**T.C.**

**FIRAT ÜNİVERSİTESİ**

**FEN BİLİMLERİ ENSTİTÜSÜ**

# BİLİŞİM SİSTEMLERİ İÇİN SALDIRI TESPİT VE ENGELLEME YAKLAŞIMLARININ TASARIMI VE GERÇEKLEŞTİRİLMESİ

**DOKTORA TEZİ**

**Yük. Müh. Muhammet BAYKARA**

**(121137202)**

**Anabilim Dalı: Yazılım Mühendisliği**

**Danışman: Doç. Dr. Resul DAŞ**

**Tezin Enstitüye Verildiği Tarih: 6 Ocak 2016**

**ŞUBAT – 2016**

T.C.

FIRAT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

# BİLİŞİM SİSTEMLERİ İÇİN SALDIRI TESPİT VE ENGELLEME YAKLAŞIMLARININ TASARIMI VE GERÇEKLEŞTİRİLMESİ

DOKTORA TEZİ

Yük. Müh. Muhammet BAYKARA

(121137202)

Tezin Enstitüye Verildiği Tarih : 6 Ocak 2016

Tezin Savunulduğu Tarih : 10 Şubat 2016

Danışman: Doç. Dr. Resul DAŞ (F.Ü)

Diğer Juri Üyeleri: Prof. Dr. İbrahim TÜRKOĞLU (F.Ü)

Prof. Dr. Abdulkadir ŞENGÜR (F.Ü)

Prof. Dr. Ali KARCI (İ.Ü)

Doç. Dr. Davut HANBAY (İ.Ü)

ŞUBAT – 2016

# ÖNSÖZ

Doktora çalışmalarım süresince değerli fikirlerini benimle paylaşan, yardımlarını esirgemeyen, her koşulda ve konuda kendisine danışabildiğim saygıdeğer danışman hocam, Fırat Üniversitesi Yazılım Mühendisliği Bölümü Öğretim Üyesi Sayın Doç. Dr. Resul DAŞ'a teşekkürlerimi sunarım.

Bu tez çalışmasını TEKF.15.04 numaralı proje ile destekleyen Fırat Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi'ne (FÜBAP) teşekkür ederim. Ayrıca yoğun çalışma temposundan dolayı kendilerine yeterince vakit ayıramadığım aileme anlayış ve desteklerinden dolayı teşekkür ve şükranlarımı sunarım.

**Muhammet BAYKARA**

**ELAZIĞ - 2016**

# İÇİNDEKİLER

**Sayfa No**

# ÖZET

Doktora Tezi

## BİLİŞİM SİSTEMLERİ İÇİN SALDIRI TESPİT VE ENGELLEME YAKLAŞIMLARININ TASARIMI VE GERÇEKLEŞTİRİLMESİ

Muhammet BAYKARA

Fırat Üniversitesi
Fen Bilimleri Enstitüsü
Yazılım Mühendisliği Anabilim Dalı
2016, Sayfa: 205

