum_sırası.ekle(rota[0]), sil kurulum_sırası[0]
20 for each düğüm ∈ kurulum_sırası do
21     akış_mesajı ← akışı_oluştur(eşleme_eylemler[düğüm], öncelik,
                                   bekleme_süresi = çr_per)
22     akış_no ← gönderme_talep_et(akış_mesajı)
```

Şekil 4.13. Tuzak sisteme gönderme servisi tarafından tuzak sisteme yönlendirme rotalarının oluşturulması

Servis başladığında, tuzak sisteme giden rota hesaplanır (satır 1) ve parametreler ilk değerlerine getirilir (satır 2). Rota belirlendikten ve ilk değerler atandıktan sonra, servis her anahtarda akış oluşturur. Rotadaki ilk anahtar, paket sınıflandırıcısı olarak hareket eder. Paket kriterlerle eşleştiğinde, NSH pakete eklenir ve servis yolu indeksi (nsh_{spi}) ve servis indeksi (nsh_{si}) güncellenir (satır 7 - 8). Rotadaki diğer anahtarlar paketleri kriterleri eşleştirmek için yalnızca nsh_{spi} ve nsh_{si} değerlerini kullanır ve son anahtara kadar iletilmesini sağlar (satır 10 - 11). Son anahtarda, ağ servisi başlığı paketten çıkarılır (satır 16 - 17). Servis,

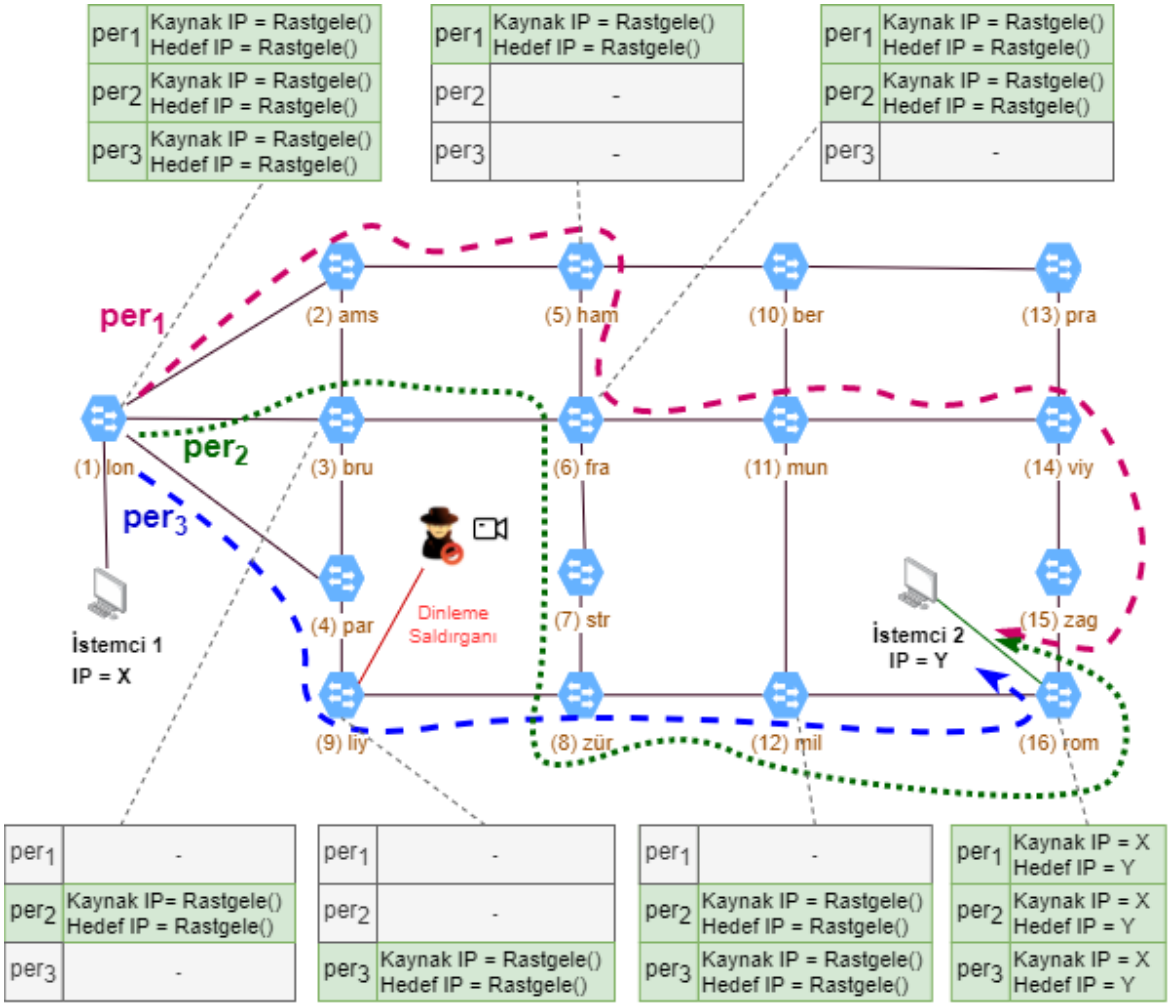
en sonunda her anahtar için belirlenen eşleşme ve eylem alanları ile akışların oluşturması talebinde bulunur (satır 19 - 22).

Tuzak sisteme gönder servisi, aynı adımlarla ters rotayı da oluşturur. Ağ servisi başlığı, ağ seviyesinde servis oluşturabilmenin en temel gereksinimidir. NSH başlığını kullanarak servisler zincirlenebilir veya paketler birden çok servis tarafından işlenebilir [190].

4.2.4. Çoklu rota servisi

Ağda bir akış başladığında aktif olan rota devre dışı bırakılmadığı veya tıkanıklık oluşmadığı sürece akışın tüm paketleri aynı iletim yolunu izler. Bu özellik, saldırganların ağ akışlarını dinlemelerine imkân veren bir zayıflığa dönüşebilmektedir. Saldırganlar ağ akışını dinleyerek iletişim verilerini (IP adresleri, çalışan servisler, kişisel veriler, gizli veriler vb.) kolayca yakalayabilir. Çoklu rota servisi YTA ağında düzenli olarak akış rotaları ile kaynak ve hedef IP adreslerini değiştirerek belirli anahtarları ele geçiren dinleme saldırganlarının akıştaki tüm trafiği ele geçirmelerini engeller. Bağlantı başladıktan sonra belirlenen aralıklarda yeni rota ve rastgele IP adresleri için veri katmanı tekrar yapılandırılır.

Şekil 4.14'te çoklu rota servisinin belirli zaman aralıklarında rotaları ve akışın IP adreslerini rastgele değiştiren örnek senaryo açıklanmaktadır. Şekil 4.14'te dinleme saldırganı 9 numaralı anahtarı dinlemektedir. İstemci 1 ile İstemci 2 arasındaki trafiğin rotaları belirli periyotlarda değiştirilerek tüm paketlerin farklı rotaları takip etmesi sağlanmaktadır. Şekilde üç farklı zaman aralığında trafiğin izlediği rotalar gösterilmektedir. İlk zaman aralığında (per_1) trafik 1 - 2 - 5 - 6 - 11 - 14 - 15 - 16 numaralı anahtarları izlemektedir. Bu zaman aralığında dinleme saldırganı trafiğin hiçbir paketini ele geçirememektedir. İkinci zaman aralığında (per_2) trafik 1 - 3 - 6 - 7- 8 - 12 - 16 numaralı anahtarları takip etmektedir. Bu zamana aralığında da dinleme saldırganı paketleri dinleyememektedir. Üçüncü zaman aralığında (per_3) ise paketler 1 – 4 – 9 – 8 - 12 -16 numaralı anahtarlardan geçmektedir. Bu zamana aralıklarında anahtarlarda paketlerin IP adresleri rastgele olarak güncellenerek akışın dinleme saldırganı tarafından takip edilememesi sağlanmaktadır. Bu işlemler çoklu rota servisi tarafından planlanmakta ve tanımlanan periyotlarda rotalar değiştirilirken her bir anahtarda akış karartma (obfuscation) uygulanmaktadır.



Şekil 4.14. Çoklu rota servisi tarafından farklı zaman aralıklarında gerçekleştirilen örnek ağ yapılandırmaları

Dinleme saldırganı dinlediği anahtara ve zamana göre paketleri hiç dinleyememekte veya belirli zamanlarda belirli bölümlerini dinleyebilmektedir. Örneğin saldırgan, 3 numaralı anahtarı dinlediğinde paketleri sadece per_1 zaman aralığında, 3 numaralı anahtarı dinlediğinde sadece per_2 zaman aralığında ve 12 numaralı anahtarı dinlediğinde ise per_2 ve per_3 zaman aralıklarında dinleyebilmektedir. Dinleme saldırganı 9 numaralı anahtardan bu zaman aralığında paketleri dinleyebilmektedir. İstemci 1'in bağlı olduğu anahtarda kaynak ve hedef IP adresleri rastgele değiştirilmektedir. Rotadaki her bir anahtarda da IP adresleri rastgele olarak değiştirilmektedir. Böylelikle birden fazla anahtar dinlense de elde edilen bilgiler ağ üzerinden izlenememektedir. Bu sayede belirli periyotta anahtarlardan geçen akışlar rastgele IP adresleri ile karartılmaktadır. Örneğin dinleme saldırganı 4 ve 9 numaralı anahtarları aynı anda dinlemesi durumunda her bir anahtardan dinlenen paketlerdeki IP adresleri farklı olmaktadır. Paketler son anahtarda gerçek IP adreslerine dönüştürülerek

istemciye gönderilmektedir. Hangi anahtarda hangi rastgele IP adreslerinin kullanılacağı kontrol katmanında belirlenmekte ve belirli periyotlarla tetiklenerek yeni rotalarla birlikte veri katmanı yapılandırılmaktadır.

Şekil 4.15'te çoklu rota yönetim algoritması açıklanmıştır.

Girdi : $kaynak_{dp,p,ip}, hedef_{dp,p,ip}, rsg_{e,anah,ağ}$, $çr_{per,maks,sayı,esik}$, $öncelik_{min}$, $çr\_listesi$

$akış \leftarrow (kaynak_{dp,p,ip}, hedef_{dp,p,ip})$
if $akıs \in çr\_listesi$ **then**
  return;
$çr\_listesi.ekle(akış)$
$i_{baslangıc} \leftarrow -1$, $i_{sonraki} \leftarrow 0$, $rotalar_{kurulmus} \leftarrow \{\}$, $rotalar_{muhtemel} \leftarrow \{\}$, $rota\_sayısı \leftarrow -1$
$durum \leftarrow AKTİF\_DEĞİL$, $öncelik \leftarrow öncelik_{min}$
while $durum \neq TAMAMLANDI$ **do**
  if $durum = AKTİF\_DEĞİL$ **then**
    $rotalar_{muhtemel}, rota\_sayısı \leftarrow optimum\_rotaları\_hesapla(akış, çr_{maks})$
    $durum \leftarrow BAŞLATILDI$
  if $durum = BAŞLATILDI$ **or** $durum = AKTİF$ **then**
    if $uzunluk(rotalar_{kurulmus}) < çr_{sayı}$ **then**
      $öncelik \leftarrow öncelik + 1$
      $rota \leftarrow akış\_kuralı\_oluştur(rotalar_{muhtemel}, i_{sonraki}, çr_{per}, öncelik, rsg_{e,anah,ağ})$
      $rotalar_{kurulumus}.ekle((rota, şimdiki\_zaman))$
      if $durum = BAŞLATILDI$ **then**
        $rota_{active} \leftarrow rotalar_{muhtemel}[i_{sonraki}]$
        $durum \leftarrow AKTİF$
    for each $rota, olusturulma\_zamanı \in rotalar_{kurulmus}$ **do**
      if $olusturulma\_zamanı + çr_{per} \leq simdiki\_zaman$ **then**
        $rota_{active} \leftarrow rotalar_{muhtemel}[i_{baslangıc} + 1]$
        $sil\ rotalar_{kurulmus}\ [i_{sonraki}]$
        $i_{baslangıc} \leftarrow i_{baslangıc} + 1$
  else if $durum = SONLANDIRILIYOR$ **then**
    $parametreleri\_ilkdeğerlerine\_getir()$
    $durum \leftarrow TAMAMLANDI$
    $sil\ çr\_listesi[akış]$
  $bekleme\_süresi \leftarrow çr_{per} - rastgele(çr_{esik})$
  $bekle(bekleme\_süresi)$
  if $akıs\_bitti\_mi(akıs)$ **then**
    $durum \leftarrow SONLANDIRILIYOR$

Şekil 4.15. Bir akış için çoklu rotaların yönetimi

Bu algoritmanın girdileri aşağıda belirtilmiş ve kısaca açıklanmıştır:

- $kaynak_{dp,p,ip}$: Sırası ile kaynak anahtarı veri yolu, bağlantı noktası ve IP adresi değerlerini temsil eder.

- $hedef_{dp,\ p,\ ip}$: Sırası ile hedef anahtar veri yolu, bağlantı noktası ve IP adresi değerlerini temsil eder.
- $rsg_{e,\ anah,\ ağ}$: rsg_e değeri, çoklu rota servisi ile rota oluşturduğunda rastgele IP adreslerinin kullanılıp kullanılmayacağını belirler. Çoklu rota servisi ile rastgele IP kullandığı durumda IP adresleri $rsg_{ağ}$ değeri ile belirlenen alt ağda oluşturulur. rsg_{anah} değerinin doğru olması durumunda rastgele IP adresleri oluşturma işlemi her bir YTA anahtarında gerçekleştirilir. rsg_{anah} değerinin yanlış olması durumunda rastgele IP değerleri sadece giriş anahtarında oluşturulur.
- $çr_{per,\ maks,\ sayı,\ eşik}$: $çr_{per}$ değeri çoklu rota servisinin rotayı saniye cinsinden ne aralıklarda değiştireceğini belirtir. Rotalar seçilirken maksimum $çr_{maks}$ uzunluğunda olması sağlanır. Bir sonraki rotayı hazırlamak için sonraki periyottan ne kadar zaman önce işleme başlanacağı $çr_{eşik}$ değerinde belirtilen eşik süresi ile belirlenir. Bir akış için bir anahtarda aynı anda oluşturulacak kuralların sayısını sınırlamak için $çr_{sayı}$ değeri kullanılır. Anahtarda $çr_{sayı}$ değeri kadar kural oluşturularak rota değişimlerinde paket kayıplarının önüne geçilir.
- $öncelik_{min}$: Birden fazla rota oluşturulduğunda trafik büyük önceliğe sahip olan akışı takip eder. İlk rota için oluşturulan akışların öncelik değeri $öncelik_{min}$ olarak belirlenir.
- $çr_{liste}$: Çoklu rota servisi, yönettiği ve aktif olan çoklu rota akışlarını $çr_{liste}$ listesinde saklar.

Çoklu rota servisi çağrıldığında, servis öncelikle aktif bir akışın olup olmadığını kontrol eder. Akış $çr_{liste}$ listesinde varsa işlemi sonlandırır (satır 2 - 3). Diğer durumda, akış $çr_{liste}$ listesine (satır 4) aktif akış olarak eklenir ve servis değişkenleri (satır 5 - 6) ilk değerlerine atanır. Çoklu rota servisi, akışların durumlarını yönetir ve durum TAMAMLANDI (satır 7) olarak belirlenene kadar rotaları değiştirir. Servis başlatıldığında, akışın başlangıç durumu AKTİF_DEĞİL olarak belirlenir. Bu durumda rotalar, rotaların maliyetlerine göre oluşturulur, rastgele sıralanır ve muhtemel yollar listesinde saklanır (satır 9). Rotalar oluşturduktan ve sıralandıktan sonra, durum BAŞLATILDI olarak değiştirilir. Durum BAŞLATILDI veya AKTİF olduğu durumda veri düzleminde yer alan aktif rota sayısı $çr_{eşik}$ değerinden küçükse yeni rotalar oluşturulur ve anahtarlara yüklenir. Servis, akış eklerken her akış için boş bekleme süresini $çr_{per}$ değeri olarak ayarlar.

Akış kuralı eklemek için Şekil 4.16'da belirtilen adımlar takip edilir. Her bir rota için akış öncelik değeri artırılır ve oluşturulan rota bilgileri aktif rotalar listesinde saklanır (satır 12 -

15). İlk rota oluşturulduğunda, kurulu rota aktif rotalar listesine eklendikten sonra akışın durumu ETKİN olarak değiştirilir. Yüklenen her rotanın sona erme durumu kontrol edilir. Rotanın süresi dolmuşsa, rota aktif rotalar listesinden silinir (satır 19 - 23). Aynı eşleşme kriterlerine sahip yeni rota eklendiğinde, diğer aktif akışlar, boş bekleme süresi boyunca anahtarlarda kalır ve süre sonunda silinir. Çoklu rota servisi, yukarıda belirtilen işlemleri yaptıktan sonra bekleme süresi boyunca işlem yapmaz. $çr_{eşik}$ parametresi genellikle bir veya birden küçüktür, bu nedenle bekleme süresi ($çr_{per}$ - 1) ile $çr_{per}$ değer aralığında rastgele olarak belirlenir (satır 28). Çoklu rota servisi, her periyotta akışın bitip bitmediğini kontrol eder. Akışın tamamlandığı anlaşıldığında akışı $çr_{list}$ listesinden siler ve süreci bitirir (satır 24-27 ve 30-31). Akış tamamlanmadıysa rotaların oluşturulmasına devam edilir.

Rastgele IP adresleri ile rotanın oluşturulması için akış kuralları Şekil 4.16'da sunulan adımlar takip edilerek oluşturulmaktadır. Oluşturulacak rota, girdi parametresi olan indeks değeri kullanılarak muhtemel rotaların arasından seçilmekte ve işlemlerde kullanılacak değişkenlerin ilk değerleri atanmaktadır (satır 1 - 2). Seçilen rotadaki her bir anahtara yüklenecek eşleme kriterleri ve eylemler sırası ile belirlenmektedir (satır 3 - 23). Belirleme aşamasında öncelikle anahtardaki giriş ve çıkış bağlantı noktaları belirlenmektedir (satır 4). Rastgele IP üretilmesini belirleyen parametre (rsg_e) değeri evet olarak seçildiğinde ve rotada birden fazla anahtar yer aldığı durumda rotadaki farklı anahtarlar için farklı eşleme kriterleri ve eylemleri belirlenir. Rotadaki ilk anahtarda eşleme kriteri olarak; kaynak IP, hedef IP, giriş bağlantı noktası belirlenir (satır 7), sonrasında rastgele IP adresleri üretilerek paketlerin kaynak ve hedef IP adresleri bu adreslerle değiştirilmesini ve çıkış bağlantı noktasına yönlendirilmesini sağlayan eylem listesi oluşturulur (satır 8 - 9). Rotanın ortasında bulunan anahtarlar için eşleme kriteri olarak bir önceki anahtarda oluşturulan rastgele IP adresleri ile giriş bağlantı noktası kullanılır (satır 11). Her anahtarda rastgele IP adresi üretilmesini belirleyen parametre (rsg_h) değeri evet olarak seçilmesi durumunda eylem listesi ilk anahtar için belirlenen şekilde oluşturulur. Aksi durumda paketin sadece çıkış bağlantı noktasına yönlendirecek eylem oluşturulur (satır 12 - 16). Rotanın son anahtara yüklenecek eşleme kriterleri orta anahtarlardakine benzer şekilde belirlenir ve bu anahtardaki eylem listesine paketin çıkış bağlantı noktasına yönlendirmeden önce IP adreslerinin gerçek IP adresleri ile değiştirilmesini sağlayan eylemler eklenir (satır 18 - 19).

```
Girdi : rotalar_muhtemel, indeks, öncelik, çr_per, rsg_e,h,net
1  rota ← rotalar_muhtemel[indeks]
2  kaynak_rsg_ip ← 0, hedef_rsg_ip ← 0, eşleme_eylemler ← {}
3  for each düğüm ∈ rota do
4      giriş_p ← düğüm[giriş], çıkış_p ← düğüm[çıkış]
5      if rsg_e = Evet and uzunluk(rota) > 1 then
6          if düğüm is ilk then
7              eşleme ← {ipv4_kaynak = kaynak_ip, ipv4_hedef = dst_ip, in_p = giriş_p}
8              kaynak_rsg_ip ← rastgele_ip(rsg_net), hedef_rsg_ip ← rastgele_ip(rsg_net)
9              eylemler = [alanı_güncelle(kaynak_rsg_ip), alanı_güncelle(hedef_rsg_ip),
                           çıkış(çıkış_p)]
10         else if düğüm is middle then
11             eşleme ← {ipv4_src = kaynak_rsg_ip, ipv4_hedef = kaynak_rsg_ip,
                         in_p = giriş_p}
12             if rsg_h = Evet then
13                 kaynak_rsg_ip ← rastgele_ip(rsg_net), hedef_rsg_ip ← rastgele_ip(rsg_net)
14                 eylemler = [alanı_güncelle(kaynak_rsg_ip), alanı_güncelle(hedef_rsg_ip),
                               çıkış(çıkış_p)]
15             else
16                 eylemler = [çıkış(çıkış_p)]
17         else
18             eşleme ← {ipv4_kaynak = kaynak_rsg_ip, ipv4_hedef = kaynak_rsg_ip,
                         in_p = giriş_p}
19             eylemler = [alanı_güncelle(kaynak_ip), alanı_güncelle(hedef_ip),
                           çıkış(çıkış_p)]
20     else
21         eşleme ← {ipv4_kaynak = kaynak_ip, ipv4_hedef = hedef_ip, in_p = giriş_p}
22         eylemler = [çıkış(çıkış_p)]
23     eşleme_eylemler[düğüm] ← (eşleme, eylemler)
24 kurulum_sırası ← path, kurulum_sırası.append(path[0]), sil kurulum_sırası[0]
25 for each düğüm ∈ kurulum_sırası do
26     akış_mesajı ← akışı_oluştur(eşleme_eylemler[düğüm], öncelik,
                                   bekleme_süresi = çr_per)
27     akış_id ← gönderme_talep_et(akış_mesajı)
```

Şekil 4.16. Çoklu rotalar için akış kurallarının oluşturulması ve anahtarlara yüklenmesi

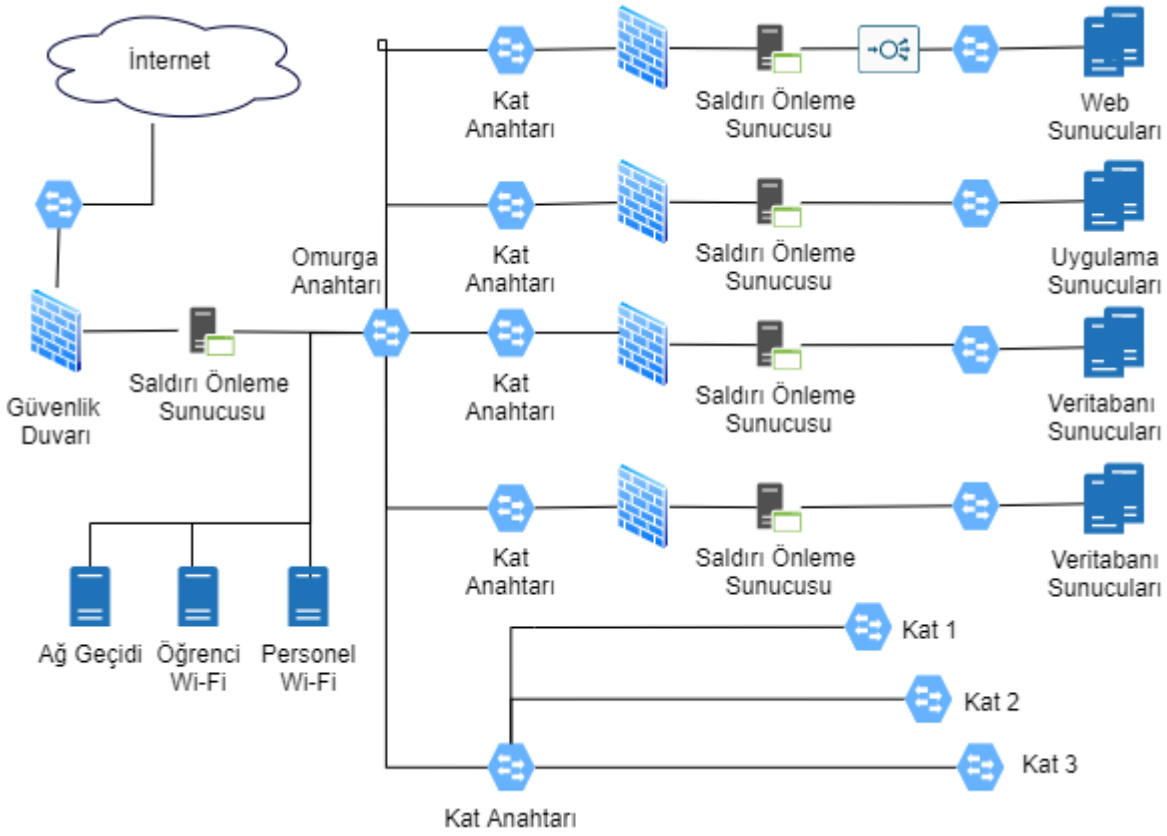
Rastgele IP üretilmesini belirleyen parametre (rsg_e) değeri hayır olarak seçildiyse veya rotada sadece bir anahtar olması durumunda, normal bir anahtar yönlendirmesi sağlanır. Eşleme kriterleri olarak kaynak IP, hedef IP ve giriş bağlantı noktası kullanılır ve paketin çıkış bağlantı noktasına yönlendirilmesini sağlayacak eylem eklenir (satır 20-22). Her bir anahtar için üretilen eşleme kriterleri ve eylem listeleri toplanarak (satır 23) ilk anahtara en son yükleme yapılmasını sağlayacak şekilde yükleme sırası belirlenmektedir (satır 24). Bu şekilde rota aktif olduğunda paket kayıplarının önüne geçilmesi sağlanmaktadır. Yükleme sırası dikkate alınarak bekleme süresi, öncelik, eşleme ve eylem listeleri ile akış mesajları oluşturulmakta, oluşturulan akış mesajının ilgili anahtara gönderilmesi için YTA-AY'ye talep gönderilmektedir (satır 24 -27).

5. YTA İÇİN ÖNERİLEN SİBER SAVUNMA SİSTEMİNİN DEĞERLENDİRMESİ

Bu bölümde, öncelikle Y-S3'ün geliştirilmesinde ve test edilmesinde kullanılan altyapılar açıklanmaktadır. Sonrasında ağ savunma servislerinin senaryolara göre (Bkz. Şekil 4.8a ve Şekil 4.8b) kullanımı için gerçekleştirilen testler açıklanmakta ve test sonuçları değerlendirilmektedir.