Bilişim dünyasında, kişisel ve kurumsal anlamda önem arz eden dijital bilgilerin saklanması ve gerektiğinde bu bilgilere güvenli ve hızlı bir biçimde tekrar erişilmesi oldukça önemli bir hale gelmiştir. İstenmeyen kişiler tarafından önemli bilgilere erişilmesinin engellenmesi çok önemlidir. Bilgi mahremiyetinin sağlanabilmesi için kurum veya kuruluşlar önemli verilerini güvenli bir şekilde korumak ve önlem almak zorundadırlar. Saldırı tespit ve engelleme sistemleri de bu önlemler arasındadır. Bu tez çalışmasında Saldırı Tespit Sistemleri (STS) ve Saldırı Engelleme Sistemleri (SES) detaylı bir biçimde araştırılmış ve günümüze kadar bu konu ile ilgili yapılan çalışmalar derinlemesine incelenmiştir. Güncel literatür araştırmaları ışığında, bu tez çalışması kapsamında, honeypot temelli sistemler STS yapıları ile birlikte kullanılarak yeni hibrit bir yaklaşım önerilmiştir. Yüksek ve düşük etkileşimli honeypot sistemlerinin avantajlarını birleştiren bu hibrit yaklaşım ile STS yapılarının en büyük dezavantajlarından biri olan yanlış alarm seviyesinin düşürülmesi sağlanmıştır. Geliştirilen honeypot temelli saldırı tespit ve engelleme mimarisinin özellikle VLAN ağlarını da içeren yerel alan ağlarında en az maliyet ve en yüksek performansı sağlayabilecek şekilde kurulabilmesi için özgün bir yazılımsal anahtar tasarımı gerçekleştirilmiş ve böylece merkezi bir kontrol sağlanmıştır. Ayrıca STS'lerin saldırı analizi yaparken 5651 sayılı kanuna uygun olarak kendi log dosyalarının güvenli bir şekilde kaydedilmesi ve yetkisiz erişim ya da değiştirmelere karşı korunmasına yönelik özgün bir sistem geliştirilmiştir. Geliştirilen sistemle adli bilişim sürecinde loglara objektif kanıt niteliği kazandırılmıştır ve log dosyalarının değiştirilmemesi garanti altına alınmıştır. Gerçekleştirilen saldırı tespit ve engelleme sistemi yazılımları, bulut bilişim çevrelerinde kullanılabilecek bir mimari yapı ile geliştirilmiş ve özel

bulut ağları üzerinde başarılı bir şekilde uygulanmıştır. Özellikle honeypot sistemler temel alınarak gerçekleştirilen bu STS/SES yapıları, saldırıları paket/tür ve lokasyon bazlı olarak tespit edebildiği için oldukça yararlı; kişi, kurum ve kuruluşların güvenlik ihtiyaçlarına cevap verebilecek nitelik ve kapsamdadır. Gerçekleştirilen uygulamaların kurumsal kampüs ağlarında kullanımı incelenerek farklı konumlandırma yöntemleri araştırılıp uygulanmış ve böylece kurulum, bakım ve yönetim maliyetleri düşürülmüştür. Honeypot sistemler, ağ trafiğinin ve akışının haritalandırılması için STS'ler ile birlikte uygulanarak özellikle kurumsal kampüs ağlarını içeren büyük ağ yapılarında özgün bir güvenlik mimarisi sağlanmıştır.

**Anahtar Kelimeler:** Saldırı Tespit Sistemleri, Saldırı Engelleme Sistemleri, Siber Saldırılar, Honeypot Sistemler, Log Güvenliği, Bilgi Güvenliği, Ağ Güvenliği.

## SUMMARY

PhD Thesis

**Design and Implementation of Intrusion Detection and Prevention Approaches for Information Systems**

Muhammet BAYKARA

Fırat University

Graduate School of Natural and Applied Sciences

Department of Software Engineering

2016, Page: 205

In the IT world, on individual and institutional sense, fast and secure retrieval of stored digital information has become very essential. It's important here to prevent unauthorized access of vital stored information. It is an obligation for institutions and corporates to take precaution for ensuring the privacy of confidential of information. One of the practiced precautions is the Intrusion Detection and Prevention Systems (IDPS). In this thesis, IDPSs have been thoroughly investigated and the literature up-to-date on this topic has been examined. In this study, based on our research investigation, we propose a hybrid approach which combines honeypots and IDPSs. The proposed hybrid approach combines the advantages of high and low interaction honeypot system to overcome the biggest drawback, the false alarms, of IDPS. The developed honeypot-based IDPS architecture enables a central control by a special designed software switch to provide least cost and highest performance in a LAN that deploys VLAN. Besides, a special software is developed for ensuring the log files security. This software is adaptable for the Act 5651 in Turkish law and prevents the unauthorized access or change in the log files. The developed system guarantees the integrity of log files so that they can be used as objective evidence for digital forensics process. To verify the usability of the developed system, the system has been successfully tested in a cloud computing environment on a private networks. In particular, honeypot-based system combined with IDPS detects malicious activities based on