5.1. STİ Verilerinin Elde Edilmesi ve Testi için Oluşturulan Altyapı

Çalışmanın ilk aşamasında STİ verilerinden ağ seviyesi tehdit göstergelerini çıkartmak için Python programlama dili ile prototip bir uygulama geliştirilmiştir. YTA kontrolcüsü olarak Opendaylight YTA kontrolcüsü [191] ve Şekil 5.1'de gösterilen kurumsal ağ topolojisi kullanılarak Mininet [158] ağında testler gerçekleştirilmiştir.



Şekil 5.1. STİ verilerinin değerlendirmesi için kullanılan ilk test ağ topolojisi

STİ verilerini toplamak için çalışmada STIX v1.2 formatını destekleyen sunucular seçilmiştir. STİ verileri paylaşan TAXII sunucularından [192–195] siber tehdit istihbarat verisi toplanmıştır. Çalışma kapsamında aşağıdaki kurallara göre STİ verileri işlenmiştir.

- Siber tehdit istihbarat verilerinden K&K sunucusu, IP listesi ve anonim bağlantı verilerini içerenler seçilmiştir.
- STIX gösterge verilerinde güvenilirlik değeri dikkate alınarak STİ verilerinin güvenilirliği belirlenmiştir. Güvenilirlik değeri olmayan verilerde; gözlemlenme sayısı, veriyi üreten, istihbaratı paylaşan kaynak sayısı gibi bilgiler kullanılarak hesaplama yapılmıştır.

Bu çalışmada STİ verilerinden çıkartılabilecek ağ göstergeleri belirlenmiştir. Altı farklı veri kümesinden STİ verisi alınarak testler gerçekleştirilmiştir. Veri kümelerindeki STİ verilerinden çıkartılabilen ağ göstergeleri ve üretilebilen savunma sayıları ölçülmüş ve Çizelge 5.1'de özetlenmiştir. STİ verilerinden elde edilen göstergeler kullanılarak YTA kontrolcüsü kuzey yönlü ara yüzü ile sunulan kural oluşturma servislerinin parametreleri oluşturulmuştur. Bu yöntem ile oluşturulan test ağında; alan savunması, karantinaya alma ve bal küpüne yönlendirme servisleri test edilmiştir.

Çizelge 5.1. Testlerde kullanılan STİ veri kaynakları ve veri kümeleri

Veri Kümesi Kaynağı	Veri Kümesi Adı	Toplam STİ Veri Sayısı	Ağ Göstergesi İçeren STİ Veri Sayısı
http://hailataxii.com	guest.MalwareDomain List_Hostlist	1236	1236
http://hailataxii.com	guest.Abuse_ch	549	296
http://edge.threatactorlab. com	stix-data	381	92
http://hailataxii.com	guest.phishtank_com	9730	0
https://otx.alienvault.com	user_AlienVault	889	0
http://hailataxii.com	guest.Lehigh_edu	502	0

Bu çalışmalar ile STİ verileri içinde ağ seviyesi savunma için kullanılabilecek bilgilerin yetersiz olduğu, STİ verilerinin daha yapısal formatta paylaşılması gerektiği görülmüştür. Tez çalışmasının sonraki aşamalarında, otomatik ağ savunması yapılabilmesi için STİ veri formatlarında olması gereken bilgilerin neler olması gerektiğine ve bu bilgilerin nasıl kullanabileceğine odaklanılmıştır.

5.2. Geliştirme ve Test Altyapısı

Y-S3'ün prototipi, yeni savunma servislerinin eklenebilmesine imkan sağlayacak şekilde Python programlama dili ve Ryu YTA çerçevesi [157] kullanılarak geliştirilmiştir. Geliştirilen prototipte ağ altyapısı olarak Mininet v2.3.0d5 [158], ağ anahtarı olarak Open vSwitch v2.9.5 [159] kullanılmıştır. Geliştirilen prototipteki bileşenler, OpenFlow 1.4.1 [17] mesajlarını kullanarak veri katmanı ile iletişim kurmaktadır. Servisler, Open vSwitch tarafından desteklenen NSH ve OpenFlow ile tanımlanan standart eylemleri kullanabilmektedir.

5.2.1. Ryu YTA çerçevesi ve YTA kontrolcüsü

Ryu, Python programlama dili kullanılarak geliştirilmiş ve kolay şekilde YTA uygulamalarının hayata geçirilebilmesini sağlayan bileşen temelli bir çerçevedir. Ryu çerçevesi OpenFlow haricinde Netflow, OF-config gibi protokolleri de desteklemektedir. Açık kaynak kodlu olması, olay tabanlı olarak tüm ağ aktivitelerinin izlenebilmesi ve ağdaki LLDP paketlerini analiz ederek anahtar bağlantı noktaları ile ilgili bilgileri toplayan hazır bileşenleri içermesinden dolayı bu çerçeve kullanılarak YTA kontrolcüsü oluşturulmuştur. Ryu çerçevesinde yer alan sınıflardan türetilerek kolaylıkla YTA kontrolcüsü oluşturulup çalıştırılabilmektedir. Sistem prototipi için YTA kontrolcüsünün yanında diğer bileşenler de Ryu çerçevesinin sunduğu özellikler kullanılarak geliştirilmiştir.

NSH protokolü birçok YTA kontrolcüsü tarafından henüz desteklenmemektedir. Ryu çerçevesi açık kaynak kodlu olduğu için NSH protokolünün desteklenmesi için Ryu kaynak kodları tez kapsamında güncellenmiş ve açık kaynak projesi ile paylaşılmıştır.

5.2.2. Open vSwitch - YTA sanal anahtarı

Open vSwitch, C programlama dili ile geliştirilen çok katmanlı yazılım temelli sanal anahtardır. Linux işletim sistemlerini ve Linux temelli sanallaştırma platformlarını [XEN, Kernel Virtual Machine (KVM) vb.] desteklemektedir. OpenFlow protokolleri ile birlikte NetFlow, sFlow, NSH, sanal genişletilebilir yerel alan ağı (virtual extensible local area network, VxLAN), Genel Yönlendirme Sarmalama Protokolü (Generic Routing Encapsulation, GRE) gibi protokolleri / özellikleri de desteklemektedir. Sanal anahtar, Linux ve veri katmanı geliştirme aracı (Data Plane Development Kit, DPDK) donanımları ile çalışabilmektedir. Sanal ortamlarda hızlı şekilde ağ yapılandırmalarının yapılabilmesi, mantıksal olarak sanal ağların oluşturulabilmesi ve mobil ağ altyapılarının yönetilmesi için aktif kullanılmaktadır. Open vSwitch yazılımı; sanal anahtarının yönetimi, testi ve yapılandırması için gerekli araçlarla birlikte kurulmaktadır.

5.2.3. Mininet - YTA sanal ağ emülatörü

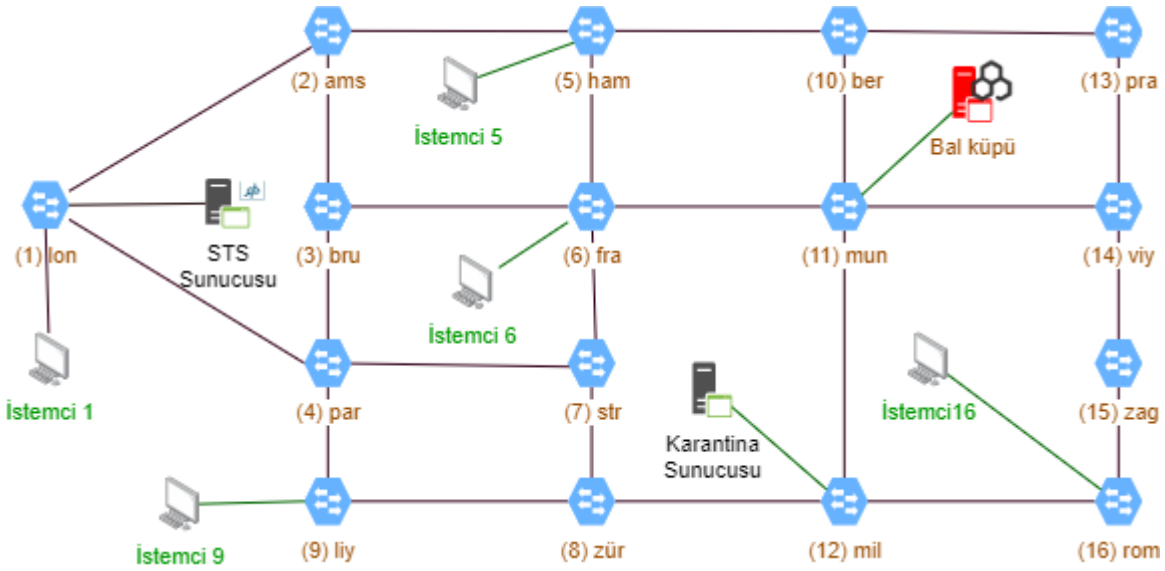
Mininet, sanal istemciler, anahtarlar, kontrolcüler ve bağlantılar dahil olmak üzere YTA temelli sanal ağlarının oluşturulmasını sağlayan bir ağ emülatörüdür. Python ve C programlama dili kullanılarak geliştirilmiştir. Mininet proses temelli sanallaştırma yaparak birden fazla anahtarı izole şekilde Linux işletim sistemi üzerinde çalıştırabilmektedir. Mininet, istemcileri ve anahtarları sanal Ethernet çifti (VETH) ile birbirine bağlamaktadır. Bu şekilde her izole ağ içinde farklı ağ ara yüzleri, yönlendirme tabloları, ARP tabloları oluşturulmaktadır. Sadece Linux işletim sistemlerine bağımlı olması ve sadece bir işletim sisteminde çalışabilmesi gibi kısıtları olmasına rağmen Mininet YTA sanal test altyapısının çok fazla avantajları vardır [158]:

- Hızlı şekilde kurulumu yapılabilmektedir. Sanal makinelerde kurulumu yapılmış şekilde dağıtımı yapılabilmektedir.
- Çok hızlı şekilde başlatılabilmekte ve ilk test ortamına geri dönülebilmektedir.
- Yüzlerce sanal istemci ve anahtar oluşturulabilmektedir.
- İstenilen ağ topolojisi maliyetsiz olarak oluşturulabilmektedir.
- Yüksek bant genişliği sunabilmektedir.
- Open vSwitch anahtarları kullanılmakta ve gerçek ağlara bağlantı yapılabilmektedir.

5.2.4. Test topolojisi ve test ortamı

Geliştirilen prototip test ağ topolojilerinden bağımsızdır. Prototip, YTA ağ topolojilerini öğrenebilmekte ve ağı yönetebilmektedir. Y-S3'ü değerlendirmek için Maesschalck ve diğerleri [196] tarafından çekirdek topoloji olarak adlandırılan test ağ topolojisi (Şekil 5.2) kullanılmıştır. Bu test topolojisi çoklu rota servisinin test edilebilmesine uygun olması ve elde edilecek sonuçların literatürdeki diğer çalışmalarla karşılaştırılabilmesi için tercih edilmiştir.

Seçilen ağ topolojisi, yedekli iletişim yolları içermekte ve çoklu rota servisinin değerlendirilebilmesine olanak sağlamaktadır. Ağ topolojisinde 16 adet Open vSwitch anahtarı yer almakta ve her bir anahtara en az bir istemci bağlanmaktadır. Test ağ topolojisinde yer alan anahtarların isimleri Avrupa'daki şehir isimlerinin ilk üç karakterinden oluşmaktadır (Örneğin, lon – Londra, rom – Roma, ber – Berlin, par - Paris). Savunma servislerini test etmek için ağda STS, bal küpü ve karantina sunucusu konumlandırılmıştır.



Şekil 5.2. Y-S3'ün değerlendirmesinde kullanılan test ağ topolojisi

Testler, 16 GB RAM ve Intel Core i7-6600U 2.60GHz işlemciye sahip bir Ubuntu 18.04 işletim sistemi kurulu bilgisayar üzerinde gerçekleştirilmiştir. Ağ anormalliklerini izlemek için Suricata [197] STS kullanılmıştır. Tuzak sisteme gönderme servisini test etmek için tüm ICMP paketlerine gerçek hedefmiş gibi yanıt veren örnek bal küpü betikleri hazırlanmıştır.

Trafiği izlemek ve Anomali Tespit Yöneticisine uyarılar göndermek için bir ağ istemcisinde Suricata v5.0.3 çalıştırılmıştır. Betik olarak çalıştırılan bal küpü diğer bir istemcide çalıştırılmış ve bu istemcinin bir tuzak sistem olarak görev yapması sağlanmıştır. Savunma servislerinin performansını değerlendirmek için Scapy kütüphanesi kullanılarak hazırlanmış özel test betikleri, nmap v7.60 [198], hping3 v3 [199] ve iperf3 v3.7 [200] ağ test araçları kullanılmıştır.

5.3. Y-S3'ün ve Ağ Savunma Servislerinin Değerlendirmesi

Aşağıdaki alt bölümlerde Y-S3 kullanılarak farklı saldırı senaryoları ve savunma servisleri değerlendirilmektedir. Bu kapsamda öncelikle, önceki bölümlerde açıklanan (Bkz. Şekil 4.8a ve Şekil 4.8b) senaryolar değerlendirilmiştir. Sonrasında dinleme saldırıları için çoklu rota servisi test edilerek değerlendirilmiştir.

5.3.1. Değerlendirme ölçütleri

Geliştirilen servisler ağ performansı ve sağladığı güvenlik açısından çeşitli ölçütler kullanılarak değerlendirilmiştir. Literatürdeki çalışmalarda ağ güvenlik servislerinin değerlendirilmesi için farklı ölçütler tanımlanmıştır. Bu tez çalışmasında kullanılan ölçütler belirlenirken literatürdeki ilgili çalışmalarda yer alan ölçütlerden faydalanılmıştır. Veri katmanında yapılandırmalar gerçekleştirdikten sonra YTA anahtarlarının performansının ölçülmesi kapsam dışı bırakılmıştır. Her servisin değerlendirmesinde kullanılan ölçütler aşağıda verilmektedir:

1. Kara liste servisi için kara listeye alınacak IP adreslerinin belirlenmesi ve OpenFlow akış kurallarının yükleneceği anahtarların tespit edilerek ilgili anahtarlara kuralların yüklenmesi için geçen *ortalama kurulum süresi* ölçütü belirlenmiştir. Kara listeye alınan IP adresleri YTA anahtarlarına yüklendikten sonra ilgili paketler düşürüldüğü için ağ güvenliği sağlanmaktadır. Bu güvenliğin sağlanabilmesi için kara liste servisi tarafından yönetilen OpenFlow mesajlarının kontrolcü güney ara yüzünde oluşturduğu *bant genişliği* diğer bir ölçüt olarak belirlenmiştir.
2. Karantina servisi için kara liste servisine benzer şekilde, karantinaya alınacak istemcilerin belirlenmesi, istemciler ile karantina sunucusu arasında rotaların hesaplanması ve OpenFlow akış kurallarının yükleneceği anahtarların tespit edilerek

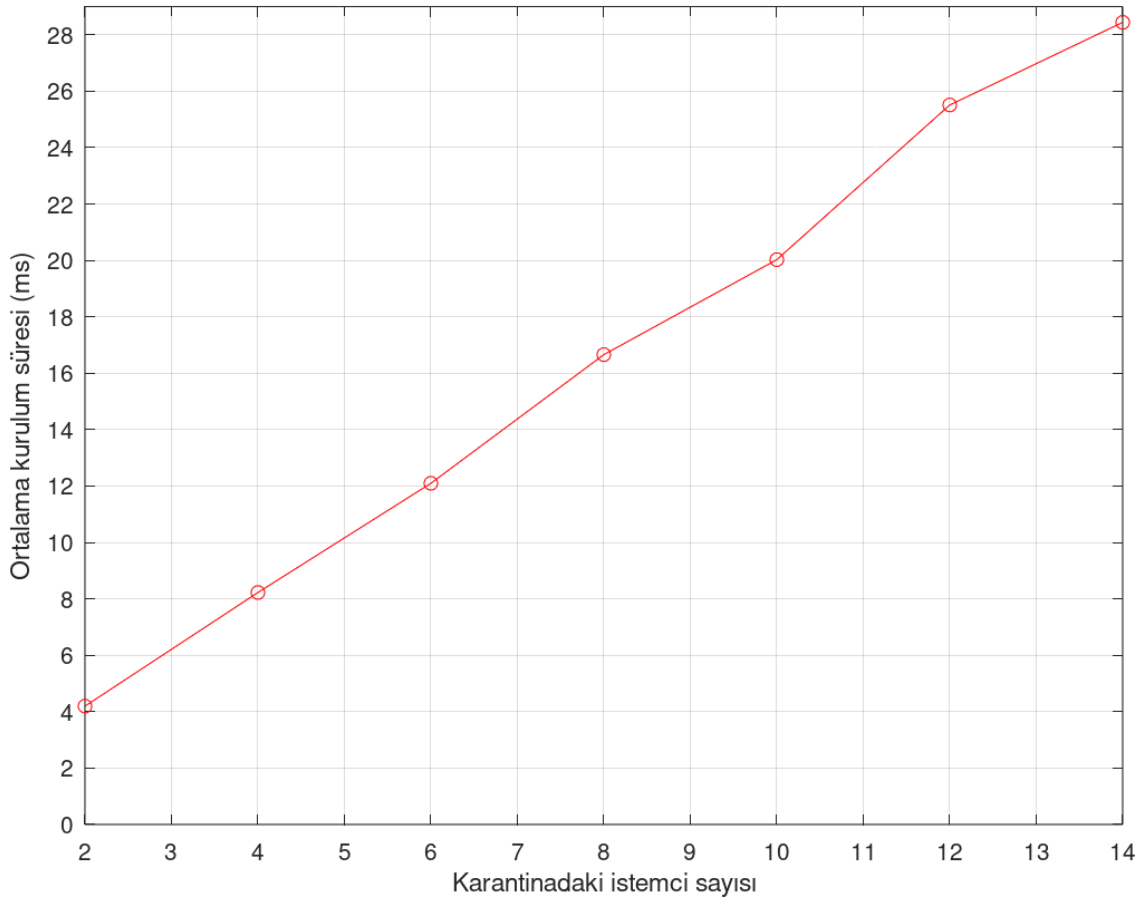
ilgili anahtarlara kuralların yüklenmesi için geçen *ortalama kurulum süresi* ölçütü belirlenmiştir.

3. Tuzak sisteme gönderme servisinin değerlendirilmesinde; tuzak sisteme gönderilecek akışların / paketlerin belirlenmesi, paketlerin tuzak sisteme gönderilmesi için gerekli rotanın hesaplanması, rotaların oluşturulması, rotadaki anahtarların tespit edilmesi, NSH protokolü kullanarak SFZ oluşturma kurallarının belirlenmesi ve anahtarların yapılandırması için geçen *ortalama kurulum süresi* ölçütü belirlenmiştir.
4. Çoklu rota servisi için maksimum rota sayılarına göre çoklu rota alternatiflerinin belirlenmesi, bu rotalar arasından bir tanesinin seçilerek veri katmanında yapılandırmanın gerçekleştirilmesi için geçen ortalama kurulum süresi ölçüt olarak belirlenmiştir. Farklı rota değiştirme periyotları kullanarak *anahtarlardan dinlenebilen verilerin yüzdesi* ve *rastgele IP adresi sayıları* da ölçüt olarak kullanılmıştır. Diğer bir ölçüt olarak, çoklu rotalar oluşturulurken anahtarlarda paketlerin IP adreslerinin rastgele değiştirilmesi ve saldırganlar tarafından birden fazla anahtar ele geçirilmesi durumlarında *aynı IP adresi ile dinlenebilen verilerin yüzdesi* seçilmiştir. Ayrıca çoklu rota servisi tarafından yönetilen OpenFlow mesajların kontrolcü güney ara yüzünde oluşturduğu *bant genişliği*, çoklu rotalardan dolayı *oluşan paket kayıp yüzdesi* ağ performans ölçütü olarak tanımlanmıştır.

5.3.2. Karantina ve kara liste servislerinin değerlendirmesi

Karantinaya alarak ve kara liste oluşturarak yapılan ağ seviyesi savunma senaryosu (Bkz. Şekil 4.8a) çoğu tehdit modeli için kullanılabilmektedir. Yapılan testte test ağında bulunan İstemci 1, DDoS saldırganı ve zararlı yazılım saldırganı için kontrol sunucusu ya da indirme saldırganının yönetimindeki zararlı bir internet sayfası olarak görev yapacak şekilde yapılandırılmıştır. Diğer istemciler İstemci 1'e bağlantı yapacak veya İstemci 1'den aldıkları komutları yerine getirecek şekilde konumlandırılmıştır. İstemci 1 üzerinde Nmap [198] test aracı çalıştırılarak saldırı trafiği üretilmiştir. Oluşturulan saldırı trafiğinin kaynak IP adresleri STİ verilerinden çıkartılan zararlı IP adreslerinden seçilmiş ve bu IP adresleri ile tüm ağ taranmıştır. STİ verilerinden çıkartılan zararlı IP adresleri kara liste servisine eklenmiştir. Trafik başladığında savunma servisleri paketleri işlemekte ve oluşturulması gereken akış değişiklik kurallarını belirlemektedir. YTA-AY tarafından akış kuralları anahtarlara yüklenerek zararlı trafik akışları engellenmektedir.

Gerçekleştirilen ilk testte, karantina akış kurallarının kurulum süreleri değerlendirilmiştir. Karantina servisi, bir istemci için karantina akışları yüklerken istemciden karantina sunucusuna giden rotayı bulmakta ve akışları rotadaki anahtarlara yüklemektedir. Gerçekleştirilen testte farklı sayılardaki zararlı IP adresleri ile farklı sayıdaki istemcilerin karantinaya alınma süreleri ölçülmüş ve test sonuçları Şekil 5.3'te grafik olarak gösterilmiştir. Test sonucunda 14 istemcinin aynı anda karantinaya alınması süresinin 28 milisaniye olarak gerçekleştiği gözlenmiştir.

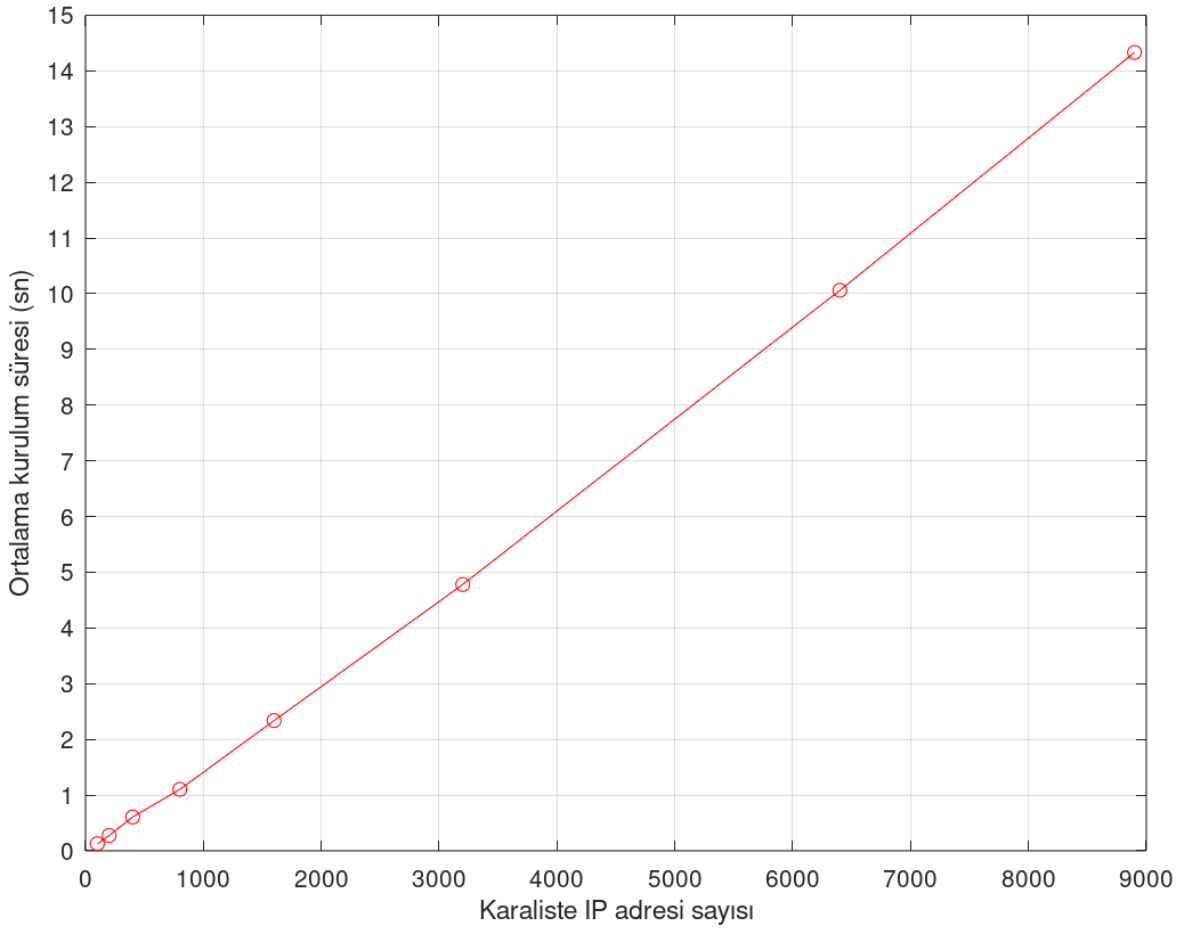


Şekil 5.3. Karantinaya alınacak istemciler için ortalama kurulum süresi

İkinci değerlendirmede, aynı test altyapısı ve savunma senaryosu (Bkz. Şekil 4.8a) kullanılarak kara listeye almak için gereken gecikme süresi ölçülmüştür.

Test verisi olarak, Ulusal Bilgisayar Olayı Müdahale Merkezi (USOM) tarafından yayımlanan [201] zararlı IP adresleri, URL ve etki alan adları işlenerek zararlı IP adresleri oluşturulmuştur. IP adresleri, URL ve etki alanı adları işlenirken IP adreslerinin doğrulaması (geçersiz IP, yerel IP, geçersiz alan adı vb.) yapılmış ve testlerde kullanılabilecek geçerli 8900 IP adresi elde edilmiştir. Bu testte SFZ oluşturmaya gerek duyulmadan sadece kara

liste ve karantina servisleri kullanılmıştır. İstemci 1 üzerinden Scapy kütüphanesi [202] kullanılarak geliştirilen özel test betikleri ile test trafiği üretilmiştir. Oluşturulan trafikte yer alan paketlerdeki kaynak IP adresi olarak bir test için oluşturulan zararlı IP adresleri kullanılmıştır. Üretilen trafikteki paketler YTA kontrolcüsüne gönderildiğinde, kara liste ve karantina savunma hizmetleri etkinleştirilir. Her kara liste IP adresi için her iki servis de anahtarlara yeni akışlar eklemek için akış değiştirme mesajları göndermektedir. Testler farklı sayılardaki zararlı IP adres sayılarına göre tekrarlanmış ve sonuçlar Şekil 5.4'te verilmiştir.