package, type and location; thus, it is very useful for both individuals and institutions as response for security needs. After testing the developed system, on different location scenarios, it is recommended to apply the developed system for enterprise campus networks using different localization which in turns reduces the installation, maintenance and management costs. Honeypot systems implemented with IDPS and applied for network traffic mapping create a special security architecture for large enterprise campus network and cloud computing environment.

**Key Words:** Intrusion Detection System, Intrusion Prevention System, Cyber Attacks, Honeypot Systems, Log Security, Information Security, Network Security.

# ŞEKİLLER LİSTESİ

**Sayfa No**

# TABLOLAR LİSTESİ

**Sayfa No**

## KISALTMALAR LİSTESİ

**IDS** : Intrusion Detection System

**IPS** : Intrusion Prevention System

**IDPS** : Intrusion Detection and Prevention System

**DoS** : Denial of Service

**DDoS** : Distributed Denial of Service

**STS** : Saldırı Tespit Sistemi

**SES** : Saldırı Engelleme Sistemi

**IPDS** : Intrusion Prevention and Detection System

**NIDS** : Network Based Intrusion Detection System

**R2L** : Yönetici Hesabı ile Yerel Oturum Açma (Remote to Local)

**U2R** : Kullanıcı Hesabının Yönetici Hesabına Yükseltilmesi (User to root)

**DCA** : Dendritic Cell Algorithm

**TLR** : Toll-Like Receptor Algorithm

**FTP** : File Transfer Protocol

**RDP** : Remote Desktop Protocol

**CPU** : Central Processing Unit

**VLAN** : Virtual Local Area Network

**VoIP** : Voice over IP

**IPv4** : Internet Protocol Version 4

**IPv6** : Internet Protocol Version 6

**URL** : Uniform Resource Locator

**DMZ** : DeMilitarized Zone

**TCP/IP** : Transmission Control Protocol / Internet Protocol

**OSI** : Open Systems Interconnection

**UDP** : User Datagram Protocol

**ACK** : Acknowledgement

**SYN** : Synchronize

**RFC** : Request for Comments

**POP** : Post Office Protocol

**DNS** : Domain Name System

**DHCP** : Dynamic Host Configuration Protocol

**ARP** : Address Resolution Protocol

**SNMP** : Simple Network Management Protocol

**SMTP** : Simple Mail Transfer Protocol

**FTP** : File Transfer Protocol

**HTTP** : Hyper Text Transfer Protocol

**ICMP** : Internet Control Message Protocol

**HTML** : Hyper Text Markup Language

**OS** : Operating System

**MAC** : Media Access Control

**MITM** : Man In The Middle

**TTL** : Time to Live

**MTA** : Mail Transfer Agent

**YSA** : Yapay Sinir Ağları

**NSA** : National Security Agency

**BGYS** : Bilgi Güvenliği Yönetim Sistemi

**SSL** : Secure Socket Layer

**FTP** : File Transfer Protocol

**SSH** : Secure Shell

**IRC** : Internet Relay Chat

**ICSI** : International Computer Science Institute

**GPL** : General Public License

**OISF** : Open Information Security Foundation

**GUI** : Graphical User Interface

**FIM** : File Integrity Monitoring

**XSS** : Cross-Site Scripting

**CSRF** : Cross-Site Request Forgery

**SQL** : Structured Query Language

**ITU** : International Telecommunication Union

**NAT** : Network Address Translation

**PAT** : Port Address Translation

**GNS3** : Graphical Network Simulator-3

**SaaS** : Software as