Şekil 5.4. Kara listeye alınacak IP adreslerinin ortalama kurulum süresi

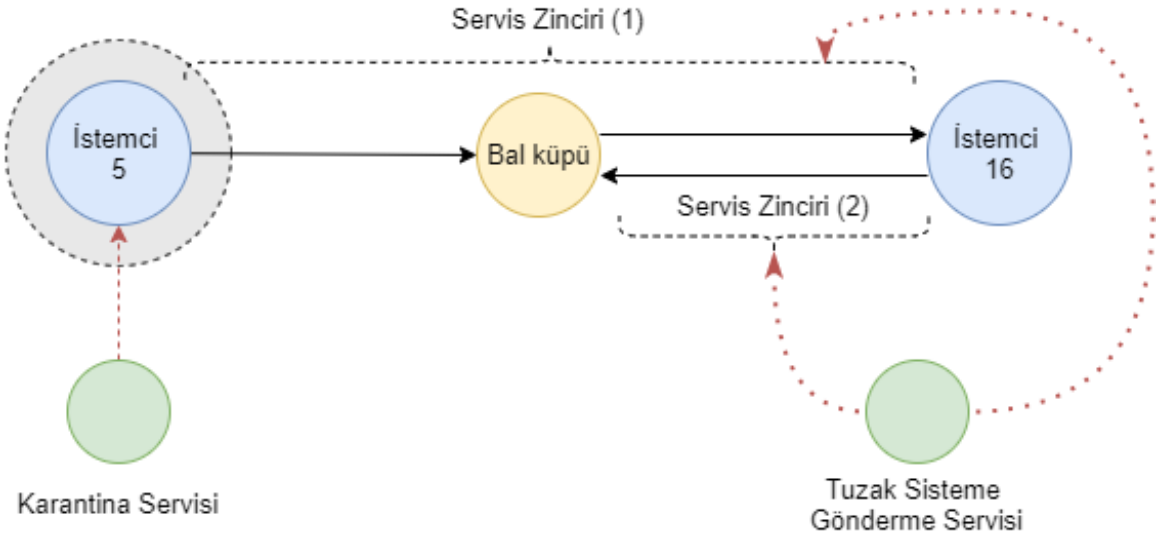
Test sonucunda 8900 kara liste adresinin 14 saniyede ağda yer alan anahtarlara yüklendiği görülmüştür. Bu test senaryosu için, SFZ oluşturmaya gerek olmamaktadır. Bu kapsamda üretilen her mesajın uzunluğunun 146 byte olduğu ölçülmüştür. Servis, anahtarlara 8900 kara liste öğesi yüklerken yaklaşık 14,33 saniye harcadığı ve bu sürede YTA kontrolcüsünün bant genişliğinin 1 Mbit/sn'den daha azını (0,725 Mbit/sn) kullandığı görülmüştür (Eş. 5.1).

$$bant\_genişliği_{kontrolcü} = \frac{öğe\_sayısı \times Byte_{mesaj} \times 8}{süre} = \frac{8900 \times 146 \times 8}{14,33} \approx 0,725 Mbit/sn \quad (5.1)$$

5.3.3. Tuzak sisteme gönderme ve karantina servislerinin değerlendirmesi

Organizasyonlar zararlı trafikleri analiz ederek istihbarat bilgisi elde etmek istediklerinde Şekil 4.8a'da yer alan ağ seviye savunma senaryosu yerine Şekil 4.8b'de yer alan savunma senaryosunu alternatif olarak kullanabilirler. Şekil 4.8b'de yer alan savunma senaryosu için Şekil 5.5'te yer alan test altyapısı oluşturulmuştur. İstemci 16, DDoS saldırganının, zararlı yazılım saldırganının ve indirme saldırganının kontrol ettiği sunucu olarak belirlenmiş ve İstemci 16'nın IP adresi, zararlı IP adresi olarak yapılandırılmıştır. Karantinaya alma ve tuzak sisteme gönderme servislerini değerlendirmek için İstemci 5 üzerinden hping3 [199] aracı ile trafik üretilmiştir.

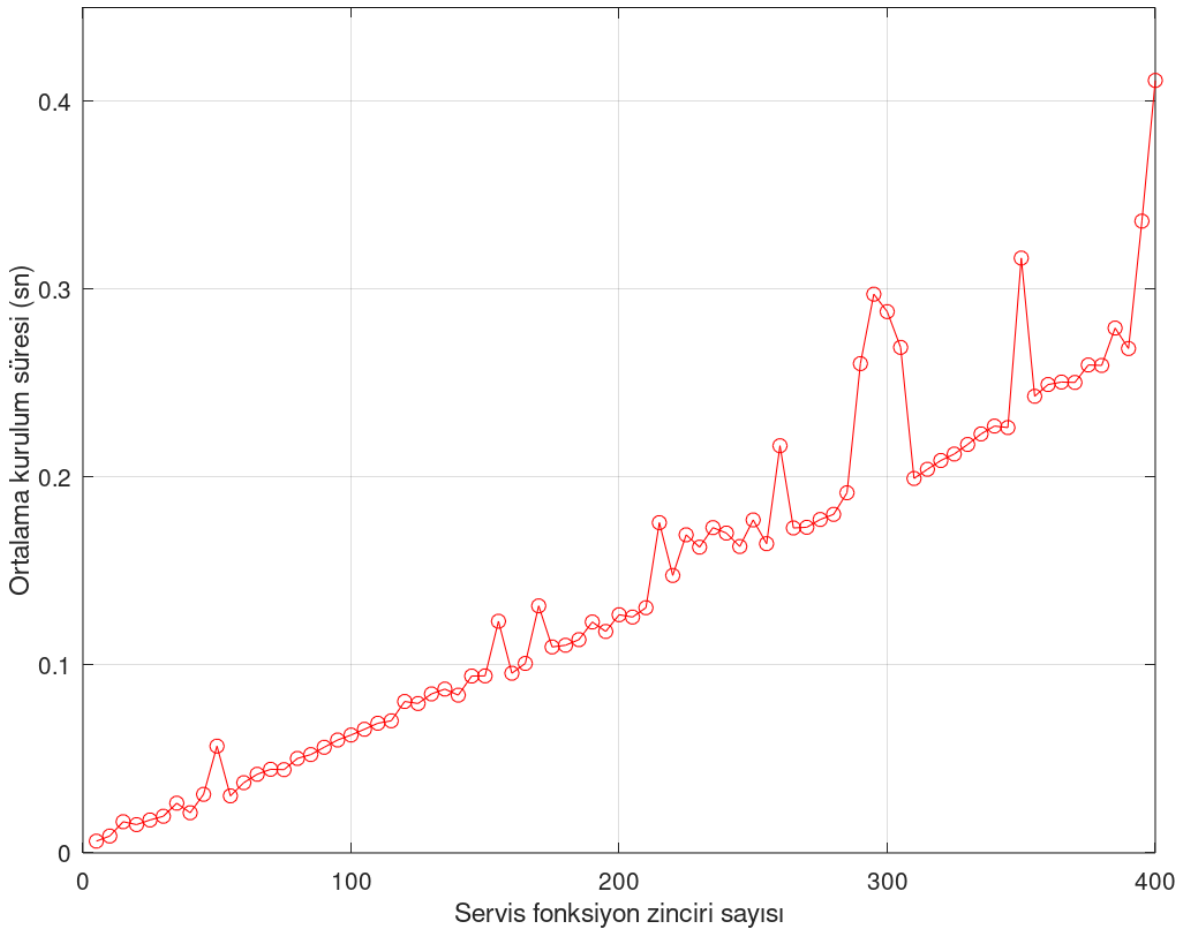
Hazırlanan test ortamında (Şekil 5.5), İstemci 16'nın IP adresi zararlı olarak belirlenmiştir. İstemci 5, bu IP adresine erişmek istediğinde ilgili savunma servisleri devreye girerek İstemci 5'i karantinaya almakta ve iki tane SFZ oluşturmaktadır. İlk zincir, İstemci 5'ten bal küpüne ve bal küpünden İstemci 16'ya olacak şekildedir. İkinci servis ise ters akışı sağlamak için İstemci 16'dan bal küpüne oluşturulmakta ve zararlı IP adresine sahip paketler İstemci 5'e ulaşmaması için düşürülmektedir.



Şekil 5.5. Karantina ve tuzak sisteme gönderme servisleri için test senaryosu

Tez çalışmasında, aynı anda birden çok siber saldırıya karşı tuzak sisteme gönderme ve karantina servislerinin kurulum süreleri test edilmiştir. Servis fonksiyon zincirleri, ilk paket alındığında oluşturulmakta ve oluşturulan zincirde ilk anahtar sınıflandırıcı olarak belirlenmektedir. Akış ile ilgili sonraki tüm paketlerin sınıflandırıcı tarafından aynı rotayı izlemeleri sağlanmaktadır.

DDoS, indirme veya oltalama saldırganları tarafından başlatılan akışlar için kurulan SFZ sayısı ile ortalama kurulum süresi arasındaki ilişki Şekil 5.6'da gösterilmektedir.

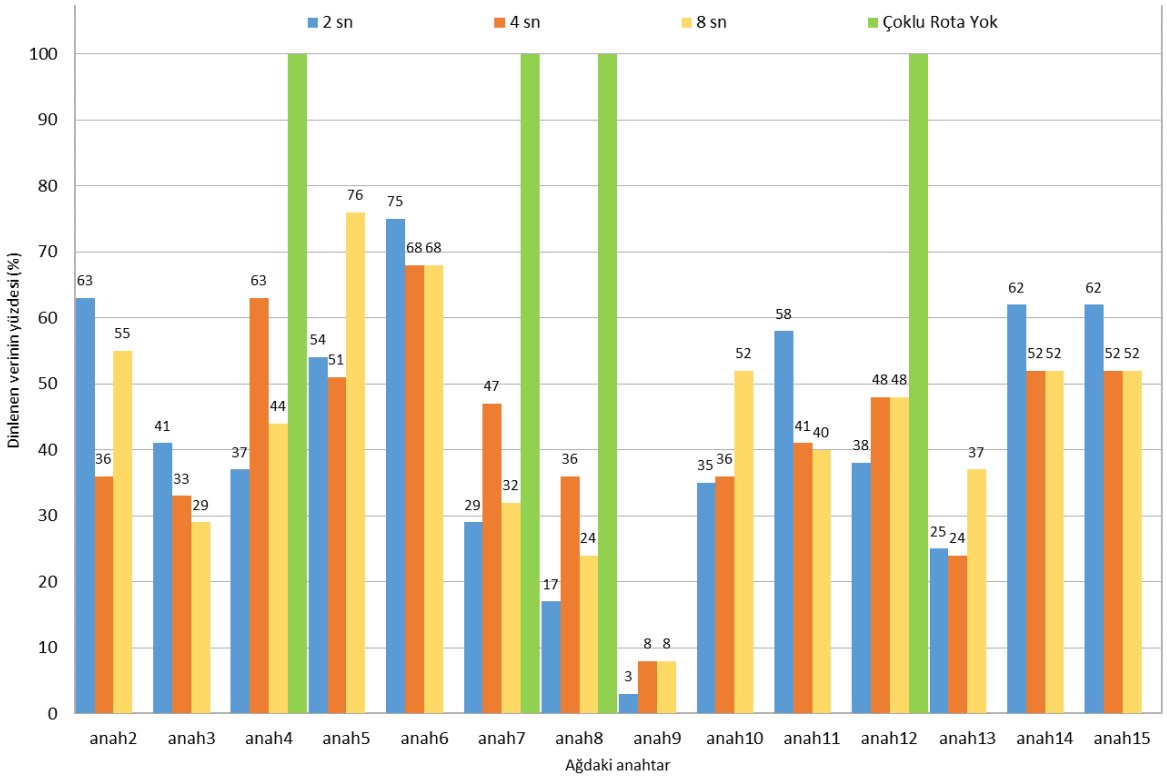


Şekil 5.6. Aynı anda kurulması talep edilen SFZ sayısı ve ortalama kurulum süresi

Servis fonksiyon zincirleri kolay ve hızlı şekilde anahtarlara yüklenebilmektedir. Örnek olarak, aynı anda başlatılan 400 saldırı karşısında fonksiyon zincirinin sadece 0,4 saniyede ağda oluşturulabildiği görülmüştür. Test sonuçları kapsamında, tuzak sisteme gönderme ve karantina servislerinin, servis fonksiyon zincirleri oluşturmak ve karantinaya almak için YTA kontrolcüsünün bant genişliğinin sadece 3,35 Kbit/sn'lik bir kısmını kullandığı ölçülmüştür.

5.3.4. Çoklu rota servisinin değerlendirmesi

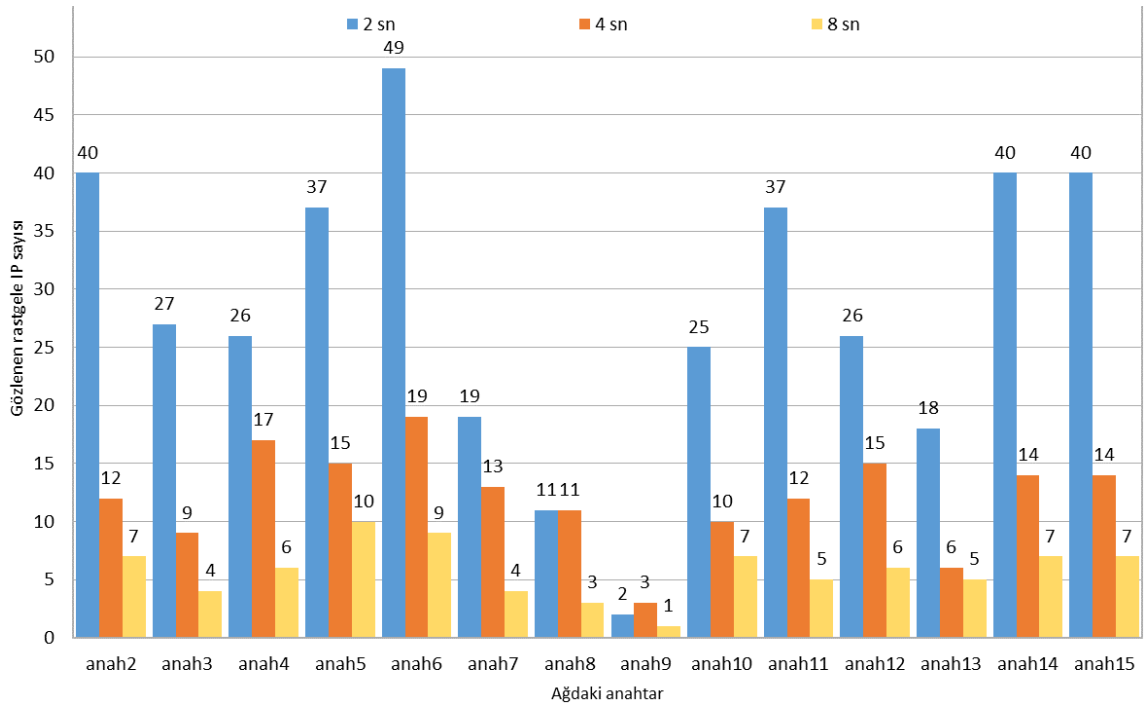
Saldırgan, bir veya daha fazla anahtar üzerinden ağ paketlerini dinleyebilir. Bu nedenle çoklu rota servisi, paketleri ağ kısıtlamalarını dikkate alarak farklı anahtarlara dağıtır. Dinlenerek elde edilen verilerin yüzdesi, dinlenen verileri değerlendirmek için temel bir ölçüttür. Dinlenen verilerin yüzdesi, değişen rotalar ve paketlerdeki rastgele IP adresleri ile değiştirilebilir. Değerlendirme için kullanılan test altyapısında performans test aracı olarak kullanılan iperf3, İstemci 16'da başlatılmış ve test trafiği İstemci 1'den 1 Gbps bant genişliği ile 50 saniye süreyle 20 tekrar ile gerçekleştirilmiştir. Test süresince tüm anahtarlardaki akışlar izlenmiş ve akış istatistikleri toplanmıştır. Toplanan veriler, yükleme zamanı, silme zamanı, anahtar kimliği, rota kimliği, akış kimliği, eşleşme kriterleri, eylemler, süre, paket sayısı bilgilerini içermektedir. Bu istatistikler kullanılarak her anahtardaki paket sayıları hesaplanmıştır. Bu testler, çoklu rota etkinleştirilmediğinde ve farklı periyotlar için tekrarlanmıştır. Ayrıca sadece giriş anahtarında ve her anahtarda rastgele IP adresi değiştirildiği durum için yukarıda belirtilen koşullarda testler tekrarlanmıştır. Testler sonucunda ağ anahtarlarındaki dinlenen veri ve IP değişikliği istatistikleri Şekil 5.7 ve Şekil 5.8'de gösterilmiştir.



Şekil 5.7. Bir akış için anahtarlardan dinlenen verilerin yüzdeleri

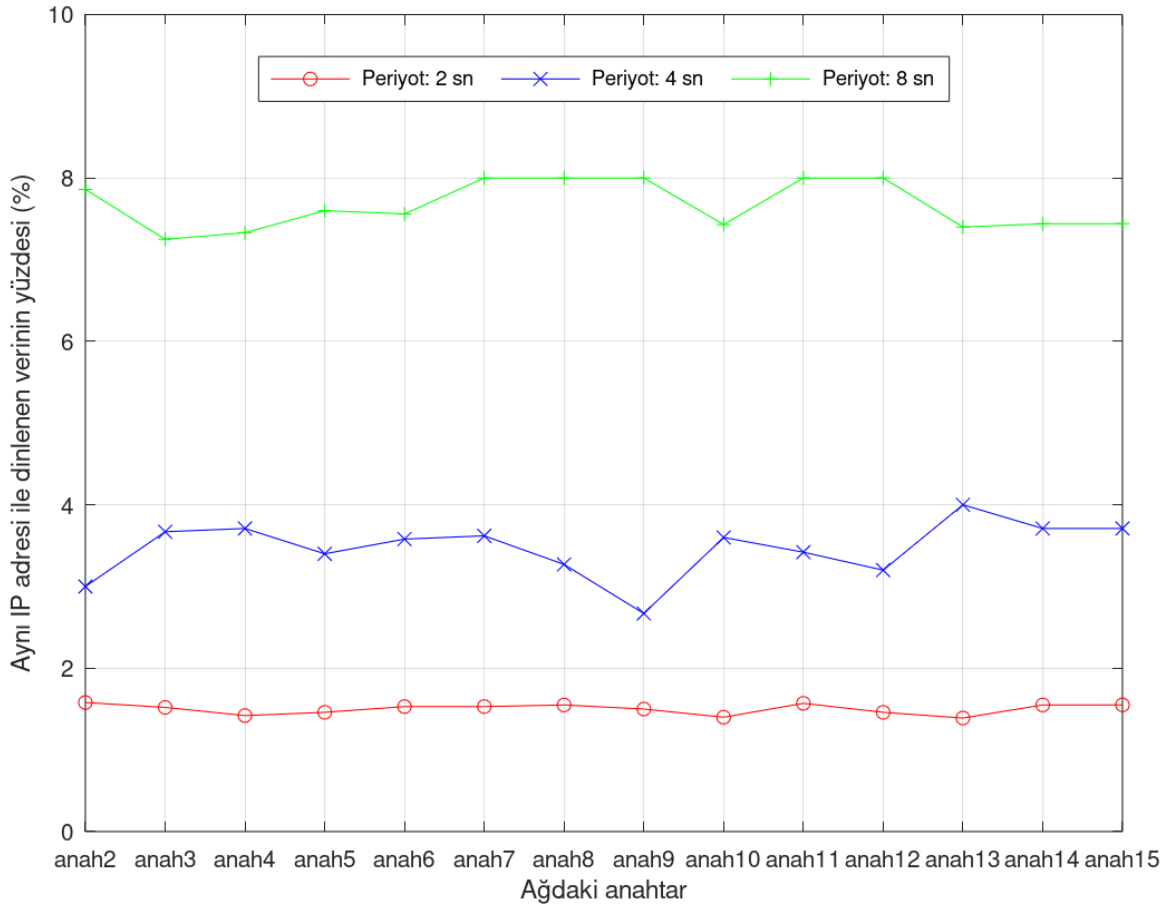
Her anahtardan dinlenen verilerin yüzdeleri, farklı parametrelerle Şekil 5.7'de gösterilmektedir. Çoklu rota servisi devre dışı bırakıldığında, trafik yalnızca bir rotayı takip eder (anahtar 1, 4, 7, 8, 12, 16) ve dinleme saldırganı bu anahtarlardan birini dinlerse tüm veriler yakalanabilir. Çoklu rota servisi etkinleştirildiğinde ve dinleme saldırganı anah4, anah7, anah8, anah12 anahtarlarından birini dinlediğinde, saldırgan dinlediği anahtara göre trafik verisinin %10 ile %70 arasındaki bir oranını yakalayabilmektedir. Çoklu rota periyodu değiştiğinde, anahtarlardan iletilen verilerin yüzdesi önemli ölçüde değişmemekte fakat üretilen rastgele IP adreslerinin sayısı değişmektedir.

Bir akış için her bir anahtarda gözlemlenen rastgele IP sayıları Şekil 5.8'de farklı periyotlar için gösterilmektedir. Periyot azaldıkça ve anahtarlar üretilen rotalarda yer aldıkça rastgele IP sayıları orantılı olarak artmaktadır. Periyot artarken rastgele IP sayısı yine orantılı olarak azalmaktadır. Saldırgan, paketleri kaynak veya hedef IP adreslerini dikkate alarak dinlerse, rastgele IP adresleriyle algılanan verilerin yüzdesi azalır. Örneğin, 4 saniyelik çoklu rota periyodunda, dinlenen verilerin yüzdesi anah4'te %63'tür (Bkz. Şekil 5.4) ancak aynı IP adreslerine sahip verilerin yüzdesi yalnızca %3,7 olmaktadır (Şekil 5.9).

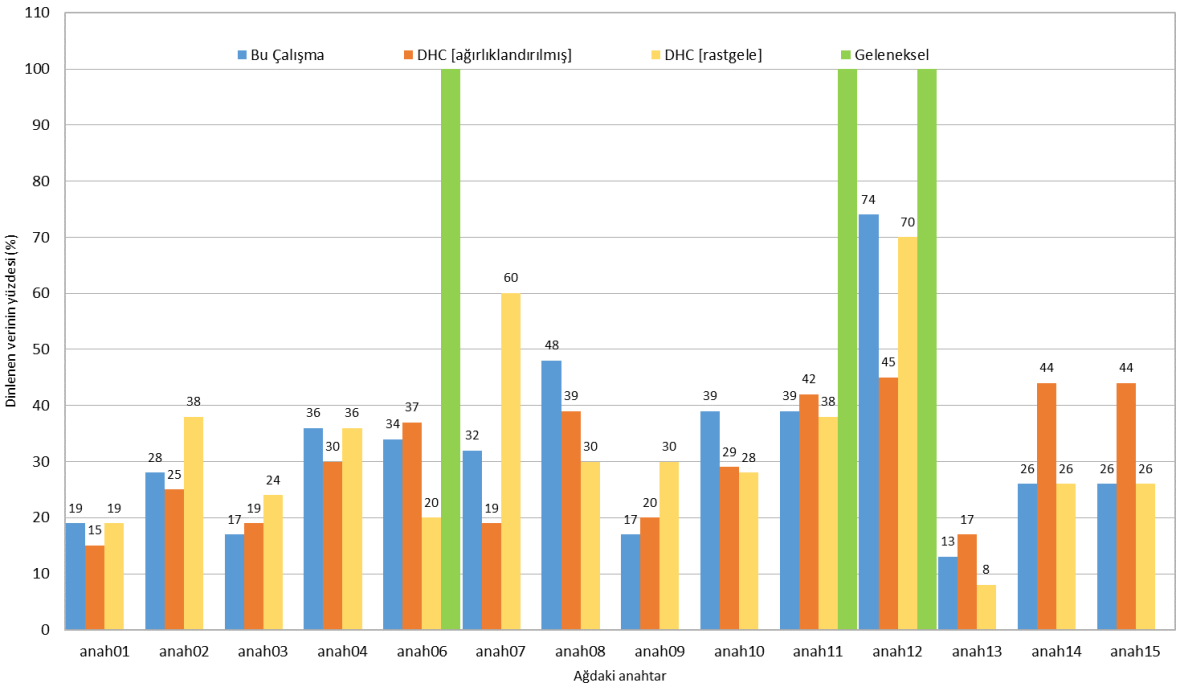


Şekil 5.8. Bir akış için anahtarlardan dinlenen rastgele IP adreslerinin sayıları

Şekil 5.10, bu tez çalışmasında elde edilen sonuçlar ile Zhao ve diğerlerinin [138] gerçekleştirdiği çalışma sonuçlarını göstermektedir.