a Service

**PaaS** : Platform as a Service

**IaaS** : Infrastructure as a Service

**ERP** : Enterprise Resource Planning

**CRM** : Customer Relationship Management

**SLA** : Service Level Agreement

**IDCC** : Intrusion Detection in Cloud Computing

**SPK** : Sermaye Piyasası Kurulu

**BDDK** : Bankacılık Düzenleme ve Denetleme Kurumu

**SIEM** : Security Information and Event Management

**TLS** : Transport Layer Security

**DARPA** : Defence Advanced Research Projects Agency

**IDEVAL** : Intrusion Detection Evaluation

**UPS** : Uninterruptible Power Supply

**VM** : Virtual Machine

# 1. GİRİŞ

Bilgi, tarihin her döneminde önemli bir kavram olmuştur. Güncel bilgiye sahip olan ve bu bilgilerini iyi organize eden milletler tarihin başarılı medeniyetlerini inşa etmişlerdir. Buradan da anlaşılabileceği gibi bilgi son derece önemli bir kavramdır ve günümüz dünyasında da önemi hızla artmaktadır. Literatürde bilgi kavramına yönelik çeşitli tanımlar olmakla birlikte bu tanımlardan elde edilebilecek öz, şu şekilde verilebilir: "Bilgi, anlaşılabilen ve iletilebilen veriler topluluğudur. Öyle ki bu veriler kâğıt üzerine yazılmış olabilir, veri tabanlarında saklı bulunabilir veya İnternet vb. üzerinde olabilir" [1]. Çalışmalarda ortak vurgu yapılan bir başka nokta şöyledir: "Bilgi, zihnin herhangi bir biçimde resmi veya gayriresmî olarak iletilen, kaydedilen, yayınlanan fikirlerin gerçek ve hayalî ürünleridir" [2]. Bilgi, en kısıtlı teknik anlamıyla, yorumlanabilir bir semboller dizisinden oluşan bir mesaj olarak tanımlanmaktadır. Öyle ki bu bilgi, bir takım işaretlerle kaydedilebilmekte ve sinyal olarak iletilebilmektedir. Bilgi, bilgiyi yorumlama yeteneğine sahip olan bir dinamik sistemin herhangi bir durumunu ifade etmek için kullanılan bir tanımlamadır [2, 3].

Bilgi, bütün toplumsal alanlarda önemli bir olgu haline gelmiştir. Gerek sosyal ve ekonomik gerekse teknolojik alanlar olsun hayatımızın her kademesinde ve bütün sektörlerde kendisini ve önemini kabul ettirmiştir. Bilginin bu önemine binaen devletler bazında ulusal bilgi politikalarının geliştirilmesi günümüz dünyasında vazgeçilmez bir unsur olmuştur. Bu politikalarla, bilgiyi kullanan kişilerden, bilgiye yön veren uzmanlardan, bilgi teknolojilerinden, bilginin maliyetinden bu bilgi politikalarını oluşturacak kuruma kadar birçok farklı alanda düzenlemeler yapılmaktadır [2]. Günümüzde bilgi çağı olarak adlandırılan ve teknolojik gelişmelerin artan ivme ile gerçekleştiği bir dünyada bilgi kavramı daha da fazla önem kazanmaktadır. John Naisbitt, "Yeni güç kaynağı azınlığın elindeki para değil, çoğunluğun elindeki bilgidir" diyerek bilginin önemini ortaya koymuştur [4]. Bilgi çağını yaşayan günümüz toplumları ve bu toplumların bireyleri açısından bakıldığında bilgi, önemli bir anahtar olarak işlev görmektedir. Çünkü bilgi, toplumlar ve bireyler açısından kalkınmışlığın göstergesi olarak nitelendirilen bir ölçüt halini almıştır.

Bir ülkenin gelişmişlik düzeyi, eskiden ürettiği çelik ve enerji miktarı ile ölçülürken artık; bilgi teknolojisini oluşturan mikro elektronik, telekomünikasyon ve bilgisayar

teknolojilerinin imkânları ile elde edilen, işlenen, iletilen, saklanan bilgi miktarı ile ölçülmeye başlanmıştır [5].