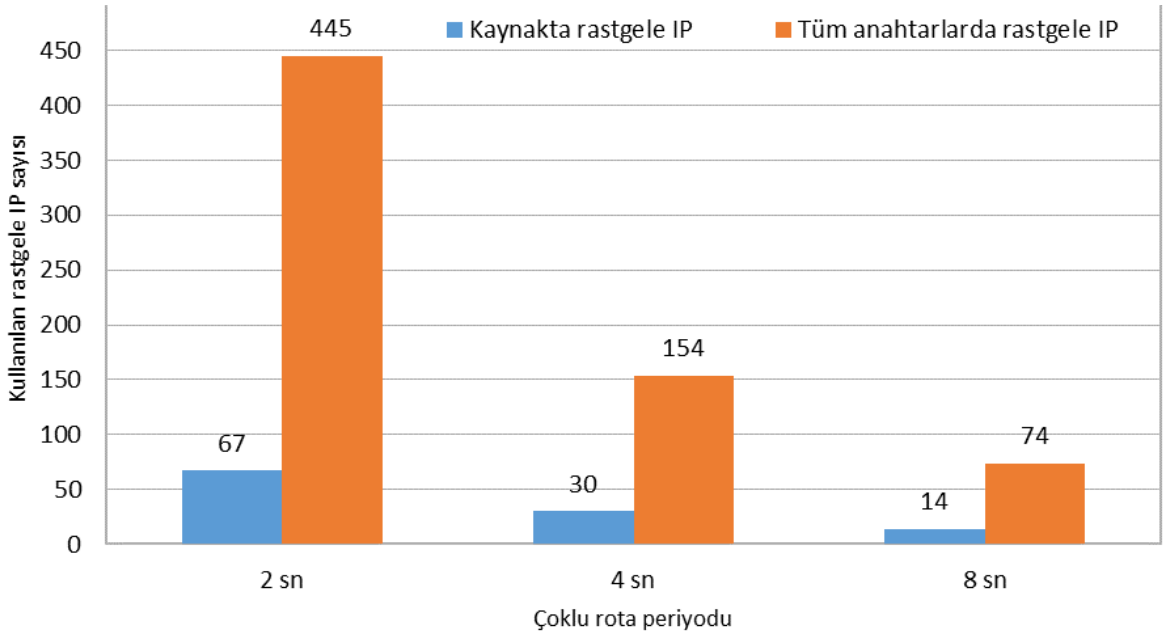
Şekil 5.9. Her anahtarda rastgele IP adresi değiştirildiğinde aynı IP adresine sahip verilerin anahtarlardan dinlenebilme yüzdeleri



Şekil 5.10. Bir akış için anahtarlardan dinlenebilen verilerin yüzdelerinin dağılımları

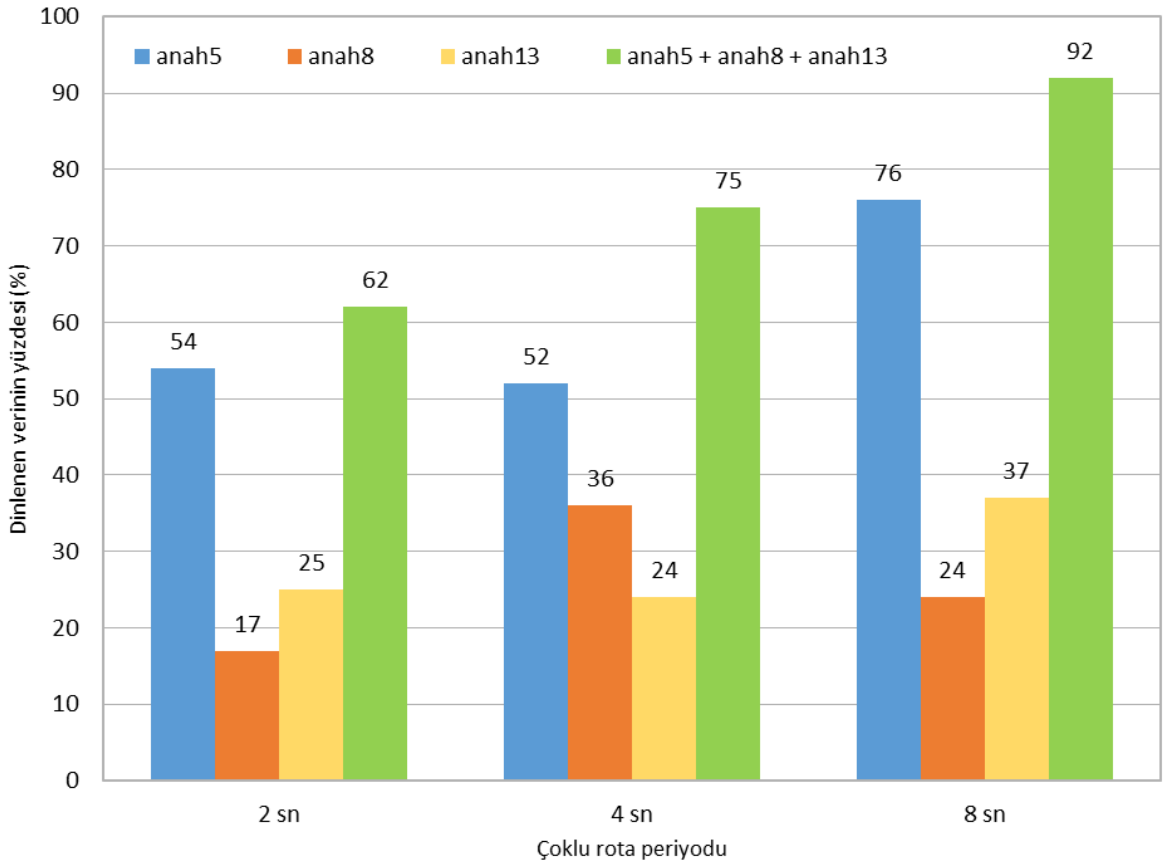
Zhao ve diğerlerinin [138] test ortamı ile benzer şekilde 5 saniye periyotlarla çoklu rota çalıştırılmış ve testler 20 kez tekrarlanarak sonuçlar elde edilmiştir. Çoklu rota kullanılmadığında trafik 6 – 11 – 12 numaralı anahtarlardan geçtiği için dinleme saldırganı bu anahtarlardan tüm verileri dinleyebilmektedir. Zhao ve diğerleri [138], önerdikleri rastgele DHC ve ağırlıklandırılmış DHC yöntemleri ile çoklu rota değiştirerek anahtarlardan dinlenebilen veri yüzdelerini ölçmüştür (Şekil 5.10). Rastgele DHC yöntemi ile en fazla bir anahtardan geçen maksimum veri yüzdesi %70, ağırlıklandırılmış DHC ile dinlenen maksimum veri yüzdesi %45 olarak sunulmuştur. Tez çalışması yapılan ölçümlerde bu değer %74 olarak ölçülmüştür. Elde edilen sonuçlar tez kapsamında geliştirilen servis ile elde edilen sonuçların DHC rastgele yöntemine yakın (%74) olduğu görülmüştür. Bunun sebebi tez çalışmasında rotaların rastgele seçilmesi ve ağırlıklandırma olarak sadece ağ anahtarlarının bant genişliği parametrelerinin kullanılmasıdır.

Tez kapsamında, saldırganların elde edilebilecekleri verilerin azaltılması için servise her anahtarda rastgele IP değiştirme özelliği eklenmiştir. Geliştirilen servis, ilgili çalışmada yer almayan bu özelliği kullandığında bir anahtardan aynı IP ile elde edebileceği maksimum veri yüzdesi %10'un altına düşmektedir (Şekil 5.9). Periyot 2 saniye olarak ayarlandığında ise dinlenen verinin yüzdesinin %1 ile %2 arasına kadar düştüğü görülmüştür. Bu yöntemle saldırganın karşılaşacağı rastgele IP adresi artmaktadır. Şekil 5.11, akış başına üretilen rastgele IP sayılarını göstermektedir.



Şekil 5.11. Çoklu rota ile bir akış için rastgele üretilen IP adreslerinin sayıları

Kaynak anahtarda rastgele IP adresleri oluşturulduğunda, diğer anahtarlar paketleri yeniden yönlendirmek için aynı IP adreslerini kullanmaktadır. Her anahtarda rastgele IP oluşturulacak şekilde yapılandırmaya gidildiğinde bir periyotta her bir rota için anahtar sayısı kadar rastgele IP adresi üretilmektedir. Çoklu rota periyodu azalırsa, rastgele IP adresi sayısı artar. Çoklu rota servisi her anahtarda rastgele IP adresleri oluşturduğunda, kullanılan rotaların uzunluğuna bağlı olarak oluşturulacak rastgele IP sayısı artmaktadır. Test için kullanılan topolojide rotaların uzunluğu ortalama 5 olarak ölçülmüştür. Dolayısıyla, bir akış için rastgele IP adresi sayısı bu faktörle çarpılmaktadır. Her anahtarda rastgele IP, dinlenen verileri de en aza indirmektedir. Şekil 5.12, bir veya daha fazla anahtar dinlendiğinde ve kaynak anahtarda rastgele IP adresi üretildiğinde elde edilen verilerin yüzdelerini göstermektedir.

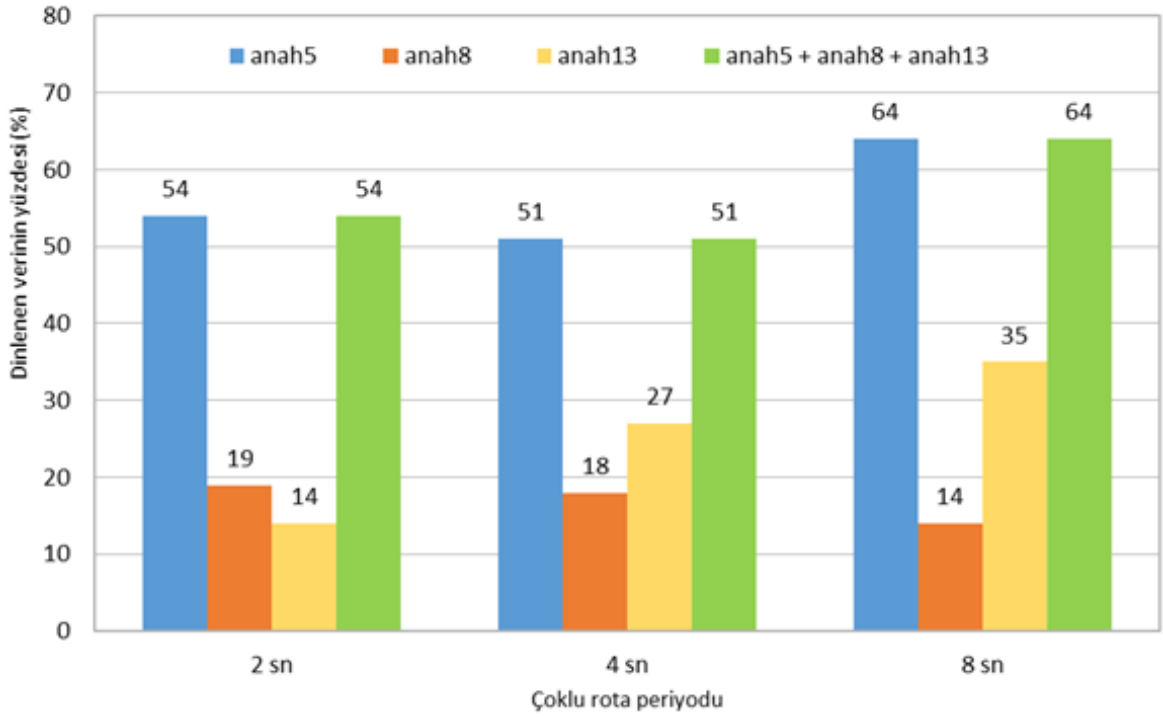


Şekil 5.12. Sadece kaynak anahtarda IP adresi değiştirildiğinde saldırganın ele geçirdiği anahtarlardan aynı IP adresi ile dinleyebildiği verilerin yüzdesi

İki saniyelik periyot ile değişen çoklu rota uygulamalarında, saldırgan anah5, anah8, anah13 anahtarlarından birini dinlerken, akış verilerinin sırasıyla %54, %17 ve %25'ini elde edilebilmektedir. Saldırgan tüm bu anahtarları aynı anda dinlerse verilerin yalnızca %62'sini dinleyebilmektedir. Her anahtarda rastgele IP adresleri üretildiğinde, tüm IP adresleri

anahtarlar için benzersiz olmakta ve diğer anahtarlarda dinlenen IP adresleriyle eşleştirilememektedir.

Şekil 5.13, her anahtarda oluşturulan rastgele IP adresleriyle dinlenen verilerin yüzdesini göstermektedir. Anah5'te dinlenen veriler 2 saniyelik periyotlarla akış verilerinin %54'üdür. Saldırgan aynı anda anah8 ve anah13'ü de dinlerse, aynı IP ile ek veri elde edememektedir ve aynı IP ile dinlenen verilerin yüzdesi değişmemektedir.



Şekil 5.13. Her anahtarda IP adresi değiştirildiğinde saldırganın ele geçirdiği anahtarlardan aynı IP adresi ile dinleyebildiği verilerin yüzdesi

Çoklu rota servisinin performans testi iperf3 aracı kullanılarak yapılmıştır. Çizelge 5.2'de, Germano da Silva ve diğerlerinin [143] elde ettiği servis ağ performans ölçümleri ile bu tez çalışması kapsamında geliştirilen servis için yapılan ölçümler yer almaktadır. Çizelge 5.2'de sonuçları gösterilen test, İstemci 1'den İstemci 16'ya 20 tekrarla 50 saniye boyunca 1 Gbps bant genişliği ile trafik üretilmesi ile gerçekleştirilmiştir. Bu testler, farklı periyotlar ve rastgele IP yapılandırması için tekrarlanmıştır. Çoklu rota servisi aktif olmadığında, paket kaybı değeri %0,45 olarak ölçülmüştür. Çoklu rota servisi etkinleştirildiğinde, maksimum paket kaybı değeri %0,91 olmuştur. Çoklu rota servisinin kullanılması durumunda, sırası değişmiş paketlerin yüzdesi %0,16'dan az olduğu ölçülmüştür.

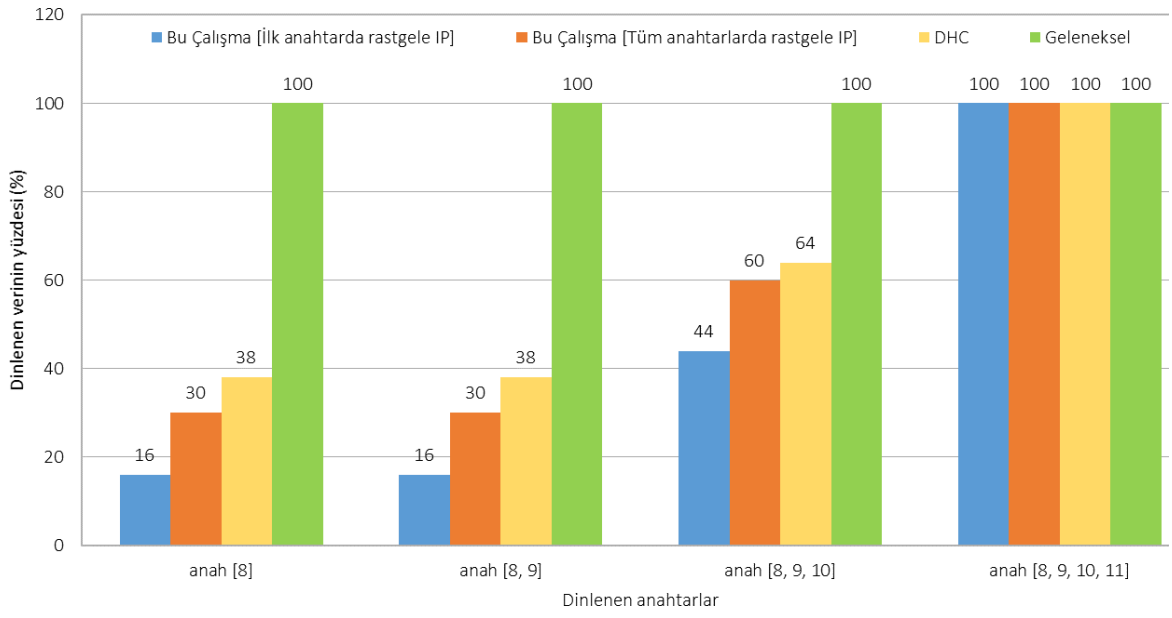
Çizelge 5.2. Çoklu rota servisi için ağ performans ölçümleri

	Germano da Silva ve diğerleri [143]		Kaynak anahtarda rastgele IP üretildiğinde		Her anahtarda rastgele IP üretildiğinde	
	Paket kaybı (%)	Sırası değişmiş paket oranı (%)	Paket kaybı (%)	Sırası değişmiş paket oranı (%)	Paket kaybı (%)	Sırası değişmiş paket oranı (%)
Periyot: 2 sn	-	-	0,51	0,16	0,82	0,02
Periyot: 4 sn	-	-	0,65	0,11	0,91	0,02
Periyot: 5 sn	3,10	-	0,46	0,07	0,76	0,08
Periyot: 8 sn	-	-	0,23	0,02	0,65	0,10
Periyot: 10 sn	2,70	-	0,22	0,03	0,62	0,07
Periyot: 15 sn	1,30	-	-	-	-	-
Periyot: 20 sn	1,10	-	-	-	-	-

Germano da Silva ve diğerleri [143] çalışmalarında 5, 10, 15 ve 20 saniyelik periyotlarda ölçüm yapmış ve paket kayıplarının sırası ile %3,1, %2,7, %1,3 ve %1,1 olduğunu raporlamıştır. Bu tez çalışmasında 2, 4 ve 8 saniye periyotlarla gerçekleştirilen testlerin yanı sıra Germano da Silva ve diğerlerinin [143] çalışması ile karşılaştırabilmek için 5 ve 10 saniye periyotlarla da testler gerçekleştirilmiştir. Periyot düştükçe ağda daha sık yeniden yapılandırma olacağı için paket kaybı oranının artması beklenmektedir. Gerçekleştirilen testlerde en yüksek paket kayıp oranları, 2 ve 4 saniyelik periyotlar için sırasıyla %0,82 ve %0,91 olarak ölçülmüştür. Bu değerler Germano da Silva ve diğerlerinin [143] çalışmasında 20 saniyelik periyotla elde edilen değerden (%1,10) daha küçük olduğu için 15 ve 20 saniyeler için ayrıca karşılaştırma yapılmamıştır. Paket kayıpları, [143] sonuçları ile karşılaştırıldığında tez çalışmasında elde edilen sonuçların çok daha düşük değerlere sahip olduğu görülmüştür.

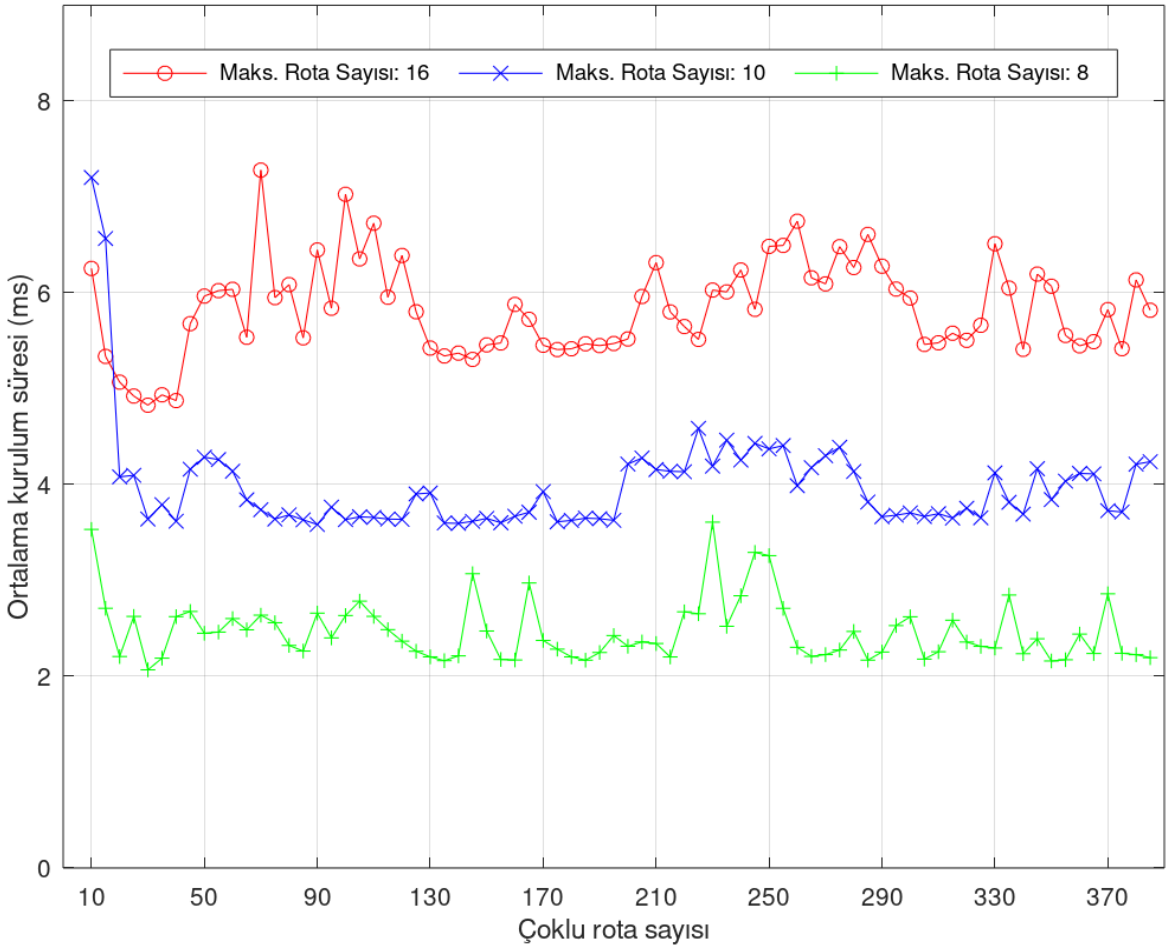
Çizelge 5.2'de verilen sonuçlar incelendiğinde periyot arttıkça paket kayıp yüzdesinin genellikle düştüğü ve her anahtarda rastgele IP adresi üretildiğinde de akıştaki paket kayıplarının arttığı görülmüştür. Bunun sebebinin paketlerin güncellenmesi için anahtarlarda gerçekleşen işlemler sonucu gecikmelerin oluşmasından kaynaklandığı değerlendirilmiştir. Sırası değişmiş paket yüzdeleri incelendiğinde ise %0,02 - %0,16 oranında sıra değişimi tespit edilmiştir. Paketlerin sırasının değişmesindeki temel etkenin rastgele oluşan rotaların uzunluklarının değişebilmesi olduğu değerlendirilmektedir. Bir önceki rotanın uzun bir sonraki rotanın kısa olması durumunda önceki paketlerin daha sonra ulaşması mümkündür. Sırası değişmiş paket oranları paket kayıp oranlarına göre çok daha düşüktür ve kaynak anahtarda rastgele IP üretildiğinde çoklu rota periyodu arttıkça sırası değişmiş paket oranı azalma eğilimi göstermiştir. Her anahtarda rastgele IP üretildiğinde ise sırası değişmiş paket oranlarında periyoda bağlı değişim daha düzensiz olarak gerçekleşmiştir.

Şekil 5.14 incelendiğinde, tüm anahtarlarda rastgele IP adresi değiştirilmesiyle saldırganın elde edebileceği veri oranının sadece ilk anahtarda IP adresi değiştirilmesiyle elde edebileceği veri oranına kıyasla %14 - %16 daha az olduğu görülmektedir. Ayrıca Germano da Silva ve diğerlerinin [143] çalışma sonuçlarıyla bu tez çalışmasında elde edilen sonuçlar kıyaslandığında, saldırganının elde edebileceği verinin %20 - %22 daha az olduğu görülmektedir.



Şekil 5.14. Birden fazla anahtar ele geçirildiğinde elde edilebilen verilerin yüzdeleri

Aynı anda aktif çoklu rotalara göre bir rotanın ortalama yüklenme süresi Şekil 5.15'te verilmiştir.

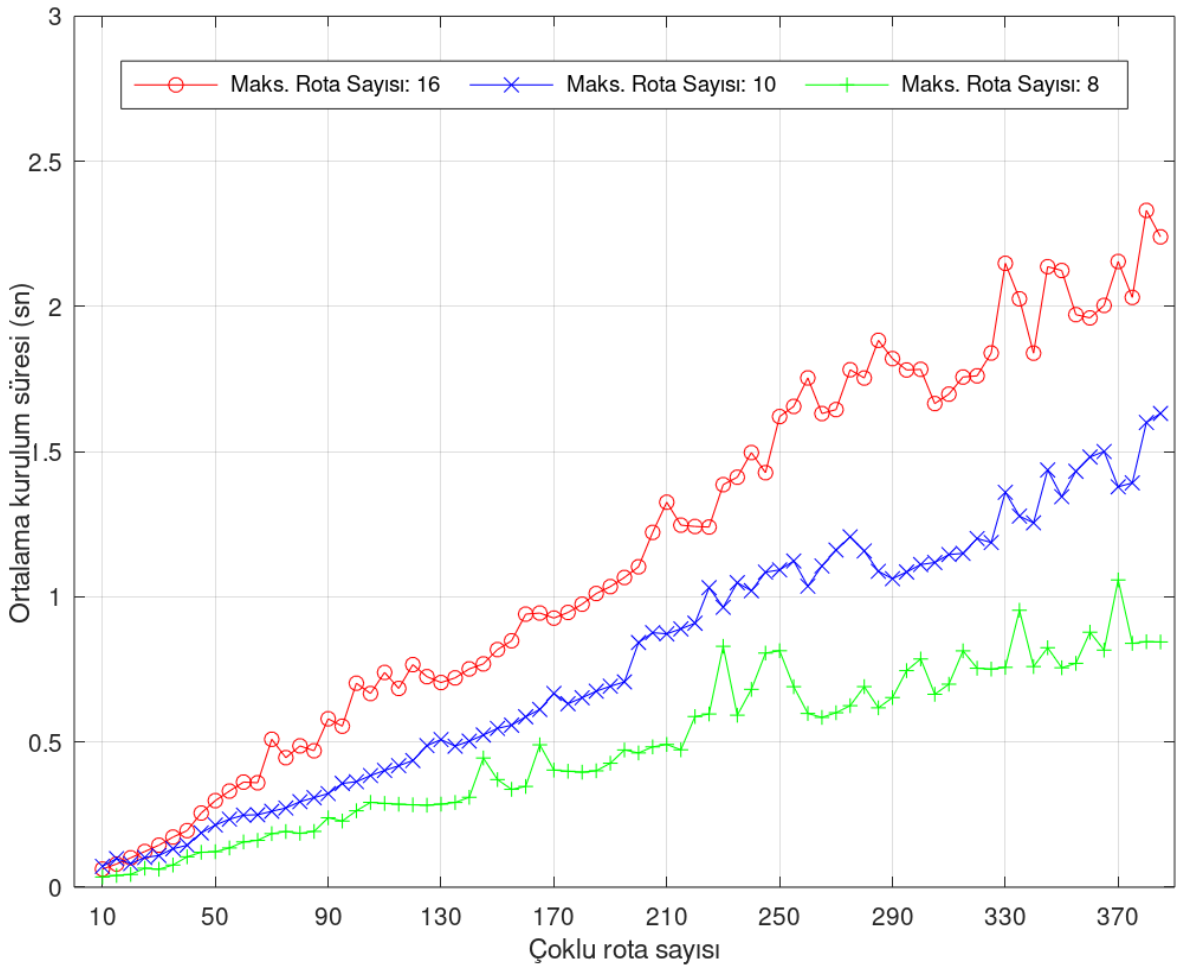


Şekil 5.15. Farklı maksimum rota uzunlukları için aktif çoklu rota sayılarına göre her bir rotanın ortalama kurulum süresi

Her akış için çoklu rota kurulum süresi, alternatif rotaları bulmak, rota maliyetlerini hesaplamak ve rastgele sıralı rotalar oluşturmak için gereklidir. Bu kapsamda gerçekleştirilen testler farklı maksimum rota sayısı parametreleri değiştirilerek 20 tekrarla gerçekleştirilmiştir. Maksimum rota uzunlukları 8, 10 ve 16 için ortalama yükleme süreleri sırasıyla 2,47 ms, 3,99 ms ve 5,84 ms olarak ölçülmüştür (Şekil 5.15). Maksimum rota sayısının sınırlandırılması durumunda rastgele olarak üretilebilecek rota sayısı azalmaktadır. En kritik parametre olan maksimum çoklu rota sayısı azaltıldığında ortalama kurulum süresi de azalmaktadır. Test sonuçlarında bazı veriler için aykırılıklar görülmektedir. Özellikle ondan az olan çoklu rota oluştururken hesaplanan sürelerin bu değerden daha yüksek sayıda çoklu rota oluşturulması için ölçülen ortalama sürelerden yüksek olmasının sebebinin test

tekrarlanırken geliştirilen servisin ilk kez başlatılması kaynaklı gecikmelere, geliştirme altyapısı olarak kullanılan dil yorumlayıcıya ve test ortamına dayandığı değerlendirilmektedir.

Şekil 5.16 incelendiğinde, maksimum rota uzunluğu 8 olarak seçildiği durumda 400 adet çoklu rotanın yüklenmesinin 1 saniyenin altında tamamlandığı görülmektedir. Zhao ve diğerleri [138], sadece örnek rotalar için ölçüm yapmış ve bir rotanın ortalama yükleme süresinin 0,2 milisaniyenin altında olduğunu belirtmiştir. Ancak ilgili çalışmada aynı anda birden fazla çoklu rotanın başlatılması durumunda kurulum ek yükü değerlendirilmemiştir. Tez çalışmasında, aynı anda birden fazla çoklu rotanın başlatılması durumunda alternatif rotaların belirlenmesi, bir rota için gerekli hesaplamaların yapılması ve veri katmanının yapılandırılması için geçen süre ölçülmüştür.



Şekil 5.16. Rota uzunluklarına göre çoklu rotaların ortalama kurulum süresi

Çoklu rota servisi, birden çok rotayı yönetir ve akış tablolarını değiştirmek için anahtarlarla iletişim kurar. YTA kontrolcüsü için çoklu rotanın bant genişliği maliyeti Eş. 5.2'de verilen formül kullanılarak hesaplanabilir. Eş. 5.2'de tanımlanan eşitlik değişkenleri şu şekilde tanımlanmaktadır. Her periyotta, rotadaki her anahtar kadar ($anah_{ort}$) en az bir akış üretilir. Paket kaybını önlemek için birden fazla aktif rota oluşturulabilir ($çr_{aktif}$). Her akışın süresi dolduğunda anahtarlar YTA kontrolcüsüne mesaj gönderir. Ortalama akış değişikliği paketi ve silinmiş akış bildirim mesajı paketinin uzunluğu sırasıyla $Byte_{i,akış}$ ve $Byte_{i,sil}$ ile belirlenir. Akış sayısı, çoklu rota yöneticisi tarafından yönetilen tüm trafik akışının sayısıdır. Anlık 400 trafik akışı, 2 saniye çoklu rota periyodu, anahtarlarda aynı anda 2 yüklü akış kuralı parametreleri ile kullanılan bant genişliği 6,99 Mbit/sn olarak hesaplanmıştır (Eş. 5.3). Akış güncelleme mesaj paketi uzunluğu ($Byte_{i,akış}$) 210 byte, silinmiş akış bildirim mesajı paketinin uzunluğu ($Byte_{i,sil}$) 154 byte olarak seçilmiştir. YTA kontrolcüsünün bant genişliği 1 Gbit/sn olduğu durumda bant genişliğinin yalnızca %0,69'u kullanılmış olmaktadır.

$$bant\_genişliği_{kontrolcü}(MBit/sn) = akış\_sayısı \times 8 \times \frac{\sum_{i=1}^{anah_{ort}} (çr_{aktif} \times [Byte_{i,akış} + Byte_{i,sil}])}{periyot} \tag{5.2}$$

$$bant\_genişliği_{kontrolcü} = 400 \times 8 \times \frac{\sum_{i=1}^{6} (2 \times [210 + 154])}{2} \approx 6,99 Mbit/sn \tag{5.3}$$

5.3.5. Sistemin genel değerlendirmesi

Literatürdeki çalışmaların çoğu STİ verilerini yönetmek için STS temelli çözümler önermektedir. Tez çalışmamızda önerilen modelde STS sadece ağdaki anormallikleri tespit etmek için kullanılmaktadır. Ayrıca bu tez çalışmasında STİ verilerinde soyut hareket tarzının yer alması için yeni bir veri modeli önerilmiştir. STİ verilerinde olması önerilen bu veri modeli kullanılarak savunma servisleri ve ağ sanallaştırma fonksiyonları etkinleştirilebilmiş veya zincir içinde kullanılabilmiştir. Testlerde birbirinden bağımsız olarak geliştirilen savunma servisleri, sistemin modüler yapısına entegre edilerek ağ savunmasında kullanılmıştır. Geliştirilen ağ savunma servisleri, sistemin ara yüzlerini kullanarak entegre olabilmiştir. Buna ek olarak sistem esnek ve genişleyebilir olduğu için farklı ağ savunma servislerinin kolaylıkla sisteme eklenerek devreye alınabileceği değerlendirilmektedir.

Y-S3; anahtarlar kontrolcüye paket gönderdiğinde, STİ verileri güncellendiğinde, STS uyarıları alındığında veya diğer ağlarla iş birliği yapıldığında YTA Siber Savunma Yöneticisi (YTA-SSY) tetiklenerek ağ savunma servislerinin devreye alınmasını sağlayacak şekilde tasarlanmıştır. Tez çalışmasında sistem, STİ verilerinden elde edilen bilgilerle ve anahtarlardan paket gönderilerek gerçekleştirilen testlerle sınanmıştır.

Kara liste servisi, tüm anahtarları kullanarak zararlı IP adreslerini yönetir. Kara liste yönetiminin çok az kurulum ve bant genişliği maliyeti vardır. 8900 kara liste öğesinin tümü, 1 MBit/sn'den daha az bant genişliği kullanılarak 14 saniyede yüklenebilmektedir. Kara liste servisi, zararlı IP adreslerini saatlik veya günlük olarak engellemek için kullanılabilir. Karantina servisi, istemcileri izole eder ve karantina sunucusu ile iletişim kurulabilmesini sağlar. Karantina servisi için zamanlama önemlidir. Örneğin, WannaCry K&K sunucusu veya diğer sunucular ile iletişim kurmaya çalışan istemci hemen karantinaya alınmalıdır. Karantina servisi, istemciyi 5 milisaniyeden daha kısa sürede karantinaya alabilmektedir. Kara liste ve karantina servisleri kullanarak, STİ verilerinde yer alan ve yaygın olarak görülen saldırıların etkisi ağ düzeyinde azaltılabilir.

Y-S3 kullanılarak birden fazla savunma servisi aktive edildiğinde SFZ oluşturularak uygulanabilmektedir. Bu fonksiyon zincirine sanal ağ fonksiyonları dahil edilebilmektedir. Ağ seviyesinde servis fonksiyon zincirleri yapabilmek için NSH protokolü ile OpenFlow protokolünde yer alan grup yapıları kullanılmaktadır. Organizasyonlar, tuzak sistemleri kullanarak zararlı veya şüpheli trafiği analiz edebilirler. Tuzak sisteme gönderme servisi, NSH protokolünü kullanarak geliştirilmiş olan tuzak sistemine zararlı veya şüpheli ağ trafiğini yönlendirir. Savunma servis yalnızca YTA anahtarlarının yeteneklerini kullanmaktadır. Her bir anahtar, aynı anda SFZ için hem sınıflandırıcı ve hem de iletici rolünü üstlenebilmektedir. Yapılan testlerde aynı anda başlatılan 400 akış için kuralların oluşturulması ve anahtarlara yüklenmesi 0,4 saniyenin altında gerçekleşmiştir.

Dinleme saldırıları çoklu rotalar kullanılarak en aza indirilebilmektedir. Çoklu rota servisi, akışları belirli zaman dilimlerinde farklı rotalardan yönlendirebilir. Bu servis, çoklu rotaları düzenler ve akışları değiştirmek için anahtarlara mesaj gönderilmesini sağlar. Çoklu rota süresi azaldığında ve rastgele IP adresleri kullanıldığında dinlenen veriler en aza indirilebilir. Saldırgan aynı anda birden fazla anahtarı dinlese bile her anahtarda farklı rastgele IP adresleri oluşturulduğundan dinlenen veriler IP adresleri ile birleştirilemez. Çoklu rota

servisinin tüm ağa veya alt ağlara uygulanması beklenmemektedir. Seçilen istemciler veya servisler için kullanılması öngörülmektedir. Çoklu rota servisi, 2 saniyelik periyotlarla 400 akışı yönettiği durumda YTA kontrolcüsünün sadece 6,99 MBit/sn'lik bant genişliğini kullanmakta ve aynı IP adresine sahip dinlenen verilerin yüzdesini %4'ün altına indirebilmektedir. Her anahtarda IP adresi değiştirilerek birden fazla anahtar ele geçirildiğinde aynı IP değerine sahip paketlerin oranının artmaması sağlanmıştır. Çoklu rotaların yönetimi sırasında ortalama paket kaybı %1'in altında, sırası değişmiş paket yüzdesi de %0,2'nin altında gerçekleşmiştir. Germano da Silva ve diğerleri [143] yaptıkları testlerde paket kayıplarının %1,1 – %3.1 aralığında olduğunu raporlamıştır. Bu tez çalışmasında elde edilen sonuçlar bu değerlerden oldukça düşüktür.

6. YTA TEMELLİ SİBER SAVUNMANIN GELECEĞİ

Bu tez çalışmasında taranan literatür çalışmalarının sayısı ve çeşitliliği, siber savunma çözümlerini geliştirmek için YTA kullanmanın popüler bir araştırma alanı olduğunu göstermektedir. YTA kullanarak önleme ve etki azaltma yaklaşımları ile ilgili henüz tam olarak cevaplanmamış bazı araştırma soruları aşağıda verilmiştir:

- Yüksek performanslı, ölçeklenebilir, yüksek seviyede kullanılabilirlik sunan, özelleştirilebilir, kurulumu daha kolay, pratik, uyarlanabilen ve güvenli YTA tabanlı savunma çözümleri nasıl geliştirilebilir?
- YTA temelli savunma çözümleri geleneksel ağ cihazlarının da bulunduğu karma ağlarda nasıl uygulanabilir?
- Yazılım tanımlı ağlar, gelecekte kullanılacak yeni teknolojileri nasıl daha güvenli hale getirebilir ve yeni nesil teknolojilerle nasıl daha etkin kullanılabilir?
- YTA temelli savunma yaklaşımları nasıl değerlendirilmeli ve diğer çözümlerle nasıl karşılaştırılmalıdır?

Aşağıdaki alt bölümlerde, yukarıda bahsedilen soru ve sorunlara dayalı öneriler ve araştırma konuları detaylı olarak sunulmuştur.

6.1. Daha Güvenli, Güvenilir ve Ölçeklenebilir YTA Mimarileri

YTA altyapı bileşenlerini korumak ve YTA kullanımından kaynaklanan yeni riskleri yönetmek, hem ağı hem de yazılımı kontrol eden iyi tasarlanmış mimarileri gerektirir. YTA için ağ ve kontrolcü mimarilerinin tasarımında sağlanması gereken üç vazgeçilmez gereksinim; güvenlik, güvenilirlik ve ölçeklenebilirliktir.

Daha ölçeklenebilir ve güvenilir mimariler için bir kontrolcü hiyerarşisi veya birden çok eşgüdümlü kontrolcü kullanılabilir. Bu kontrolcüler, genişletilebilir ağ savunma yaklaşımlarını gerçekleştirmek için birlikte çalışabilir. Uyumlu çalışan çoklu kontrolcülerin sorumluluklarını yöneten çözümler geliştirmek önemli bir araştırma konusudur. İlgili bir çalışmada, Hu ve diğerleri [203], esnekliği ve güvenliği artıran siber taklit savunma (cyber mimic defense) fikrine dayanan Taklit Ağ İşletim Sistemi (Mimic Network Operating System, MNOS) adı verilen bir YTA kontrolcü mimarisi önermiştir. Bir başka anlamda, MNOS birden fazla farklı kontrolcüyü kullanır ve bu kontrolcülerin bir alt kümesinin belirli

bir zamanda birlikte aktif olacak şekilde seçilmesi için YTA kontrolcü mimarisine bir taklit katmanı ekler. Bu mimari, kontrolcüyü ele geçirmek ve verileri değiştirmek için YTA kontrolcüsündeki güvenlik açıklarından yararlanmaya çalışan saldırılara karşı güvenliği artırır. Kontrolcü kümesindeki heterojenlik, belirli bir kontrolcüdeki güvenlik açığının bir saldırı vektörü olma riskini azaltır.

Çok kontrolcülü mimarilerde, kontrolcülerin eylemlerini koordine etmek ve durum tutarlılığını her zaman korumak için yeni bileşenler kullanılması gerekmektedir. Bu yeni bileşenler, yeni güvenlik açıklarına neden olma riskini en aza indirecek şekilde geliştirilmelidir. Ek olarak, kontrolcüler fonksiyonellik açısından birbirlerinin yerine kullanılabilecek ölçüde benzer ancak aynı güvenlik açığının iki farklı kontrolcüde mevcut olmamasını sağlayacak kadar farklı olmalıdır. Bu konudaki bir başka önemli zorluk, verilen iki kontrolcü arasındaki fonksiyonel benzerliği ve yapısal farklılığı değerlendirmek ve bir dizi farklı fakat işlevsel olarak eşdeğer kontrolcü seçmenin en iyi yolunu bulmak için metodolojiler tasarlamaktır.

Dinamik, dağıtık, yüksek oranda kullanılabilir ve ölçeklendirilebilir YTA temelli ağ savunma mimarileri ağ servis fonksiyonlarından faydalanır. Güvenlik fonksiyonlarının yönetimi ve düzenlenmesi popüler bir araştırma alanıdır. 5G, IoT ve Dokunsal İnternet (Tactile Internet) gibi gelişmekte olan alanlarda bu soruna daha fazla odaklanmaya ihtiyaç vardır. Özellikle, SFZ ve NSH [18] gibi protokollerin ağ savunma çözümlerinin performansı üzerindeki potansiyel etkisini analiz etmek için daha fazla çalışmaya ihtiyaç duyulmaktadır.

Literatürdeki çoğu çalışma, önerilen savunma mekanizmalarını değerlendirmek için küçük topolojiler kullanmış veya basit senaryolarla testler gerçekleştirmiştir. Bu durum, önerilen yöntemlerin büyük ölçekli operasyonel ağlarda uygulanabilirliğinin sorgulanmasına yol açmaktadır. Çalışmalarda sunulan yöntemlerin büyük ağlarda ve gerçek ortamlarda yapılmasına ihtiyaç duyulmaktadır. Ayrıca çoğu öneri, tüm ağın bir YTA altyapısından oluştuğunu varsaymaktadır. Gerçekte çoğunlukla yazılım tanımlı ağlar ve geleneksel ağlar birlikte kullanılmaktadır. Önerilerin karma ağları göz önünde bulunduracak şekilde güncellenmesi ve karma ağlarda değerlendirilmesi gerekmektedir.

6.2. YTA Mimarilerinin Diğer Teknolojilerle Entegrasyonu

Ağlarda yazılımlaşma, genişletilmiş saldırı yüzeyi ve saldırılar için potansiyel etki gibi ağ güvenliği için yeni zorluklar ve riskler getirir. YTA ve AFS emelli bulut ve taşıyıcı ağlar için artan risklere karşı yeni proaktif savunma mekanizmaları geliştirilmelidir. Ağ dilimlemesi, 5G ve gelecek nesil telekomünikasyon teknolojileriyle yaygın olarak kullanılacaktır. Ağ dilimlemesi, dilimler arasında güçlü izolasyon gereksinimlerini karşılamak ve saldırı anında performansı garantilemek için kullanılabilecek bir yaklaşımdır. 5G ve ötesinde ağ dilimleme güvenliğinin önümüzdeki birkaç yıl için önemli bir araştırma konusu olması beklenmektedir.

Blok zincir, siber güvenlik içeren çok çeşitli uygulamalara sahip yeni bir teknolojidir. Çeşitli iş alanları için; merkezi olmayan, şeffaf, sağlam, denetlenebilir ve güvenli mimari çözümleri sunar. Araştırmacılar, siber saldırılarla başa çıkmak ve dağıtık güvenlik mekanizmaları gerçekleştirmek için YTA ve blok zincir tabanlı güvenlik mimarileri önermişlerdir. YTA ve blok zinciri bir araya getiren iki ana araştırma alanı vardır: YTA özellikli ağlar için blok zincir tabanlı güvenlik geliştirme ve YTA kullanarak blok zincir tabanlı dağıtık DDoS saldırılarının etkisini azaltma yöntemleri. YTA özellikli ağlar için blok zincir tabanlı güvenlik geliştirmeleri IoT [204–206], 5G [207, 208], yazılım tanımlı optik ağlar (software defined optical networking, SDON) [208] gibi teknolojik alanlarda önerilmiştir. Literatürde blok zincir tabanlı dağıtık DDoS saldırılanın etkisini azaltma alanında, YTA kullanan birkaç yeni çalışma [132, 134, 135] vardır. DDoS saldırılarının etkilerini azaltmak için ne blok zincir ne de YTA tek başına yeterli olamamaktadır. Ancak son çalışmalar, kaynak tüketimini azaltmaya ve saldırı önleme yöntemlerinin performansını iyileştirmeye yardımcı olmak için blok zincirin potansiyeline odaklanmıştır. Ayrıca blok zincir teknolojisi kullanılarak birden fazla etki alanı arasında STİ verilerini paylaşmak ve fidye yazılımı, oltalama gibi diğer siber saldırıları önlemek ve bu saldırıların etkilerini azaltmak için çalışmaların yapılabileceği düşünülmektedir.

IoT, son yıllarda blok zincir uygulamaları için aktif bir alan haline gelmiştir. Ali ve diğerleri [209] blok zincirler için genel uygulama alanlarını özetlemiştir. Pohrmen ve diğerleri [210] de blok zincir tabanlı SDN-IoT mimarileri ve çerçevelerini incelemiştir. Blok zincir tabanlı IoT çözümlerinin doğasında kaynak kısıtlamaları vardır, bu nedenle IoT için optimize edilmiş ve hafif blok zincir tabanlı YTA güvenlik mekanizmaları konusunda araştırma

fırsatları bulunmaktadır. Uygulamayı basitleştirmenin bir yolu, ağ güvenliği çözümleri için gerekli olmayan bazı blok zincir özelliklerini (çift harcamaya karşı güvenlik özellikleri vb.) kullanmamak olabilir. Ayrıca IoT için çoğu blok zincir çözümü akıllı sözleşmeler kullanır, ancak güvenli sözleşmeler yazmak için IoT çözümlerinde güvenlik standartlarının oluşturulması gerekmektedir. Bu nedenle, YTA kullanarak yeni güvenlik açıkları oluşturmayacak şekilde IoT akıllı sözleşmeleri geliştirme konusunda da çalışmalar gerçekleştirilebilir.

Başka bir araştırma konusu da IoT için dağıtık YTA kontrolüdür. Blok zincirler, güvenliği merkezi olmayan ve güvenliği artırmak için YTA kontrol katmanına entegre edilebilir [209]. Yang ve diğerleri [211], 5G'de blok zincir tabanlı güvenilir çok alanlı iş birliği sistemlerinin bir örneğini sunmuştur. Bu örneğe benzer şekilde; IoT ağlarını yönetmek, kaynakları yönetmek ve saldırılara karşı savunmak için blok zincir tabanlı farklı mimarilerin geliştirilmesi konusunda daha çok çalışma yapılabileceği değerlendirilmektedir. 5G'de, blok zincir teknolojisi, ağlarda güvenilir düğüm iş birliği sağlayabilir ve böylece birden fazla ağ etki alanı blok zincirlerini kullanarak güvenli şekilde iletişim kurabilir. Blok zincir ile güvenilir çoklu alan veya güvenilir ağ iletişimi konusunda yeni araştırmaların yapılabileceği öngörülmektedir.

Siber saldırı ve savunma alanında yapay zekâ kullanımı oldukça yaygın hale gelmiştir. Yapay zekâ, siber saldırılarının her aşamasında (keşif, silahlanma, dağıtım, sömürü, yükleme, komuta kontrol ve eylem) kullanılabilmektedir. Yapay zekâ kullanılan saldırıların tespiti ve önlenmesi için yine yapay zekâ kullanan yeni yaklaşımların geliştirilmesi gerekmektedir. Latah ve Toker [212], yapay zekanın YTA ağlarında kullanımı konusunda geniş bir tarama yapmıştır. Yazılım tanımlı ağlarda yapay zeka ve makine öğrenmesinin; iş birliği ile saldırı önleme [213], DDoS saldırı tespiti [214], zararlı yazılım tespiti [215], botnet tespiti [216] gibi alanlarda kullanıldığı görülmektedir. Siber tehditler için bu alanda daha çok çalışma yapılması gerektiği değerlendirilmektedir.

6.3. YTA Temelli Yeni Savunma Yaklaşımları

Literatürde sınırlı sayıda YTA temelli savunma yaklaşımı olduğu görülmüştür. Modern siber tehditlerle başa çıkmak için hem yeni yaklaşımlar sunan hem de birden çok yaklaşımı bir arada kullanan modüler savunma sistemleri gereklidir.

Araştırmacılar, yaygın ağ protokollerinin (ARP, DHCP, DNS vb.) ve IPv6 protokolü için kullanılan NDP gibi daha yeni protokollerin güvenliğini artırmak için gelişmiş yöntemler geliştirmelidir. Savunma mekanizmaları uyarlanabilir olmalı ve tehdit kategorilerini, saldırgan yeteneklerini ve kullanılabilir ağ kaynaklarını göz önünde bulundurmalıdır. Örneğin, kötücül ağ adli bilişim yaklaşımı, güvenlik araştırmacılarına yeni zorluklar getirmektedir. YTA mimarisinin açıklığından yararlanan saldırı yöntemlerine dirençli ağlar tasarlamak gelecek çalışmalar için diğer bir araştırma konusudur.

Yazılım tanımlı ağların en temel özelliklerinden biri merkezi olarak ağın yönetilebilmesidir. Bu özellik kullanılarak ağ savunmasının belirli bir cihaz veya ağ fonksiyonu ile karşılamak yerine, bütüncül ve iş birliğine dayalı savunma yöntemlerinin geliştirilmesi gerekmektedir. Bu kapsamda saldırılar karşısında bir ağdaki tüm YTA bileşenlerinin savunmaya katılması için yöntemler geliştirilebileceği gibi birden çok ağın iş birliği içinde siber tehditlere karşı mücadele edebileceği yenilikçi yaklaşımlar konusunda araştırmaların yapılabileceği değerlendirilmektedir.

YTA özellikli ağları STİ hizmetleriyle entegre etmek, güvenlik için önemli fırsatlar sunmaktadır. STİ verileri siber tehditler hakkında tehdit türü, göstergeler, tehdit aktörü, taktikler, prosedürler, karşı önlemler gibi bilgileri içerebilmektedir. Bu bilgileri alan organizasyonlar, bilgileri yorumlamak ve savunma mekanizmalarına dahil edilmesini sağlamak için güvenlik ekiplerini görevlendirebilir. Fakat bu süreç zamanında harekete geçebilecek kadar çevik olmayabilir. Bunun yerine, STİ kaynaklarından toplanan veriler, siber tehditlere daha hızlı yanıt vermek için bu tez çalışmasında önerildiği gibi yazılım tanımlı ağlara otomatik olarak yönlendirme kuralları oluşturmak için kullanılabilir. Her STİ verisi bir YTA anahtarında bir akış tablosu girişine kolayca dönüştürülemeyebilir. Belirli bir istihbarat bilgisinin bir akış kuralı oluşturularak kullanıp kullanamayacağını belirlemek için bir ön işleme aşaması olması gerekir. Bu amaçla daha güçlü ve daha verimli bir ağ savunma sistemi için yapay zeka veya makine öğrenmesi yoluyla STİ verilerinin sınıflandırılması, zenginleştirilmesi ve önceliklendirilmesi sağlanabilir. Ayrıca sosyal medya, internet sayfaları, derin web vb. yapısal olmayan kaynaklardan istihbarat bilgisi veya siber tehditlere karşı önlemler oluşturulabilir.

Ağ savunması için mevcut ve yeni geliştirilen çözümlerin performanslı şekilde sanal ağ fonksiyonları olarak kullanılabilmesi gerekmektedir. Mevcut ağ fonksiyonlarının çoğu SFZ içerisinde doğrudan kullanılamamaktadır. SFZ yapılarını ve NSH protokolünü destekleyen daha çok sanal ağ güvenlik fonksiyonuna ihtiyaç vardır. Sanal güvenlik fonksiyonları arasında paylaşılan verilerin nasıl yorumlanacağı konusunun da önemli bir araştırma konusu olabileceği değerlendirilmektedir.

Siber saldırılara karşı savunma yaparken ağ performansının artırılması için saldırı paketlerinin YTA kontrolcüsüne gelmeden analiz edilebilmesi gerekmektedir. Bunun için savunmanın kontrol katmanına gelmeden veri katmanında yapılabilmesi performansı çok fazla artırmaktadır [217]. Literatürdeki bazı yeni çalışmalar [136, 137], ağ savunmasını veri katmanında yaparak kontrolcüye olan bağımlılığın ortadan kaldırılması yönünde adımlar atmış ve ağ performansının iyileştirilebileceğini göstermiştir. Ayrıca bu alanda veri katmanında programlama yapılabilmesini sağlayan P4 [218, 219] ve Domino [220] gibi programlama dilleri önerilmiştir. Veri katmanında savunma yaklaşımları konusundaki çalışmaların yeni olduğu görülmüş olup bu konuda daha çok araştırma yapılmasına ihtiyaç duyulacağı değerlendirilmektedir.

Siber saldırıların tespiti ve analizi için bal küpü gibi çeşitli aldatma yaklaşımları kullanılmaktadır. YTA altyapısı ile entegre, akıllı ve yüksek etkileşimli tuzak sistemlerin geliştirilmesi alanında yapılacak araştırmaların, siber saldırıların tespiti ve önlenmesi için çok önemli olacağı değerlendirilmektedir.