Yukarıda verilen tanımlar sonucunda bilgi politikaları ile ilgili temel unsurlar şunlardır;

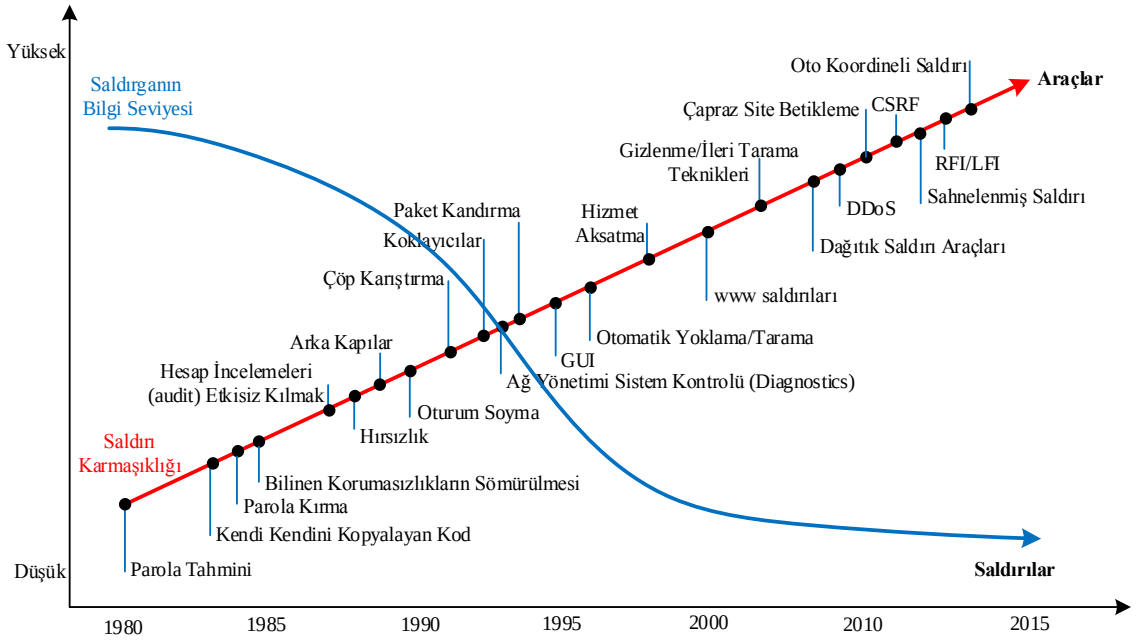
- Bilgi kullanıcıları,
- Bilgi mevzuatı,
- Bilgi ile ilgili insan gücü,
- Bilgi teknolojileri,
- Bilginin maliyeti,
- Bilgi ile ilgili örgütlenme ve organizasyonlardır.

Bilgi kavramının önemi anlaşıldıkça bilginin bir varlık olarak korunmasının ve güvenliğinin sağlanmasının önemi de açıkça ortaya çıkmaktadır. Günümüz bilgi çağında gerek kişisel gerekse kurumsal bilgiler çok büyük oranda dijital ortamlarda saklanmaktadır. Bilgilerin kâğıtlar üzerinde arşivlendiği ve saklandığı yapılarda bilgi güvenliği, bilgilerin fiziksel olarak korunmasından ibaret sayılabilir. Ancak günümüz dünyasında bilgisayar sistemlerinde, dijital ortamlarda saklanan bilgilerin korunması, fiziksel olarak korunmanın yanı sıra dijital olarak birtakım önlemlerin alınması açısından birçok sorumluluğu da beraberinde getirmektedir. Tüm bu sorumluluklar, bilginin korunmasına yönelik yöntem, yaklaşım ve eylemler kısaca *bilgi güvenliği* olarak anlandırılmaktadır.