Özellikle karma ağlarda akıllı çoklu rotalar kullanılarak YTA tabanlı saldırıları aldatma ve önleme konusunda daha fazla çalışmaya ihtiyaç vardır. Önerilecek çoklu rota çözümleri, veri katmanı öğelerinin yetenekleri arasındaki farklardan kaynaklanan sorunların üstesinden gelebilmelidir. Ayrıca tehdit kategorisine dayalı ağ servis fonksiyonu zincirleri oluşturularak ve bu zincirleri etkin bir şekilde uygulayarak çeşitli siber tehditlere karşı savunma yapabilecek örnek uygulamalar geliştirilmelidir. Her bir saldırı türüne karşı koymak için farklı YTA temelli savunma ilkelerinin nasıl kullanılabileceği konusunda çeşitli önerilere ve uygulamalara ihtiyaç duyulmaktadır.

6.4. YTA Temelli Siber Savunma Çözümlerini Değerlendirme Kriterleri ve Yöntemleri

Tez çalışması süresince YTA temelli savunma yöntemlerini değerlendirmek için farklı yaklaşımların kullanıldığı gözlemlenmiştir. Örneğin, farklı çalışmalarda kullanılan test topolojileri; yalnızca birkaç istemciye sahip test ağları, Mininet, kampüs ağları ve büyük ölçekli test yatakları olarak sıralanabilir. Bunun yanı sıra bazı çalışmalar sadece tek veya çok az sayıda saldırganı düşünürken, bazıları da önerilerinin koordineli, dağıtık saldırılara karşı davranışını incelemiştir. Son olarak, farklı çalışmaların değerlendirme metriklerinde önemli farklılıklar vardır. Örneğin, DoS'a karşı çözümler sunan bazı makaleler birincil metrikleri olarak işlemci ve akış tablosu alanı gibi kaynakların kullanımına odaklanırken, diğerleri uçtan uca veri aktarım hızı / miktarı ve gidiş-dönüş süresi gibi metrikleri kullanmıştır. Bu farklılıklar, literatürdeki çeşitli çözümlerin karşılaştırmalı olarak değerlendirilebilmesini zorlaştırmaktadır.

HHS alanında, Connell ve diğerleri [221] HHS çözümlerini değerlendirmek için analitik bir model önermiştir. Önerilen model, belirli bir HHS sistemi için yanıt süresi, kaynak kullanılabilirliği, saldırgan için başarı şansı vb. metriklerini içerir. Ayrıca yeniden yapılandırma hızı, istek varış hızı, bir kaynağı yeniden yapılandırmak için harcanan zaman ve maksimum eş zamanlı yeniden yapılandırma sayısı gibi çeşitli parametreleri değiştirerek güvenlik ve performans arasındaki dengeyi keşfetmeye olanak tanır. Araştırmacılar, performans, ölçeklenebilirlik, kullanılabilirlik, uygulanabilirlik ve güvenlik temelinde tüm YTA savunma yaklaşımlarını değerlendirmek için benzer metodolojilerden ve kriterlerden faydalanabilir.

Ağ trafiği test verilerinin miktarı, kalitesi ve çeşitliliği de önerilen savunma yöntemlerini değerlendirmek için önemlidir. Bu nedenle, gerçekçi YTA temelli ağ topolojileri üzerinde üretilen ve toplanan trafik verilerinin farklı saldırı türleri için çeşitli veri kümelerinde derlenmesi ve araştırmacıların kullanımına sunulması gerekir. Bu veri ihtiyacı, özellikle yazılım tanımlı ağlarda saldırı algılama ve yanıtı için makine öğrenimi tabanlı çözümler geliştirmek ve değerlendirmek için önemlidir.

Siber tehditleri önlemek için yapılan çalışmalarda kullanılabilecek STİ test verilerinin sayısı ve çeşitliliği de azdır. Ayrıca mevcut STİ verilerinin kalitesi de düşüktür. Doğal dil işleme, makine öğrenmesi ve yapay zekâ teknolojileri ve yöntemleri kullanılarak yapısal veya

yapısal olmayan STİ verilerinden, tehditlere karşı uygulanabilecek hareket tarzlarını, tehditlerin muhtemel hedeflerini ve STİ verisi güvenilirlik bilgilerini otomatik olarak çıkartılabilmesi sağlayacak yeni araştırmalar yapılabileceği değerlendirilmektedir. Bu çalışmalarda elde edilen bilgilerin güvenilirliğini artırmak için açıklanabilir yapay zekâ (Explainable Artificial Intelligence, XAI) yaklaşımlarının kullanılabileceği değerlendirilmektedir.

7. SONUÇ VE ÖNERİLER

Ağların dinamik olarak yönetilmesini sağlayabilecek yeni ağ teknolojileri arasında YTA ve AFS teknolojileri yer almaktadır. Günümüzde siber tehditleri anlamak ve ağları güvenli hale getirmek için siber tehdit istihbarat verilerini kullanmak, bilinen ve sıfırıncı gün saldırılarına karşı etkili savunma için kritik hale gelmiştir. Bu tez kapsamında, ağ savunmasını otomatik olarak yapılandırabilecek STİ destekli ve YTA temelli bir siber savunma sistemini literatürde ilk kez geliştirme amacı doğrultusunda aşağıdaki araştırma sorularına cevap aranmıştır:

1. Modüler, genişletilebilir, iş birliği içinde çalışabilen ağ savunma sistemi, YTA mimarisi çerçevesinde nasıl tasarlanmalıdır?
2. Yazılım tanımlı ağlarda, saldırıları önlemek için ağ seviyesinde ihtiyaç duyulacak temel savunma servisleri nelerdir?
3. STİ verilerini işleyerek siber saldırılara karşı otomatik ağ seviyesi savunma servisleri nasıl oluşturulabilir?
4. Farklı saldırı türlerine karşı kullanılabilecek savunma servisleri ve servis fonksiyon zincirleri nasıl belirlenebilir?
5. Ağ seviyesi savunma servisleri ile sanal ağ fonksiyonlarının orkestrasyonu nasıl sağlanabilir?

Tez çalışmasında cevap aranan araştırma soruları kapsamında; literatür taraması yapılmış, teknolojiler araştırılmış, sistem tasarlanarak yazılım mimarisi oluşturulmuş, önerilen sistem için prototip geliştirilmiş ve testler gerçekleştirilerek etkinliği değerlendirilmiştir. Her bir araştırma sorusu için yapılan çalışmalar ve çalışmaların sonuçları sonraki paragraflarda sırası ile açıklanmıştır.

Tezin birinci araştırma sorusu kapsamında; YTA temelli siber savunma sistemi (Y-S3) tasarlanmış, geliştirilmiş ve değerlendirilmiştir. Y-S3'ün yazılım mimarisinin modüler olabilmesini sağlamak için sistem bileşenleri belirlenmiş ve bu bileşenlerin görevleri ayrıştırılmıştır. Belirlenen bileşenler servis odaklı mimari yaklaşımı kullanılarak geliştirilmiştir. Y-S3, ağ savunma servislerinin dinamik olarak yönetebilmesini sağlayacak şekilde tasarlanmış, yeni ağ savunma servislerinin eklenebilmesi ve Y-S3'ün genişletilebilir olması sağlanmıştır. Ayrıca Y-S3'ün birden fazla ağa kurulması durumunda sistemlerin

birbirleri arasında yapabilecekleri iş birlikleri belirlenmiştir. Siber tehdit verisi paylaşımı yapılabilmesini ve siber saldırılarda ortak hareket edilebilmesini sağlayacak bir bileşen, Y-S3'e eklenmiştir. Eklenen bileşen diğer ağlardaki bileşenlerle iletişim kurarak ağ savunma servislerinin başlatılması talebi iletebilmekte, STİ verisi alabilmekte veya paylaşabilmektedir.

Tezin ikinci araştırma sorusu için STİ verilerine göre tehdit sınıflandırması yapılmıştır. Tez çalışması kapsamında 5 tehdit kategorisi (tarama, hizmet engelleme, kimlik sahtekârlığı, dinleme saldırıları ile zararlı yazılım, sosyal mühendislik ve web uygulama saldırıları) belirlenmiştir. Literatürdeki çalışmalardan bu tehditlere karşı ağ seviyesi savunma yaklaşımları çıkartılmış ve bu yaklaşımların YTA ile nasıl uygulanabileceğini belirlemek için detaylı analiz çalışması yürütülmüştür. Bu çalışma ile literatürdeki YTA temelli siber savunma çözümlerinin taksonomisi oluşturulmuş ve bu taksonomi kullanılarak; tehdit kategorileri, savunma stratejileri, savunma türleri ve savunma yaklaşımları detaylandırılmıştır. İncelenen araştırmalarda kullanılan geliştirme ve test altyapıları, YTA kontrolcüleri, YTA anahtarları, test ortamları, çözümün kurulum yapıldığı yerler ile ilgili analizler gerçekleştirilmiştir. Bu çalışma sonucunda, tehdit kategorileri için YTA temelli savunma çözümü olarak 14 savunma yaklaşımı tespit edilmiştir. Bu savunma yaklaşımları tezin ilerleyen aşamalarında temel ağ seviyesi savunma servisleri olarak değerlendirilmiştir. Bu çalışmada elde edilen bilgilerin, araştırmacıların siber tehdit odaklı YTA temelli siber güvenlik savunma çözümlerini değerlendirebilmelerini kolaylaştıracağı ve gelecek araştırma konu önerilerinin bu alanda çalışma yapacak araştırmacılara yol göstereceği değerlendirilmektedir.

Tezin üçüncü araştırma sorusu kapsamında; ağ savunma servislerinin yönetimin nasıl olması gerektiği ve ağ servisleri ile etkileşimin hangi ara yüzlerle olacağı belirlenmiştir. Y-S3'e eklenecek savunma servislerinin, kontrol katmanı güney ara yüzünde yer alan her bir fonksiyonu kullanabilecek esneklikte, zamana bağlı veya belirli olaylara göre tetiklenebilecek şekilde oluşturulabilmesi gerektiği belirlenmiş ve bu gereksinimleri karşılayacak şekilde servislerin geliştirilebilmesini destekleyen sistem tasarlanmıştır. Tez çalışmasında belirlenen ağ savunma servisleri, gerekli ara yüzleri gerçekleştirecek ve diğer bileşenlerin ara yüzlerini kullanacak şekilde geliştirilmiştir. Bu ara yüzler kullanılarak servislerin başlatılabilmesi ve servislere gelen paket analiz talebi gönderilmesi sağlanabilmektedir. Ayrıca bu servislerin veri katmanını yapılandırmaları, gerçekleşen

olayları tetikleyebilmeleri ve oluşan olayları takip edebilmelerini sağlayacak sistem ara yüzleri de belirlenerek sistem geliştirilmiştir.

Tezin dördüncü araştırma sorusu kapsamında; farklı saldırı türlerine karşı kullanılabilecek savunma servisleri ve servis zincirlerinin nasıl belirlenebileceği konusunda çalışmalar yürütülmüştür. STİ verileri; siber tehditler için olası hedef varlık türlerini ve savunma yöntemlerini içermesi durumunda ağ seviyesindeki savunma süreci daha etkili hale gelebilmektedir. Tez çalışmasında, STİ verisi ile paylaşılabilecek ve ağ seviyesinde savunmaları otomatikleştirilebilmesini sağlayan bilgileri içeren bir veri modeli önerisi oluşturulmuştur. Bu veri modeli, en çok kullanılan STİ veri formatlarına entegre edilebilecek şekildedir. Sonraki adımda STİ verilerini, kurumsal BT varlıkları, ağ topolojileri, ağda uygulanan güvenlik politikaları ve mevcut savunma servisleri göz önünde bulundurularak otomatik ağ seviyesi savunma servisleri oluşturma adımları belirlenmiş ve Y-S3'e entegre edilmiştir. STİ verilerinde yer alan en yaygın saldırgan modelleri; DDoS saldırganı, zararlı yazılım kontrolcüsü, oltalama saldırganı ve dinleme saldırganı modelleridir. Belirlenen bu saldırgan modellerine dayalı üç siber saldırı senaryosu oluşturulmuştur. Tez çalışması kapsamında kara liste, karantina, tuzak sisteme gönderme ve çoklu rota savunma servisleri geliştirilmiş ve belirlenen saldırı senaryoları ile değerlendirilmiştir.

Tezin beşinci araştırma sorusu kapsamında; YTA ağlarındaki savunma servislerini ve sanal ağ fonksiyonlarını kullanarak servis fonksiyonu zincirleri oluşturulmasını sağlayan bir yöntem önerilmiştir. Bu yöntemde SFZ'ler, tehditlere göre oluşturulmakta, OpenFlow ve NSH protokolleri kullanılarak anahtarlara yüklenmektedir. Zararlı veya şüpheli paketin tespit edildiği anahtarda, pakete otomatik olarak NSH başlığı eklenmekte ve son anahtarda da bu başlık paketten çıkartılmaktadır. SFZ için ağdaki herhangi bir anahtar hem sınıflandırıcı hem de yönlendirici olarak çalışabilmektedir. Birden fazla servis aynı anda çalışması gerektiğinde OpenFlow protokolü grup yapılarından faydalanılmaktadır. Bu yöntemin ağlarda kullanılması ile şüpheli ve zararlı trafik, ağdan izole edilerek iletilebilmekte ve ağda bulunan sanal ağ fonksiyonlarına yönlendirilebilmektedir.

Yürütülen bu tez çalışması sonucunda özetle;

- STİ verisine dayalı YTA tabanlı otomatik savunma yaklaşımının modern ağları saldırılardan korumada etkili olduğu,

- Karantina ve kara liste ağ savunma servislerinin indirme, DDoS, oltalama ve zararlı yazılım saldırılarına karşı birlikte daha etkin kullanılabildiği,
- Tuzak sisteme gönderme ağ savunma servisi kullanılarak zararlı / şüpheli trafiğin bal küplerine yönlendirilmesi için gerekli ağ yapılandırmasının dinamik ve etkin şekilde gerçekleştirildiği,
- Çoklu rota ağ savunma servisinin dinleme saldırılarına karşı etkin şekilde kullanılabildiği, bu servis kullanılarak her anahtarda IP adresi değiştirilmesi durumunda saldırganların birden fazla anahtarı dinleyerek elde edebilecekleri bilginin azaltılabildiği,
- Ağ savunma servislerinin YTA kontrolcüsünün sahip olduğu bant genişliğinin düşük bir oranını kullandığı,
- Tez kapsamında önerilen ağ seviyesi savunma veri modelinin, organizasyonların siber tehdit bilgisi paylaşımında pratik ve hızlı ağ savunması açısından katkı sağlayabildiği görülmüştür.

Tez çalışması süresince NSH protokolünün açık kaynak projelere entegre edilmesi için geliştirmeler yapılmış ve geliştirilen kodları diğer araştırmacıların da kullanabilmesini sağlayacak şekilde Ryu [157] ve Scapy [202] projelerinin kaynak kodları güncellenmiştir. Güncelleme kodları ilgili projeler tarafından onaylanmış ve ana sürüme dahil edilmiştir. Bu projelerin güncel sürümleri yüklendiğinde NSH desteği kullanılabilmektedir. Bu geliştirme sonucunda Ryu çerçevesi ile NSH başlıkları ile işlem yapılabilecek OpenFlow mesajları oluşturulabilmekte, Scapy kütüphanesi ile Ethernet çerçevelerindeki NSH başlıkları işlenebilmektedir. Bu geliştirme ile ayrıca, NSH protokolünü desteklemeyen sanal ağ fonksiyonlarının Y-S3'e entegre edilerek test edilebilmesini sağlayacak SFZ vekil uygulamasının geliştirilebileceği altyapı oluşturulmuştur.

Bu tez çalışması sürecinde elde edilen tecrübeler ve karşılaşılan zorluklar aşağıda özetlenmiştir:

- YTA temelli savunma servislerinin testi için veri setlerinin ve güvenlik test platformlarının olmaması tez çalışmalarında karşılaşılan en büyük zorluklardan biri olmuştur.
- STİ verilerinin çoğu, tehditlere karşı alınabilecek hareket tarzı içermemekte, hareket tarzı içeren verilerde de bu bilgiler düz metin olarak yer almaktadır. Bu sebeple mevcut

STİ verilerinin otomatik ağ savunmasında kullanılabilmesi için ek çalışmalar yapılması gerekmiştir.

- SFZ oluşturmak için kullanılan NSH [18] gibi protokoller tez çalışması süresince sürekli değişmiştir. Bunun sonucunda Y-S3'ün prototipinin kaynak kodları ve test ortamları her seferinde güncellenmiştir.
- Tez kapsamında kullanılan NSH protokolü açık kaynak kodlu projelerde kullanabilmek için Ryu ve Scapy açık kaynak kodlu projelerin kaynak kodlarının güncellenmesi gerekmiştir. Bu da tez çalışmalarındaki geliştirme ve test aşamalarının uzun sürmesine sebep olmuştur.
- Sanal ağ fonksiyonlarının çoğu SFZ yapılarını ve NSH protokolünü desteklememektedir. Bundan dolayı tez çalışmasında gerçekleştirilen testler için NSH destekli örnek ağ fonksiyonlarının yazılması gerekmiştir.
- YTA kontrolcülerinin kuzey yönlü ara yüzleri ve yönetim ara yüzleri sürekli güncellenmektedir. Bu sebeple geliştirilen savunma çözümlerinin YTA kontrolcülerinde yapılan güncellemeler ile birlikte gözden geçirilmesi gerekmiştir. Bu durum da test altyapılarının sürekli güncellenmesine sebep olmuştur.

Bu tezde yeni nesil ağlarda siber tehditlere karşı genişletilebilir, çevik ve proaktif ağ savunması sağlayan Y-S3 adında bir sistem önerisi sunulmuştur. Gelecek çalışma önerileri olarak;

- Y-S3'ün, farklı YTA kontrolcüleri ile çalışabilmesini sağlayacak şekilde güncellenebileceği ve değerlendirilebileceği,
- Y-S3'ün, farklı veya aynı türden birden çok YTA kontrolcüsünü yönetebilecek şekilde genişletilebileceği ve değerlendirilebileceği,
- Y-S3'ün YTA ve geleneksel ağları içeren karma ağları göz önünde bulunduracak şekilde güncellenebileceği ve değerlendirilebileceği,
- YTA anahtarlarındaki istatistik bilgilerini toplayabilecek ve zamansal ölçümler yaparak farklı savunma yöntemleri uygulayabilecek ağ savunma servislerinin geliştirilebileceği ve değerlendirilebileceği,
- Y-S3 kullanılarak YTA savunma servisleri ve servis fonksiyon zincirleri için ağ adli bilişim çözümlerinin geliştirilebileceği,
- Y-S3'e entegre edilecek savunma servislerinin test edilmesi için kullanılabilecek veri setlerinin oluşturulabileceği,

- DDoS saldırı etkisi azaltma servisi, kimlik sahtekârlığı servisi ve hız sınırlama servisi gibi yeni ağ savunma servislerinin ve güvenlik sanal ağ fonksiyonlarının geliştirilerek Y-S3'e eklenebileceği,
- Y-S3'ün büyük ağlarda ve gerçek ortamlarda test edilebileceği,
- Y-S3 ile ağlar arası güvenli ve güvenilir savunma iş birliğini sağlayacak test senaryolarının geliştirilebileceği ve test edilebileceği,
- Y-S3 içinde yer alan çoklu rota servisinin, farklı kısıtlamalarla (entropi, servis kalitesi vb.) ve farklı rota seçim algoritmaları ile test edilebileceği değerlendirilmektedir.

Tez çalışmasının farklı aşamalarında, YTA temelli siber savunma alanındaki bazı açık problemler ve gelecekte ele alınabilecek araştırma konuları derlenmiş ve ayrıntılı olarak Altıncı Bölümde açıklanmıştır. Belirlenen araştırma konularının YTA temelli savunma çözümleri geliştirilmesi için çalışma yapacak araştırmacıların yeni çalışmalarında yol gösterici olacağı değerlendirilmektedir.

KAYNAKLAR

1. Qamar, S., Anwar, Z., Rahman, M. A., Al-Shaer, E., and Chu, B. T. (2017). Data-driven analytics for cyber-threat intelligence and information sharing. *Computers and Security*, *67*, 35–58. https://doi.org/10.1016/j.cose.2017.02.005

2. İnternet: McAfeeLabs. *Threats Report, August 2019*. URL: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-aug-2019.pdf, Son Erişim Tarihi: 08.08.2020.

3. İnternet: Kaspersky. *Security Bulletin 2019, Statistics*. URL: https://securelist.com/kaspersky-security-bulletin-2019-statistics/95475/, Son Erişim Tarihi: 08.08.2020.

4. Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017). DDoS in the IoT: Mirai and other botnets. *IEEE Computer*, *50*(7), 80–84. https://doi.org/10.1109/MC.2017.201

5. Jasper, S. E. (2017). U.S. cyber threat intelligence sharing frameworks. *International Journal of Intelligence and CounterIntelligence*, *30*(1), 53–65. https://doi.org/10.1080/08850607.2016.1230701

6. Masoudi, R., and Ghaffari, A. (2016). Software defined networks: A survey. *Journal of Network and Computer Applications*, *67*, 1–25. https://doi.org/10.1016/j.jnca.2016.03.016

7. Kreutz, D., Ramos, F. M. V., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., and Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of The IEEE*, *103*(1), 14–76. https://doi.org/10.1109/JPROC.2014.2371999

8. McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). OpenFlow: Enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, *38*(2), 69–74. https://doi.org/10.1145/1355734.1355746

9. İnternet: *Network Functions Virtualisation (NFV) architectural framework*. URL: https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf, Son Erişim Tarihi: 09.11.2020.

10. İnternet: *Service Function Chaining (SFC) architecture - RFC 7665*. URL: https://tools.ietf.org/html/rfc7665, Son Erişim Tarihi: 09.11.2020.

11. İnternet: *Structured Threat Information eXpression, STIX 2.1 specification,*. URL: https://www.oasis-open.org/standards/#stix2.1, Son Erişim Tarihi: 09.06.2020.

12. İnternet: *RFC 5070 - The Incident Object Description Exchange Format (IODEF)*. URL: https://tools.ietf.org/html/rfc7203, Son Erişim Tarihi: 09.06.2020.

13. İnternet: *Open Indicators of Compromise (OpenIoC)*. URL: https://github.com/mandiant/OpenIOC_1.1, Son Erişim Tarihi: 09.06.2020.

14. İnternet: Sethi, A., and Barnum, S. *Introduction to attack patterns*. URL: https://www.us-cert.gov/bsi/articles/knowledge/attack-patterns/introduction-to-attack-patterns, Son Erişim Tarihi: 09.06.2020.

15. İnternet: MITRE. *Common Attack Pattern Enumeration and Classification (CAPEC)*. URL: https://capec.mitre.org, Son Erişim Tarihi: 09.06.2020.

16. İnternet: MITRE. *Adversarial tactics, techniques, and common knowledge (MITRE ATT&CK)*. URL: https://attack.mitre.org, Son Erişim Tarihi: 09.06.2020.

17. İnternet: ONF. *OpenFlow Switch Specification, v1.4.1*. URL: https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.4.1.pdf, Son Erişim Tarihi: 08.11.2020.

18. İnternet: *IETF RFC8300 Network Service Header (NSH)*. URL: https://tools.ietf.org/html/rfc8300, Son Erişim Tarihi: 09.06.2020.

19. Mijumbi, R., Serrat, J., Gorricho, J., Bouten, N., Turck, F. De, and Raouf, B. (2016). Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys and Tutorials*, *18*(1), 236–262. https://doi.org/10.1109/COMST.2015.2477041

20. Scott-Hayward, S., Natarajan, S., and Sezer, S. (2016). A survey of security in software defined networks. *IEEE Communications Surveys and Tutorials*, *18*(1), 623–654. https://doi.org/10.1109/COMST.2015.2474118

21. Cui, Y., Yan, L., Li, S., Xing, H., Pan, W., Zhu, J., and Zheng, X. (2016). SD-anti-DDoS: Fast and efficient DDoS defense in software-defined networks. *Journal of Network and Computer Applications*, *68*, 65–79. https://doi.org/10.1016/j.jnca.2016.04.005

22. Shu, Z., Wan, J., Li, D., Lin, J., Vasilakos, A. V., and Imran, M. (2016). Security in software-defined networking: Threats and countermeasures. *Mobile Networks and Applications*, *21*(5), 764–776. https://doi.org/10.1007/s11036-016-0676-x

23. Wang, S., Wu, J., Yang, W., and Guo, L. (2018). Novel architectures and security solutions of programmable software-defined networking: A comprehensive survey. *Frontiers of Information Technology & Electronic Engineering*, *19*(12), 1500–1521. https://doi.org/10.1631/FITEE.1800575

24. Han, T., Jan, S. R. U., Tan, Z., Usman, M., Jan, M. A., Khan, R., and Xu, Y. (2019). A comprehensive survey of security threats and their mitigation techniques for next-generation SDN controllers. *Concurrency and Computation Practice and Experience*, *e5300*, 3–5. https://doi.org/10.1002/cpe.5300

25. Benabbou, J., Elbaamrani, K., and Idboufker, N. (2019). Security in OpenFlow-based SDN, opportunities and challenges. *Photonic Network Communications*, *37*(1), 1–23. https://doi.org/10.1007/s11107-018-0803-7

26. Thimmaraju, K., Shastry, B., Fiebig, T., Hetzelt, F., Seifert, J.-P., Feldmann, A., and Schmid, S. (2018). Taking control of SDN-based cloud systems via the data plane. *Proceedings of the Symposium on SDN Research (SOSR 2018)*, 1–15. https://doi.org/10.1145/3185467.3185468

27. Achleitner, S., La Porta, T., Jaeger, T., and McDaniel, P. (2017). Adversarial network forensics in software defined networking. *The Symposium on SDN Research (SOSR 2017)*, 8–20. https://doi.org/10.1145/3050220.3050223

28. Ghosh, U., Chatterjee, P., and Shetty, S. (2018). Securing SDN-enabled smart power grids: SDN-enabled smart grid security. In J. J. Rodrigues and A. Gawanmeh (Eds.), *Cyber-Physical Systems for Next-Generation Networks* , 79–98. IGI Global. https://doi.org/10.4018/978-1-5225-5510-0.ch004

29. Demirci, S., and Sagiroglu, S. (2018). Software-defined networking for improving security in smart grid systems. *7th International Conference on Renewable Energy Research and Applications (ICRERA 2018)*, 1021–1026. https://doi.org/10.1109/ICRERA.2018.8567005

30. Petroulakis, N. E., Fysarakis, K., Askoxylakis, I., and Spanoudakis, G. (2018). Reactive security for SDN/NFV-enabled industrial networks leveraging service function chaining. *Transactions on Emerging Telecommunications Technologies*, *29*(7), e3269, 1–15. https://doi.org/10.1002/ett.3269

31. Tounsi, W., and Rais, H. (2018). A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Computers and Security*, *72*, 212–233. https://doi.org/10.1016/j.cose.2017.09.001

32. de Melo e Silva, A., Gondim, J. J. C., de Oliveira Albuquerque, R., and Villalba, L. J. G. (2020). A methodology to evaluate standards and platforms within cyber threat intelligence. *Future Internet*, *12*(6), 1–23. https://doi.org/10.3390/fi12060108

33. Ramsdale, A., Shiaeles, S., and Kolokotronis, N. (2020). A comparative analysis of cyber threat intelligence sources, formats and languages. *Electronics*, *9*(5), 1–22. https://doi.org/10.3390/electronics9050824

34. Sauerwein, C., Sillaber, C., Mussmann, A., and Breu, R. (2017). Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives. *The 13th International Conference on Wirtschaftsinformatik*, 837–851.

35. İnternet: *Trusted Automated Exchange of Intelligence Information, TAXII v2.1 Specification*. URL: https://www.oasis-open.org/standards/#taxii2.1, Son Erişim Tarihi: 09.06.2020.

36. İnternet: Darley, T., Kirillov, I., Piazza, R., and Beck, D. *The CybOX language specification*. URL: http://docs.oasis-open.org/cti/cybox/v2.1.1, Son Erişim Tarihi: 09.06.2020.

37. Rhoades, D. (2014). Machine actionable indicators of compromise. *International Carnahan Conference on Security Technology (ICCST 2014)*, 1–5. https://doi.org/10.1109/CCST.2014.6987016

38. Asgarli, E., and Burger, E. (2016). Semantic ontologies for cyber threat sharing standards. *IEEE Symposium on Technologies for Homeland Security (HST 2016)*, 1–6. https://doi.org/10.1109/THS.2016.7568896

39. Tounsi, W., and Rais, H. (2018). A Survey on technical threat intelligence in the age of sophisticated cyber attacks. *Computers and Security*, *72*, 212–233. https://doi.org/10.1016/j.cose.2017.09.001

40. Ring, T. (2014). Threat intelligence: Why people don't share. *Computer Fraud & Security*, *2014*(3), 5–9. https://doi.org/https://doi.org/10.1016/S1361-3723(14)70469-5

41. Choo, K.-K. R. (2011). The cyber threat landscape: Challenges and future research directions. *Computers & Security*, *30*(8), 719–731. https://doi.org/https://doi.org/10.1016/j.cose.2011.08.004

42. İnternet: OASIS. *Structured Threat Information eXpression, STIX 1.2.1 specification*. URL: https://www.oasis-open.org/standards#stix1.2.1, Son Erişim Tarihi: 09.06.2020.

43. Barnum, S., Martin, R., Worrell, B., and Kirillov, I. (2012). *The CybOX language specification*.

44. İnternet: *STIX and TAXII documentation*. URL: https://oasis-open.github.io/cti-documentation/, Son Erişim Tarihi: 10.14.2020.

45. İnternet: *TAXII*. URL: http://taxiiproject.github.io/about/, Son Erişim Tarihi: 05.28.2017.

46. Canbek, G., Sagiroglu, S., and Baykal, N. (2016). New comprehensive taxonomies on mobile Security and malware analysis. *International Journal of Information Security Science*, *5*(4), 106–138.

47. Hansman, S., and Hunt, R. (2005). A taxonomy of network and computer attacks. *Computers and Security*, *24*, 31–43. https://doi.org/10.1016/j.cose.2004.06.011

48. Jin, S., Wang, Y., Cui, X., and Yun, X. (2009). A review of classification methods for network vulnerability. *IEEE International Conference on Systems, Man, and Cybernetics (SMC 2009)*, 1171–1175. https://doi.org/10.1109/ICSMC.2009.5345951

49. Simmons, C. B., Shiva, S. G., Bedi, H., and Dasgupta, D. (2014). AVOIDIT: A cyber attack taxonomy. *The 9th Annual Symposium on Information Assurance (ASIA 2014)*, 2–12.

50. Amer, S. H., and Hamilton Jr., J. A. (2010). Intrusion detection systems (IDS) taxonomy - A short review. *Defensive Cyber Security: Processes and Policies*, *13*(2), 22–30.

51. Lough, D. L. (2001). *A taxonomy of computer attacks with applications to wireless networks*. Doktora Tezi, Virginia Polytechnic Institute and State University Computer Engineering, Blacksburg, Virginia, 47-88.

52. İnternet: *The NTT Security, Global Threat Intelligence Report 2019 - 7*. URL: http://web.archive.org/web/20200521145640/https://www.nttsecurity.com/docs/libraries provider3/resources/2019-gtir/2019_gtir_report_2019_uea_v2.pdf, Son Erişim Tarihi: 09.06.2020.

53. İnternet: Neely, L. *SANS 2017 threat landscape survey: Users on the front line*. URL: https://www.qualys.com/forms/whitepapers/sans-2017-threat-landscape-survey-users-front-line, Son Erişim Tarihi: 09.06.2020.

54. İnternet: Symantec. *Internet Security Threat Report 2017 / 22*. URL: https://www.websecurity.digicert.com/reports-leadgen/istr, Son Erişim Tarihi: 09.06.2020.

55. Wu, Z., Ou, Y., and Liu, Y. (2011). A Taxonomy of network and computer attacks based on responses. *International Conference of Information Technology, Computer Engineering and Management Sciences (ICM 2011)*, 26–29. https://doi.org/10.1109/ICM.2011.363

56. Kjaerland, M. (2006). A taxonomy and comparison of computer security incidents from the commercial and government sectors. *Computers and Security*, *25*, 522–538. https://doi.org/10.1016/j.cose.2006.08.004

57. Uzunov, A. V, and Fernandez, E. B. (2014). An extensible pattern-based library and taxonomy of security threats for distributed systems. *Computer Standards and Interfaces*, *36*(4), 734–747. https://doi.org/10.1016/j.csi.2013.12.008

58. Hoque, N., Bhuyan, M. H., Baishya, R. C., Bhattacharyya, D. K., and Kalita, J. K. (2014). Network attacks: Taxonomy, tools and systems. *Journal of Network and Computer Applications*, *40*, 307–324. https://doi.org/10.1016/j.jnca.2013.08.001

59. Fu, M., Zhenxing, W., Yi, G., and Zhang, L. (2016). A security threats taxonomy for routing system intrusion detection. *12th International Conference on Computational Intelligence and Security (CIS 2016)*, 267–270. https://doi.org/10.1109/CIS.2016.67

60. İnternet: *McAfee Labs 2019 August Threats Report*. URL: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-aug-2019.pdf, Son Erişim Tarihi: 09.06.2020.

61. İnternet: *McAfee Labs 2017 September Threats Report*. URL: https://www.mcafee.com/us/resources/reports/rp-quarterly-threats-sept-2017.pdf, Son Erişim Tarihi: 09.06.2020.

62. Alsmadi, I., and Xu, D. (2015). Security of software defined networks: A survey. *Computers and Security*, *53*, 79–108. https://doi.org/10.1016/j.cose.2015.05.006

63. Ahmad, I., Namal, S., Ylianttila, M., and Gurtov, A. (2015). Security in software defined networks: A survey. *IEEE Communication Surveys and Tutorials*, *17*(4), 2317–2346. https://doi.org/10.1109/COMST.2015.2474118

64. Shaghaghi, A., Kaafar, M. A., Buyya, R., and Jha, S. (2020). Software-defined network (SDN) data plane security: Issues, solutions, and future directions. In B. Gupta, G. Perez, D. Agrawal, and D. Gupta (Eds.), *Handbook of Computer Networks and Cyber Security* , 341–387. Springer. https://doi.org/10.1007/978-3-030-22277-2_14

65. Rawat, D. B., and Reddy, S. R. (2017). Software defined networking architecture, security and energy efficiency: A survey. *IEEE Communications Surveys & Tutorials*, *19*(1), 325–346. https://doi.org/10.1109/COMST.2016.2618874

66. Chica, J. C. C., Imbachi, J. C., and Botero, J. F. (2020). Security in SDN: A comprehensive survey. *Journal of Network and Computer Applications*, 102595. https://doi.org/10.1016/j.jnca.2020.102595

67. Farris, I., Taleb, T., Khettab, Y., and Song, J. (2019). A survey on emerging SDN and NFV security mechanisms for IoT systems. *IEEE Communications Surveys and Tutorials*, *21*(1), 812–837. https://doi.org/10.1109/COMST.2018.2862350

68. Swami, R., Dave, M., and Ranga, V. (2019). Software-defined networking-based DDoS defense mechanisms. *ACM Computing Surveys*, *52*(2), 1–36. https://doi.org/10.1145/3301614

69. Shin, S., Porras, P., Yegneswaran, V., Fong, M., Gu, G., and Tyson, M. (2013). FRESCO: Modular composable security services for software-defined networks. *Network and Distributed System Security Symposium (NDSS 2013)*, 1–16.

70. Reich, J., Monsanto, C., Foster, N., Rexford, J., and Walker, D. (2013). Modular SDN programming with Pyretic. *The Usenix Magazine*, *38*(5), 40–47.

71. Dhawan, M., Poddar, R., Mahajan, K., and Mann, V. (2015). SPHINX: Detecting security attacks in software-defined networks. *Network and Distributed System Security Symposium (NDSS 2015)*, 8–11. https://doi.org/10.14722/ndss.2015.23064

72. Chowdhary, A., Huang, D., Ahn, G.-J., Kang, M., Kim, A., and Velazquez, A. (2019). SDNSOC: Object oriented SDN framework. *Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, 7–12. https://doi.org/10.1145/3309194.3309196

73. Zarca, A. M., Bernabe, J. B., Trapero, R., Rivera, D., Villalobos, J., Skarmeta, A., Bianchi, S., Zafeiropoulos, A., and Gouvas, P. (2019). Security management architecture for NFV/SDN-aware IoT systems. *IEEE Internet of Things Journal*, *6*(5), 8005–8020. https://doi.org/10.1109/JIOT.2019.2904123

74. Karmakar, K. K., Varadharajan, V., and Tupakula, U. (2019). Mitigating attacks in software defined networks. *Cluster Computing*, *22*(4), 1143–1157. https://doi.org/10.1007/s10586-018-02900-2

75. Achleitner, S., Porta, T. La, McDaniel, P., Sugrim, S., Krishnamurthy, S. V, and Chadha, R. (2016). Cyber deception: Virtual networks to defend insider reconnaissance. *8th ACM CCS International Workshop on Managing Insider Security Threats (MIST 2016)*, 57–68. https://doi.org/10.1145/1235

76. Achleitner, S., Porta, T. F. La, McDaniel, P., Sugrim, S., Krishnamurthy, S. V, and Chadha, R. (2017). Deceiving network reconnaissance using SDN-based virtual topologies. *IEEE Transactions on Network and Service Management*, *14*(4), 1098–1112. https://doi.org/10.1109/TNSM.2017.2724239

77. Chiang, C.-Y. J., Gottlieb, Y. M., Sugrim, S. J., Chadha, R., Serban, C., Poylisher, A., Marvel, L. M., and Santos, J. (2016). ACyDS: An adaptive cyber deception system. *The 35th Military Communications Conference (MILCOM 2016)*, 800–805. https://doi.org/10.1109/MILCOM.2016.7795427

78. Robertson, S., Alexander, S., Micallef, J., Pucci, J., Tanis, J., and Macera, A. (2015). CINDAM: Customized information networks for deception and attack mitigation. *IEEE 9th International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW 2015)*, 114–119. https://doi.org/10.1109/SASOW.2015.23

79. Jafarian, J. H., Al-Shaer, E., and Duan, Q. (2012). Openflow random host mutation: Transparent moving target defense using software defined networking. *First Workshop on Hot Topics in Software Defined Networks (HotSDN 2012)*, 127–132. https://doi.org/10.1145/2342441.2342467

80. Jafarian, J. H. H., Al-Shaer, E., and Duan, Q. (2014). Spatio-temporal address mutation for proactive cyber agility against sophisticated attackers. *First ACM Workshop on Moving Target Defense (MTD 2014)*, 69–78. https://doi.org/10.1145/2663474.2663483

81. Jafarian, J. H., Al-Shaer, E., and Duan, Q. (2015). An effective address mutation approach for disrupting reconnaissance attacks. *IEEE Transactions on Information Forensics and Security*, *10*(12), 2562–2577. https://doi.org/10.1109/TIFS.2015.2467358

82. Jafarian, J. H., Al-Shaer, E., and Duan, Q. (2015). Adversary-aware IP Address Randomization for Proactive Agility Against Sophisticated Attackers. *IEEE Conference on Computer Communications (INFOCOM 2015)*, 738–746. https://doi.org/10.1109/INFOCOM.2015.7218443

83. Ma, D., Lei, C., Wang, L., Zhang, H., Xu, Z., and Li, M. (2016). A self-adaptive hopping approach of moving target defense to thwart scanning attacks. In K. Lam, C. Chi, and S. Qing (Eds.), *Lecture Notes in Computer Science (LNCS)* , Vol. 9977, 39–53. Springer. https://doi.org/10.1007/978-3-319-50011-9_4

84. Macfarland, D. C., and Shue, C. A. (2015). The SDN shuffle: Creating a moving-target defense using host-based software-defined networking. *2nd ACM Workshop on Moving Target Defense (MTD 2015)*, 37–41. https://doi.org/10.1145/2808475.2808485

85. Wang, K., Chen, X., and Zhu, Y. (2017). Random domain name and address mutation (RDAM) for thwarting reconnaissance attacks. *PLOS ONE*, *12*(5), 1–22. https://doi.org/10.1371/journal.pone.0177111

86. Wang, L., and Wu, D. (2016). Moving target defense against network reconnaissance with software defined networking. In M. Bishop and A. Nascimento (Eds.), *Lecture Notes in Computer Science* , Vol. 9866, 203–217. https://doi.org/10.1007/978-3-319-45871-7_13

87. Zhao, Z., Liu, F., and Gong, D. (2017). An SDN-based fingerprint hopping method to prevent fingerprinting attacks. *Security and Communication Networks*, *2017*, 1–12. https://doi.org/10.1155/2017/1560594

88. Kampanakis, P., Perros, H., and Beyene, T. (2014). SDN-based solutions for moving target defense network protection. *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2014)*, 1–6. https://doi.org/10.1109/WoWMoM.2014.6918979

89. Cabaj, K., Gregorczyk, M., Mazurczyk, W., Nowakowski, P., and Zorawski, P. (2018). SDN-based mitigation of scanning attacks for the 5G Internet of radio light system. *Proceedings of the 13th International Conference on Availability, Reliability and Security*, 1–10. https://doi.org/10.1145/3230833.3233248

90. İnternet: *Snort - Network intrusion detection & prevention system*. URL: https://www.snort.org/, Son Erişim Tarihi: 09.06.2020.

91. Achleitner, S., La Porta, T., Jaeger, T., and McDaniel, P. (2017). Adversarial network forensics in software defined networking. *Proceedings of the Symposium on SDN Research*, 8–20. https://doi.org/10.1145/3050220.3050223

92. Sahri, N. M., and Okamura, K. (2016). CAuth - Protecting DNS application from spoofing attacks. *International Journal of Computer Science and Network Security*, *16*(6), 125.

93. Sahri, N., and Okamura, K. (2016). Protecting DNS Services from IP Spoofing: SDN Collaborative Authentication Approach. *11th International Conference on Future Internet Technologies (CFI 2016)*, 83–89. https://doi.org/10.1145/2935663.2935666

94. Sahri, N., and Okamura, K. (2016). Collaborative spoofing detection and mitigation-SDN based looping authentication for DNS services. *IEEE 40th Annual Computer Software and Applications Conference (COMPSAC 2016)*, 565–570. https://doi.org/10.1109/COMPSAC.2016.6

95. Masoud, M. Z., Jaradat, Y., and Jannoud, I. (2015). On preventing ARP poisoning attack utilizing software defined network (SDN) paradigm. *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT 2015)*, 1–5. https://doi.org/10.1109/AEECT.2015.7360549

96. Cox, J. H., Clark, R. J., and Owen, H. L. (2016). Leveraging SDN for ARP security. *IEEE Region 3 South East Conference (SoutheastCon 2016)*, 1–8. https://doi.org/10.1109/SECON.2016.7506644

97. Nehra, A., Tripathi, M., and Gaur, M. S. (2017). FICUR: Employing SDN programmability to secure ARP. *IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC 2017)*, 1–8. https://doi.org/10.1109/CCWC.2017.7868450

98. Ubaid, F., Amin, R., Ubaid, F. Bin, and Iqbal, M. M. (2017). Mitigating address spoofing attacks in hybrid SDN. *International Journal of Advanced Computer Science and Applications*, *8*(4), 562–570. https://doi.org/10.14569/IJACSA.2017.080474

99. Alharbi, T., Durando, D., Pakzad, F., and Portmann, M. (2016). Securing ARP in software defined networks. *The 41st Annual IEEE Conference on Local Computer Networks (LCN 2016)*, 523–526. https://doi.org/10.1109/LCN.2016.83

100. Lu, Y., Wang, M., and Huang, P. (2017). An SDN-based authentication mechanism for securing neighbor discovery protocol in IPv6. *Security and Communication Networks*, *2017*, 1–9. https://doi.org/10.1155/2017/5838657

101. Mattos, D. M. F., and Duarte, O. C. M. B. (2016). AuthFlow: Authentication and access control mechanism for software defined networking. *Annals of Telecommunications*, *71*(11–12), 607–615. https://doi.org/10.1007/s12243-016-0505-z

102. Kuliesius, F., and Dangovas, V. (2016). SDN enhanced campus network authentication and access control system. *The Eighth International Conference on Ubiquitous and Future Networks*, 894–899. https://doi.org/10.1109/ICUFN.2016.7536925

103. Kwon, J., Seo, D., Kwon, M., Lee, H., Perrig, A., and Kim, H. (2015). An incrementally deployable anti-spoofing mechanism for software-defined networks. *Computer Communications*, *64*, 1–20. https://doi.org/10.1016/j.comcom.2015.03.003

104. Liu, B., Bi, J., and Zhou, Y. (2016). Source address validation in software defined networks. *ACM SIGCOMM Conference (SIGCOMM 2016)*, 595–596. https://doi.org/10.1145/2934872.2960425

105. Yao, G., Bi, J., and Xiao, P. (2011). Source address validation solution with OpenFlow/NOX architecture. *International Conference on Network Protocols (ICNP 2011)*, 7–12. https://doi.org/10.1109/ICNP.2011.6089085

106. Yao, G., Bi, J., Feng, T., Xiao, P., and Zhou, D. (2014). Performing software defined route-based IP spoofing filtering with SEFA. *International Conference on Computer Communications and Networks*, 1–8. https://doi.org/10.1109/ICCCN.2014.6911784

107. Jafarian, J. H., Al-Shaer, E., and Duan, Q. (2013). Formal approach for route agility against persistent attackers. In J. Crampton, S. Jajodia, and K. Mayes (Eds.), *Lecture Notes in Computer Science (LNCS)* , Vol. 8134, 237–254. Springer International Publishing. https://doi.org/10.1007/978-3-642-40203-6_14

108. Gillani, F., Al-Shaer, E., Lo, S., Duan, Q., Ammar, M., and Zegura, E. (2015). Agile virtualized infrastructure to proactively defend against cyber attacks. *IEEE Conference on Computer Communications (INFOCOM 2015)*, 729–737. https://doi.org/10.1109/INFOCOM.2015.7218442

109. Fichera, S., Galluccio, L., Grancagnolo, S. C., Morabito, G., and Palazzo, S. (2015). OPERETTA: An Openflow-based remedy to mitigate TCP SYNFLOOD attacks against web servers. *Computer Networks*, *92*, 89–100. https://doi.org/10.1016/j.comnet.2015.08.038

110. Shin, S., Yegneswaran, V., Porras, P., and Gu, G. (2013). AVANT-GUARD: Scalable and vigilant switch flow management in software-defined networks. *ACM SIGSAC Conference on Computer and Communications Security (CCS 2013)*, 413–424. https://doi.org/10.1145/2508859.2516684

111. Giotis, K., Argyropoulos, C., Androulidakis, G., Kalogeras, D., and Maglaris, V. (2014). Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments. *Computer Networks*, *62*, 122–136. https://doi.org/10.1016/j.bjp.2013.10.014

112. Hussein, A., Elhajj, I. H., Chehab, A., and Kayssi, A. (2016). SDN security plane: An architecture for resilient security services. *IEEE International Conference on Cloud Engineering Workshop (IC2EW 2016)*, 54–59. https://doi.org/10.1109/IC2EW.2016.15

113. Joldzic, O., Djuric, Z., and Vuletic, P. (2016). A transparent and scalable anomaly-based DoS detection method. *Computer Networks*, *104*, 27–42. https://doi.org/10.1016/j.comnet.2016.05.004

114. Li, J., Berg, S., Zhang, M., Reiher, P., and Wei, T. (2014). DrawBridge: Software-defined DDoS-resistant traffic engineering. *ACM SIGCOMM Computer Communication Review*, *44*(4), 591–592. https://doi.org/10.1145/2619239.2631469

115. Miao, R., Yu, M., and Jain, N. (2014). NIMBUS: Cloud-scale attack detection and mitigation. *ACM SIGCOMM Computer Communication Review*, *44*, 121–122. https://doi.org/10.1145/2740070.2631446

116. Oktian, Y. E., Lee, S., and Lee, H. (2014). Mitigating denial of service (DoS) attacks in OpenFlow networks. *International Conference on Information and Communication Technology Convergence*, 325–330. https://doi.org/10.1109/ICTC.2014.6983147

117. Piedrahita, A. F. M., Rueda, S., Mattos, D. M. F., and Duarte, O. C. M. B. (2015). FlowFence: A denial of service defense system for software defined networking. *Global Information Infrastructure and Networking Symposium (GIIS 2015)*, 1–6. https://doi.org/10.1109/SP.2013.19

118. Wang, T., Chen, H., Cheng, G., and Lu, Y. (2018). SDNManager: A safeguard architecture for SDN DoS attacks based on bandwidth prediction. *Security and Communication Networks, 2018*, 1–16. https://doi.org/10.1155/2018/7545079

119. Wang, B., Zheng, Y., Lou, W., and Hou, Y. T. (2015). DDoS attack protection in the era of cloud computing and software-defined networking. *Computer Networks*, *81*, 308–319. https://doi.org/10.1016/j.comnet.2015.02.026

120. Shang, G., Zhe, P., Bin, X., Aiqun, H., and Kui, R. (2017). FloodDefender: Protecting data and control plane resources under SDN-aimed DoS attacks. *2017 IEEE International Conference on Computer Communications (INFOCOM 2017)*, 1–9. https://doi.org/10.1109/INFOCOM.2017.8057009

121. Wang, S., Chandrasekharan, S., Gomez, K., Kandeepan, S., Al-Hourani, A., Asghar, M. R., Russello, G., and Zanna, P. (2018). SECOD: SDN secure control and data plane algorithm for detecting and defending against DoS attacks. *2018 IEEE/IFIP Network Operations and Management Symposium (NOMS 2018)*, 1–5. https://doi.org/10.1109/NOMS.2018.8406196

122. Goksel, N., and Demirci, M. (2019). DoS attack detection using packet statistics in SDN. *2019 IEEE International Symposium on Networks, Computers and Communications (ISNCC 2019)*, 1–6. https://doi.org/10.1109/ISNCC.2019.8909114

123. Xie, R., Xu, M., Cao, J., and Li, Q. (2019). SoftGuard: Defend against the low-rate TCP attack in SDN. *IEEE International Conference on Communications (ICC 2019)*, 1–6. https://doi.org/10.1109/ICC.2019.8761806

124. Kang, M. S., Lee, S. B., and Gligor, V. D. (2013). The crossfire attack. *2013 IEEE Symposium on Security and Privacy*, 127–141. https://doi.org/10.1109/SP.2013.19

125. Wang, L., Li, Q., Jiang, Y., and Wu, J. (2016). Towards mitigating link flooding attack via incremental SDN deployment. *IEEE Symposium on Computers and Communication (ISCC 2016)*, 397–402. https://doi.org/10.1109/ISCC.2016.7543772

126. Wang, J., Wen, R., Li, J., Yan, F., Zhao, B., and Yu, F. (2019). Detecting and mitigating target link-flooding attacks using SDN. *IEEE Transactions on Dependable and Secure Computing*, *16*(6), 944–956. https://doi.org/10.1109/TDSC.2018.2822275

127. İnternet: *Floodlight OpenFlow Controller (OSS)*. URL: https://github.com/floodlight, Son Erişim Tarihi: 09.06.2020.

128. Zheng, J., Li, Q., Gu, G., Cao, J., Yau, D. K. Y., and Wu, J. (2018). Realtime DDoS defense using COTS SDN switches via adaptive correlation analysis. *IEEE Transactions on Information Forensics and Security*, *13*(7), 1838–1853. https://doi.org/10.1109/APSITT.2015.7217098

129. Li, C., Wu, Y., Yuan, X., Sun, Z., Wang, W., Li, X., and Gong, L. (2018). Detection and defense of DDoS attack-based on deep learning in OpenFlow-based SDN. *International Journal of Communication Systems*, *31*(5), e3497, 1–15. https://doi.org/10.1002/dac.3497

130. Cui, J., He, J., Xu, Y., and Zhong, H. (2018). TDDAD: Time-based detection and defense scheme against DDoS attack on SDN controller. In W. Susilo and G. Yang (Eds.), *Lecture Notes in Computer Science* , Vol. 10946, 649–665. Springer. https://doi.org/10.1007/978-3-319-93638-3_37

131. Singh, R., Tanwar, S., and Sharma, T. P. (2020). Utilization of blockchain for mitigating the distributed denial of service attacks. *Security and Privacy*, *3*, e96, 1–13. https://doi.org/10.1002/spy2.96

132. El Houda, Z. A., Hafid, A. S., and Khoukhi, L. (2019). Cochain-SC: An intra-and inter-domain DDoS mitigation scheme based on blockchain using SDN and smart contract. *IEEE Access*, *7*, 98893–98907. https://doi.org/10.1109/ACCESS.2019.2930715

133. İnternet: *Ethereum - A global, open source platform for decentralized applications*. URL: https://ethereum.org, Son Erişim Tarihi: 09.06.2020.