Bilgi güvenliği, bilgilerin, izinsiz erişimlerden, kullanımından, ifşa edilmesinden, yok edilmesinden, değiştirilmesinden veya hasar verilmesinden korunma işlemidir. Bilgi güvenliği, bilgisayar güvenliği ve bilgi sigortası terimleri, sıkça birbirinin yerine kullanılmaktadır. Bu alanlar birbirleri ile alakalıdırlar ve kişisel veya kurumsal mahremiyetin, bütünlüğün ve bilginin ulaşılabilirliğinin korunması hususunda ortak amaçlara sahiptirler. İnternetin yaygın olarak kullanılmaya başlanmasıyla birlikte bilişim sistemlerindeki güvenlik açıkları da artmaya başlamıştır. Bilişim sistemlerinde, gizlilik, bütünlük ve sürekliliğin sağlanması için birçok güvenlik ürünü ve projesi geliştirilmiştir ve hâlâ geliştirilmektedir [6].

Bilgisayar ağ sistemlerinin yaygın olarak kullanılmaya başlanmasıyla birlikte ağ üzerinden yapılan saldırılarda da artışlar yaşanmaya başlanmıştır. Saldırı olaylarının bir kısmına bilinçsiz kullanıcılar neden olurken bir kısmına da bilerek sisteme zarar vermek isteyen kötü niyetli kişiler neden olmaktadır. 1980'li yıllarda bilgisayar haberleşmelerinde

TCP/IP protokol kümesi, dünya çapında kabul görmüş ve İnternet bu protokol aracılığı ile yaygınlaşmıştır. İnternetin yaygınlaşması ile bilgisayar haberleşmelerindeki saldırıların sayısı ve çeşidi de artmıştır [6, 8]. Bu saldırılar karşısında kimlik doğrulama, yetkilendirme, antivirüs programları gibi güvenlik çözümleri geliştirilmeye çalışılmıştır. İlk çıkan saldırılar daha çok basit kod yürütme, parola tahminleri gibi etkisi ve olasılığı düşük saldırılar olmasına karşın süreç içerisinde saldırıların karmaşıklığı ve etkileri artmıştır. Buna rağmen bu saldırıları kullanabilmek için gereken bilgi düzeyi düşmüştür. Çünkü bu bilgiler çoğunlukla İnternet üzerinden kontrolsüz olarak yayılmıştır. Şekil 1.1'de saldırıların zamana göre değişimleri, etkileri ve onları kullanabilmek için gerekli bilgi düzeyi görülmektedir [6, 8].




**Şekil 1.1** Saldırıların zamana göre değişimleri

Bu tez çalışması, bilgi politika unsurlarından "bilgi teknolojileri" kapsamında değerlendirilebilecek olan ve bilgi sistemleri güvenliğinin sağlanmasına yönelik bir yapı olarak kullanılan Saldırı Tespit Sistemleri (IDS-STS) ve Saldırı Engelleme Sistemleri (IPS-SES)'nin incelenmesi ve bu sistemlerin detaylı analizlerini içermektedir. Ayrıca günümüzün güncel ve yoğun trafiğe sahip dijital sistem ve teknolojileri için önerilen hibrit çözümleri, yüksek başarımda üretebilmek açısından yeni teknikler kullanan çözümler sunmaktadır.

## 1.1 Amaç

Bilişim sistemlerinde bilgi ve bilgi güvenliği, günümüzde oldukça önemli bir konudur. Sanal ortamdaki güvenliğin sağlanmasında saldırı tespit ve engelleme sistemleri büyük bir

rol oynamaktadır. Dünyada ve ülkemizde kişi, kurum ve kuruluşlar açısından da IDS/IPS yapıları çok önemlidir.