134. Rodrigues, B., Bocek, T., Lareida, A., Hausheer, D., Rafati, S., and Stiller, B. (2017). A blockchain-based architecture for collaborative DDoS mitigation with smart contracts. In D. Tuncer, R. Koch, R. Badonnel, and B. Stiller (Eds.), *Lecture Notes in Computer Science (LNCS)* , Vol. 10356, 16–29. https://doi.org/10.1007/978-3-319-60774-0_2

135. Abou El Houda, Z., Hafid, A., and Khoukhi, L. (2019). Co-IoT: A collaborative DDoS mitigation scheme in IoT environment based on blockchain using SDN. *2019 IEEE Global Communications Conference (GLOBECOM 2019)*, 1–6. https://doi.org/10.1109/GLOBECOM38437.2019.9013542

136. Xing, J., Wu, W., and Chen, A. (2019). Architecting programmable data plane defenses into the network with Fastflex. *Proceedings of the 18th ACM Workshop on Hot Topics in Networks (HotNets 2019)*, 161–169. https://doi.org/10.1145/3365609.3365860

137. Zhang, M., Li, G., Wang, S., Liu, C., Chen, A., Hu, H., Gu, G., Li, Q., Xu, M., and Wu, J. (2020). Poseidon: Mitigating volumetric DDoS attacks with programmable switches. *The Network and Distributed System Security Symposium (NDSS)*, 1–18. https://doi.org/10.14722/ndss.2020.24007

138. Zhao, Z., Gong, D., Lu, B., Liu, F., and Zhang, C. (2016). SDN-based double hopping communication against sniffer attack. *Mathematical Problems in Engineering*, *2016*, 1–13. https://doi.org/10.1155/2016/8927169

139. Duan, Q., Al-Shaer, E., and Jafarian, H. (2013). Efficient random route mutation considering flow and network constraints. *IEEE Conference on Communications and Network Security (CNS 2013)*, 260–268. https://doi.org/10.1109/CNS.2013.6682715

140. Liu, J., Zhang, H., and Guo, Z. (2017). A defense mechanism of random routing mutation in SDN. *IEICE Transactions on Information and Systems*, *E100D*(5), 1046–1054. https://doi.org/10.1587/transinf.2016EDP7377

141. Furukawa, M., Kuroda, K., Ogawa, T., and Miyaho, N. (2015). Highly secure communication service architecture using SDN switch. *2015 10th Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT 2015)*, 1–3. https://doi.org/10.1109/APSITT.2015.7217098

142. Ma, D., Wang, L., Lei, C., Xu, Z., Zhang, H., and Li, M. (2016). Thwart eavesdropping attacks on network communication based on moving target defense. *The 35th IEEE International Performance Computing and Communications Conference (IPCCC 2016)*, 1–2. https://doi.org/10.1109/PCCC.2016.7820610

143. Germano Da Silva, E., Dias Knob, L. A., Wickboldt, J. A., Gaspary, L. P., Granville, L. Z., and Schaeffer-Filho, A. (2015). Capitalizing on SDN-based SCADA systems: An anti-eavesdropping case-study. *IFIP/IEEE International Symposium on Integrated Network Management*, 165–173. https://doi.org/10.1109/INM.2015.7140289

144. Villain, B., Ridoux, J., Rotrou, J., and Pujolle, G. (2014). Mutualized OpenFlow architecture for network access management. *IEEE 3rd International Conference on Cloud Networking*, 413–419. https://doi.org/10.1109/CloudNet.2014.6969030

145. Cabaj, K., and Mazurczyk, W. (2016). Using software-defined networking for ransomware mitigation: The case of CryptoWall. *IEEE Network*, *30*(6), 14–20. https://doi.org/10.1109/MNET.2016.1600110NM

146. Ceron, J. M., Margi, B. M., Granville, L. Z., Gualberto, L., and Goncalves, B. (2016). MARS: An SDN-based malware analysis solution. *IEEE Symposium on Computers and Communication*, 525–530. https://doi.org/10.1109/ISCC.2016.7543792

147. Hu, Y., Zheng, K., Wang, X., and Yang, Y. (2017). WORM-HUNTER : A worm guard system using software-defined networking. *KSII Transactions on Internet And Information Systems*, *11*(1), 484–510. https://doi.org/10.3837/tiis.2017.01.026

148. Jin, R., and Wang, B. (2013). Malware detection for mobile devices using software-defined networking. *Second GENI Research and Educational Experiment Workshop (GREE 2013)*, 81–88. https://doi.org/10.1109/GREE.2013.24

149. Masoud, M., Jaradat, Y., and Ahmad, A. Q. (2016). On tackling social engineering web phishing attacks utilizing software defined networks (SDN) approach. *2nd International Conference on Open Source Software Computing (OSSCOM 2016)*, 1–6. https://doi.org/10.1109/OSSCOM.2016.7863679

150. Chin, T., Xiong, K., and Hu, C. (2018). Phishlimiter: A phishing detection and mitigation approach using software-defined networking. *IEEE Access*, *6*, 42516–42531. https://doi.org/10.1109/ACCESS.2018.2837889

151. Lim, S., Ha, J., Kim, H., Kim, Y., and Yang, S. (2014). A SDN-oriented DDoS blocking scheme for botnet-based attacks. *6th International Conference on Ubiquitous and Future Networks (ICUFN 2014)*, 63–68. https://doi.org/10.1109/ICUFN.2014.6876752

152. Shtern, M., Sandel, R., Litoiu, M., Bachalo, C., and Theodorou, V. (2014). Towards mitigation of low and slow application DDoS attacks. *IEEE International Conference on Cloud Engineering*, 604–609. https://doi.org/10.1109/IC2E.2014.38

153. Yurekten, O., and Demirci, M. (2017). Using cyber threat intelligence in SDN security. *International Conference on Computer Science and Engineering (UBMK 2017)*, 377–382. https://doi.org/10.1109/UBMK.2017.8093415

154. Bakhshi, T. (2017). State of the art and recent research advances in software defined networking. *Wireless Communications and Mobile Computing*, *2017*, 1–35. https://doi.org/10.1155/2017/7191647

155. İnternet: *The POX network software platform*. URL: https://github.com/noxrepo/pox, Son Erişim Tarihi: 09.06.2020.

156. Gude, N., Koponen, T., Pettit, J., Pfaff, B., Casado, M., McKeown, N., and Shenker, S. (2008). NOX: Towards an operating system for networks. *ACM SIGCOMM Computer Communication Review*, *38*(3), 105–110. https://doi.org/10.1145/1384609.1384625

157. İnternet: *Ryu SDN framework*. URL: https://github.com/faucetsdn/ryu, Son Erişim Tarihi: 09.06.2020.

158. İnternet: Mininet. *Mininet: An instant virtual network on your laptop (or other PC)*. URL: http://mininet.org/, Son Erişim Tarihi: 09.06.2020.

159. İnternet: *Open vSwitch, Open virtual switch*. URL: http://openvswitch.org, Son Erişim Tarihi: 09.06.2020.

160. Jo, H., Kim, J., Porras, P., Yegneswaran, V., and Shin, S. (2020). GapFinder: Finding inconsistency of security information from unstructured text. *IEEE Transactions on Information Forensics and Security*, *16*, 86–99. https://doi.org/10.1109/tifs.2020.3003570

161. Liao, X., Yuan, K., Wang, X., Li, Z., Xing, L., and Beyah, R. (2016). Acing the IoC game: Toward automatic discovery and analysis of open-source cyber threat intelligence. *ACM SIGSAC Conference on Computer and Communications*, 755–766. https://doi.org/10.1145/2976749.2978315

162. Kim, G., Lee, C., Jo, J., and Lim, H. (2020). Automatic extraction of named entities of cyber threats using a deep Bi-LSTM-CRF network. *International Journal of Machine Learning and Cybernetics*, *11*(10), 2341–2355. https://doi.org/10.1007/s13042-020-01122-6

163. Husari, G., Al-Shaer, E., Ahmed, M., Chu, B., and Niu, X. (2017). TTPDrill: Automatic and accurate extraction of threat actions from unstructured text of CTI Sources. *The 33rd Annual Computer Security Applications Conference*, 103–115. https://doi.org/10.1145/3134600.3134646

164. Sabottke, C., Suciu, O., and Dumitras, T. (2015). Vulnerability disclosure in the age of social media: Exploiting Twitter for predicting real-world exploits. *Proceedings of the 24th USENIX Security Symposium*, 1041–1056.

165. Zhao, J., Yan, Q., Li, J., Shao, M., He, Z., and Li, B. (2020). TIMiner: Automatically extracting and analyzing categorized cyber threat intelligence from social data. *Computers and Security*, *95*, 1–14. https://doi.org/10.1016/j.cose.2020.101867

166. Joshi, A., Lal, R., Finin, T., and Joshi, A. (2013). Extracting cybersecurity related linked data from text. *2013 IEEE 7th International Conference on Semantic Computing (ICSC 2013)*, 252–259. https://doi.org/10.1109/ICSC.2013.50

167. Jones, C. L., Bridges, R. A., Huffer, K. M. T., and Goodall, J. R. (2015). Towards a relation extraction framework for cyber-security concepts. *ACM International Conference Proceeding Series*, 1–4. https://doi.org/10.1145/2746266.2746277

168. Oosthoek, K., and Doerr, C. (2020). Cyber threat intelligence: A product without a process? *International Journal of Intelligence and CounterIntelligence*, 1–16. https://doi.org/10.1080/08850607.2020.1780062

169. Kaloudi, N., and Jingyue, L. I. (2020). The AI-based cyber threat landscape: A survey. *ACM Computing Surveys*, *53*(1), 1–34. https://doi.org/10.1145/3372823

170. İnternet: Mittal, S., Joshi, A., and Finin, T. *Cyber-All-Intel: An AI for security related threat intelligence*. URL: http://arxiv.org/abs/1905.02895, Son Erişim Tarihi: 09.15.2020.

171. Koloveas, P., Chantzios, T., Tryfonopoulos, C., and Skiadopoulos, S. (2019). A crawler architecture for harvesting the clear, social, and dark web for IoT-related cyber-threat intelligence. *Proceedings - 2019 IEEE World Congress on Services, SERVICES 2019*, 3–8. https://doi.org/10.1109/SERVICES.2019.00016

172. Khurana, N., Mittal, S., Piplai, A., and Joshi, A. (2019). Preventing poisoning attacks on AI based threat intelligence systems. *IEEE International Workshop on Machine Learning for Signal Processing (MLSP 2019)*, 1–6. https://doi.org/10.1109/MLSP.2019.8918803

173. Iqbal, Z., and Anwar, Z. (2020). SCERM—A novel framework for automated management of cyber threat response activities. *Future Generation Computer Systems*, *108*, 687–708. https://doi.org/10.1016/j.future.2020.03.030

174. Schlette, D., Böhm, F., Caselli, M., and Pernul, G. (2020). Measuring and visualizing cyber threat intelligence quality. *International Journal of Information Security*, 1–9. https://doi.org/10.1007/s10207-020-00490-y

175. Al-Mohannadi, H., Awan, I., and Al Hamar, J. (2020). Analysis of adversary activities using cloud-based web services toenhance cyber threat intelligence. *Service Oriented Computing and Applications*, *14*(3), 175–187. https://doi.org/10.1007/s11761-019-00285-7

176. Arıkan, S. M. (2019). *Veri madenciliği temelli siber tehdit istihbaratı*. Yüksek Lisans Tezi, Gazi Üniversitesi Bilişim Enstitüsü, Ankara, 1-148.

177. Preuveneers, D., Joosen, W., Bernal Bernabe, J., and Skarmeta, A. (2020). Distributed security framework for reliable threat intelligence sharing. *Security and Communication Networks, 2020*, 1–15. https://doi.org/10.1155/2020/8833765

178. Gong, S., and Lee, C. (2020). BLOCIS: Blockchain-based cyber threat intelligence sharing framework for sybil-resistance. *Electronics*, *9*(3), 1–20. https://doi.org/10.3390/electronics9030521

179. Homan, D., Shiel, I., and Thorpe, C. (2019). A new network model for cyber threat intelligence sharing using blockchain technology. *2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS 2019)*. https://doi.org/10.1109/NTMS.2019.8763853

180. Riesco, R., Larriva-Novo, X., and Villagra, V. A. (2020). Cybersecurity threat intelligence knowledge exchange based on blockchain. *Telecommunication Systems*, *73*(2), 259–288. https://doi.org/10.1007/s11235-019-00613-4

181. Amthor, P., Fischer, D., Kühnhauser, W. E., and Stelzer, D. (2019). Automated cyber threat sensing and responding: Integrating threat intelligence into security-policy-controlled systems. *The 14th International Conference on Availability, Reliability and Security*, 1–10. https://doi.org/10.1145/3339252.3340509

182. Appala, S., Way, C., Jose, S., Cam-winget, N., Mcgrew, D., and Verma, J. (2015). An actionable threat intelligence system using a publish-subscribe communications model. *The 2nd ACM Workshop on Information Sharing and Collaborative Security*, 61–70. https://doi.org/10.1145/2808128.2808131

183. Sung, Y., Sharma, P. K., Lopez, E. M., and Park, J. H. (2016). FS-OpenSecurity: A taxonomic modeling of security threats in SDN for future sustainable computing. *Sustainability*, *8*(9), 1–26. https://doi.org/10.3390/su8090919

184. Yücel, Ç., Koltuksuz, A., Ödemiş, M., Kademi, A. M., and Özbilgin, G. (2018). A Programmable Threat Intelligence Framework for Containerized Clouds. *The 13th International Conference on Cyber Warfare and Security*, 1–1.

185. Crespo, B. G. N., and Garwood, A. (2014). Fighting botnets with cyber-security analytics. *The 9th International Conference on Availability, Reliability and Security (ARES 2014)*, 192–198. https://doi.org/10.1109/ARES.2014.33

186. Ancieta, J. R. Q., and Rothenberg, C. E. (2015). IntelFlow: Towards adding cyber threat intelligence to software defined networks. *XV Brazilian Symposium on Information and System Security (SBSEG 2015)*, 1–4.

187. Garcia, J. B., Vilchez, V. S., Castro, J. Z., and Arroyo, J. L. Q. (2019). Using cyber threat intelligence to prevent malicious known traffic in a SDN physical testbed. *The IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing*, 1–4. https://doi.org/10.1109/INTERCON.2019.8853616

188. İnternet: *Collective Intelligence Framework (CIF)*. URL: https://github.com/csirtgadgets/bearded-avenger-deploymentkit, Son Erişim Tarihi: 09.06.2020.

189. Jin, R., and Wang, B. (2013). Malware detection for mobile devices using software-defined networking. *Second GENI Research and Educational Experiment Workshop (GREE2013)*, 81–88. https://doi.org/10.1109/GREE.2013.24

190. Peuster, M., Kampmeyer, J., and Karl, H. (2018). Containernet 2.0: A rapid prototyping platform for hybrid service function chains. *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft 2018)*, 335–337. https://doi.org/10.1109/NETSOFT.2018.8459905

191. İnternet: *Opendaylight*. URL: https://www.opendaylight.org/, Son Erişim Tarihi: 09.06.2020.

192. İnternet: *AlienVault*. URL: http://otx.alienvault.com, Son Erişim Tarihi: 09.06.2020.

193. İnternet: *Threat Actor Lab*. URL: http://edge.threatactorlab.com/, Son Erişim Tarihi: 09.06.2020.

194. İnternet: *Taxii Stand*. URL: https://test.taxiistand.com/, Son Erişim Tarihi: 09.06.2020.

195. İnternet: *Hail a Taxii*. URL: http://hailataxii.com/, Son Erişim Tarihi: 09.06.2020.

196. De Maesschalck, S., Colle, D., Lievens, I., Pickavet, M., Demeester, P., Mauz, C., Jaeger, M., Inkret, R., Mikac, B., and Derkacz, J. (2003). Pan-european optical transport networks: An availability-based comparison. *Photonic Network Communications*, *5*(3), 203–225. https://doi.org/10.1023/A:1023088418684

197. İnternet: *Suricata - Open source IDS / IPS / NSM engine*. URL: https://suricata-ids.org, Son Erişim Tarihi: 09.06.2020.

198. İnternet: *Nmap: The network mapper*. URL: https://nmap.org/, Son Erişim Tarihi: 09.06.2020.

199. İnternet: *Hping - Active network security tool*. URL: http://www.hping.org, Son Erişim Tarihi: 09.06.2020.

200. İnternet: *iPerf - The ultimate speed test tool for TCP, UDP and SCTP*. URL: https://iperf.fr, Son Erişim Tarihi: 09.06.2020.

201. İnternet: *Ulusal Siber Olaylara Müdahale Merkezi (USOM), zararlı bağlantılar*. URL: https://www.usom.gov.tr/zararli-baglantilar/1.html, Son Erişim Tarihi: 08.06.2020.

202. İnternet: *Scapy - Packet crafting for Python2 and Python3*. URL: https://scapy.net/, Son Erişim Tarihi: 09.06.2020.

203. Hu, H., Wang, Z., Cheng, G., and Wu, J. (2017). MNOS: A mimic network operating system for software defined networks. *IET Information Security*, *11*(6), 345–355. https://doi.org/10.1049/iet-ifs.2017.0085

204. Sharma, P. K., Singh, S., Jeong, Y.-S., and Park, J. H. (2017). Distblocknet: A distributed blockchains-based secure SDN architecture for IoT networks. *IEEE Communications Magazine*, *55*(9), 78–85. https://doi.org/10.1109/MCOM.2017.1700041

205. Yang, H., Liang, Y., Yao, Q., Guo, S., Yu, A., and Zhang, J. (2019). Blockchain-based secure distributed control for software defined optical networking. *China Communications*, *16*(6), 42–54. https://doi.org/10.23919/JCC.2019.06.004

206. Sharma, P. K., and Park, J. H. (2018). Blockchain based hybrid network architecture for the smart city. *Future Generation Computer Systems*, *86*, 650–655. https://doi.org/10.1016/j.future.2018.04.060

207. Yang, H., Liang, Y., Yuan, J., Yao, Q., Yu, A., and Zhang, J. (2020). Distributed blockchain-based trusted multi-domain collaboration for mobile edge computing in 5G and beyond. *IEEE Transactions on Industrial Informatics*, *16*(11), 7094–7104. https://doi.org/10.1109/tii.2020.2964563

208. Xie, L., Ding, Y., Yang, H., and Wang, X. (2019). Blockchain-based secure and trustworthy Internet of things in SDN-enabled 5G-VANETs. *IEEE Access*, *7*, 56656–56666. https://doi.org/10.1109/ACCESS.2019.2913682

209. Ali, M. S., Vecchio, M., Pincheira, M., Dolui, K., Antonelli, F., and Rehmani, M. H. (2018). Applications of blockchains in the Internet of things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, *21*(2), 1676–1717. https://doi.org/10.1109/COMST.2018.2886932

210. Pohrmen, F. H., Das, R. K., and Saha, G. (2019). Blockchain-based security aspects in heterogeneous Internet-of-things networks: A survey. *Transactions on Emerging Telecommunications Technologies*, *30*(10), 1–26. https://doi.org/10.1002/ett.3741

211. Yang, H., Yuan, J., Yao, H., Yao, Q., Yu, A., and Zhang, J. (2019). Blockchain-based hierarchical trust networking for JointCloud. *IEEE Internet of Things Journal*, *7*(3), 1667–1677. https://doi.org/10.1109/JIOT.2019.2961187

212. Latah, M., and Toker, L. (2019). Artificial intelligence enabled software-defined networking: A comprehensive overview. *IET Networks*, *8*(2), 79–99. https://doi.org/10.1049/iet-net.2018.5082

213. Chen, X.-F., and Yu, S.-Z. (2016). CIPA: A collaborative intrusion prevention architecture for programmable network and SDN. *Computers and Security*, *58*, 1–19. https://doi.org/10.1016/j.cose.2015.11.008

214. Phan, T. V., Bao, N. K., and Park, M. (2017). A novel hybrid flow-based handler with DDoS attacks in software-defined networking. *16th IEEE International Conference on Scalable Computing and Communications*, 350–357. https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0069

215. Boero, L., Marchese, M., and Zappatore, S. (2017). Support vector machine meets software defined networking in IDS domain. *Proceedings of the 29th International Teletraffic Congress, ITC 2017*, *3*, 25–30. https://doi.org/10.23919/ITC.2017.8065806

216. Su, S. C., Chen, Y. R., Tsai, S. C., and Lin, Y. B. (2018). Detecting P2P botnet in software defined networks. *Security and Communication Networks*, *2018*. https://doi.org/10.1155/2018/4723862

217. Dargahi, T., Caponi, A., Ambrosin, M., Bianchi, G., and Conti, M. (2017). A survey on the security of stateful SDN data planes. *IEEE Communications Surveys and Tutorials*, *19*(3), 1701–1725. https://doi.org/10.1109/COMST.2017.2689819

218. İnternet: *P4 Language Specification*, *v1.2.1*. URL: https://p4.org/p4-spec/docs/P4-16-v1.2.1.html, Son Erişim Tarihi: 09.12.2020.

219. Bosshart, P., Daly, D., Gibb, G., Izzard, M., McKeown, N., Rexford, J., Schlesinger, C., Talayco, D., Vahdat, A., Varghese, G., and Walker, D. (2014). P4: Programming protocol-independent packet processors. *Computer Communication Review*, *44*(3), 87–95. https://doi.org/10.1145/2656877.2656890

220. Sivaraman, A., Cheung, A., Budiu, M., Kim, C., Alizadeh, M., Balakrishnan, H., Varghese, G., McKeown, N., and Licking, S. (2016). Packet transactions: High-level programming for line-rate switches. *Proceedings of the 2016 ACM Conference on Special Interest Group on Data Communication (SIGCOMM 2016)*, 15–28. https://doi.org/10.1145/2934872.2934900

221. Connell, W., Menasce, D. A., and Albanese, M. (2018). Performance modeling of moving target defenses with reconfiguration limits. *IEEE Transactions on Dependable and Secure Computing*, 1–14. https://doi.org/10.1109/TDSC.2018.2882825

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : YÜREKTEN, Özgür

Uyruğu : T.C.

Doğum tarihi ve yeri : 17.08.1979, Tavas

Medeni hali : Evli

Telefon : 0 (312) 427 73 66 - 4222

Faks : 0 (312) 427 88 44

e-posta : ozgur.yurekten@tubitak.gov.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Doktora	Gazi Üniversitesi / Bilgisayar Mühendisliği	Devam ediyor
Yüksek lisans	Gazi Üniversitesi / Bilgisayar Mühendisliği	2007
Lisans	Başkent Üniversitesi / Bilgisayar Mühendisliği	2001
Lise	Yunusemre Anadolu Öğretmen Lisesi	1997

İş Deneyimi

Yıl	Yer	Görev
2016-Halen	TÜBİTAK / Siber Güvenlik Enstitüsü	Başuzman Araştırmacı
2012-2016	TÜBİTAK / Yazılım Teknolojileri Ens.	Başuzman Araştırmacı
2006-2012	TÜBİTAK / UEKAE – G222	Uzman Araştırmacı
2001-2006	TÜBİTAK / UEKAE	Araştırmacı
2000-2001	TÜBİTAK / UEKAE	Yarı Zamanlı Arş.

Yabancı Dil

İngilizce

Yayınlar

1 Yurekten, O. and Demirci, M. (Değerlendirmede). Citadel: cyber threat intelligence assisted defense system for software-defined networks. *Computer Networks*.

2. Yurekten, O. and Demirci, M. (2021). SDN-based cyber defense: A survey. *Future Generation Computer Systems*, 115, 126–149.

3. Arıkan, S. M. ve Yürekten, Ö. (2020). Kurumsal projelerde yazılım risk yönetimi süreç iyileştirme deneyimi. *14. Ulusal Yazılım Mühendisliği Sempozyumu (UYMS 2020)*. İstanbul.

4. Yılmaz, T., Arıkan, S.M., Su, F. ve Yürekten, Ö., (2020). Kurumsal uygulamalarda kişisel verilerin korunması. *14. Ulusal Yazılım Mühendisliği Sempozyumu (UYMS 2020)*. İstanbul.

5. Demirci, S., Yürekten, Ö., ve Demirci, M. (2019). Yazılım tanımlı ağlar ve siber güvenlik. Ş. Sağıroğlu (Ed.), Siber Güvenlik ve Savunma: Standartlar ve Uygulamalar. 122–152.

6. Yurekten, O. and Demirci, M. (2017). Using cyber threat ıntelligence in SDN security. *2nd International Conference on Computer Science and Engineering (UBMK)*, 377–385. Antalya. https://doi.org/10.1109/UBMK.2017.8093415

7. Selçuk, G. H., Yürekten, Ö., Çoştu, A. B., Karagöl, Y., Sünsüli, M. ve Gök, H. (2016). Kamu kurumlari için bir e-dönüşüm planlama modeli. *10. Ulusal Yazılım Mühendisliği Sempozyumu (UYMS 2016)*, 123–128. Çanakkale.

8. Yürekten, Ö. and Sağıroğlu, Ş. (2013). Secure software development and threat modelling. *1st International Symposium on Digital Forensics and Security (ISDFS 2013)*, 313–322. Elazığ.

9. Yürekten, Ö. ve Bilge, H. S. (2009). Hiyerarşik bir veritabanının xml ile modellenmesi. *Gazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi*, 24(4), 629-639.

10. Yürekten Ö., Dinçer K., Akar B., Sungur M. and Özbudak E.K. (2006). Migrating a hierarchical legacy database application onto an xml-based service-oriented web platform. *Lecture Notes in Computer Science*, 4263, 645–654.

11. Tümay A., Dinçer K., Yürekten Ö., Sungur M., Dikici A. and Tambağ Y. (2006). A metadata repository model to ıntegrate heterogeneous databases of hardware dependent legacy systems. *Lecture Notes in Computer Science*, 4243, 149–157.

12. Yürekten, Ö., Dinçer, K., Tanyer, S. G. ve İngeç, T. S. (2002). İşaret karakteristikleri kütüphanesi uygulama yazılımı, 10*. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2002),* 676-680. Denizli.

13. Yürekten, Ö., Oğuz, B., Durgar, İ. ve Dinçer, K. (1999). Üniversite ortamında internet erişimli bir öğrenci yönetim ve karar destek sistemi uygulaması. *5. Türkiye'de İnternet Konferansı.* Ankara.

Hobiler

Masa Tenisi, Pul Koleksiyonu

GAZİ GELECEKTİR..