Bu doktora tez çalışmasının amacı, bilişim sistemlerinde kullanılan gerçek zamanlı saldırı tespit ve engelleme sistemleri için etkili ve başarımı yüksek yeni yaklaşımlar geliştirmek ve gerçekleştirmektir. Bu amaç doğrultusunda;

1. Web tabanlı saldırılar için gerçek zamanlı *honeypot temelli* hibrit bir saldırı tespit ve engelleme yaklaşımı geliştirilmiştir.
2. Saldırı tespit ve engelleme sistemleri için *honeypot temelli* yeni bir yaklaşım geliştirilmiş, kamu kurum ve kuruluşları için bu yapı önerilmiştir.
3. Bilişim sistemleri için vazgeçilmez olan log dosyalarının yetkisiz kişiler tarafından değiştirilmemesinin garanti edilmesi için yeni bir kriptoğrafik log güvenliği yazılım aracı geliştirilmiştir.
4. VLAN içeren büyük ölçekli kurumsal yerleşke ağlarının iç tehditlerden korunması, merkezi kontrol yapılabilmesi ve yanlış alarm seviyesinin düşürülebilmesi için yazılımsal switch yaklaşımı geliştirilmiştir.
5. Geliştirilen ve önerilen dört farklı yaklaşımın özel bulut ağları üzerinde kullanılabilmesi sağlanmıştır.

## 1.2 Tezin Organizasyonu ve Katkılar

Tez çalışmasının birinci bölümünde, teze genel bir bakış açısı kazandırılmaya yönelik temel bilgiler verilmiştir. Diğer bölümlerin organizasyonu ile birlikte orijinal katkılar aşağıda sunulmaktadır:

***Bölüm 2***'de, ağ güvenliği ve saldırı türleri konusu detaylı bir şekilde ele alınmış, literatürde geçen alt bölümleri ile birlikte açıklanmıştır. Özellikle tez çalışmasında temel çalışma alanı olarak belirlenen bilgi güvenliği açısından ağ iletişimi ve protokollerinin olası açıklıkları üzerinde durulmuştur.

***Bölüm 3***'te, bilgi güvenliği temel ilkeleri, saldırı tespit ve engelleme sistemleri konusu detaylı bir şekilde ele alınmıştır. Bu bölümde ayrıca mevcut saldırı tespit ve engelleme yazılımları ile STS'lerde kullanılan veri kümeleri hakkında incelemeler sunulmuştur.

***Bölüm 4***'te web tabanlı saldırılar için gerçek zamanlı *honeypot temelli* hibrit bir saldırı tespit ve engelleme yaklaşımı geliştirilmiştir. Bu kısımda, literatürde geçen ve en çok bilinen web açıklıkları üzerinde durulmuş ve bu açıklıkların tespitine yönelik yazılımlar gerçekleştirilmiştir.

***Bölüm 5***'te, saldırı tespit ve engelleme sistemlerinin yanlış alarm seviyesinin düşürülmesi ve böylece başarımın arttırılabilmesi için düşük ve yüksek etkileşimli honeypot sistemler, saldırı tespit ve engelleme sistemleri ile birlikte kullanılarak yeni, başarılı bir yaklaşım sunulmuştur. Ayrıca bu bölümde, honeypot sistemlerin kurumsal kampüs ağlarında kullanımı incelenerek kurulum, bakım ve yönetim maliyetlerinin düşürülmesi sağlanmıştır. Honeypot sistemlerin STS'ler ile birlikte kullanılması sayesinde, özellikle anormallik tespiti yönteminin dezavantajlarından biri olan yanlış alarm seviyesinin düşürülmesi sağlanmıştır. Honeypot sunucularda oluşan ağ trafikleri ve veri akışı görselleştirilerek sistem güvenlik yöneticilerine kolaylık sağlanmıştır.

***Bölüm 6***'da, bilgi güvenliği sistemleri için oldukça önemli olan log dosyalarının yetkisiz erişimlere karşı korunmasını garantileyen yeni bir log güvenliği yaklaşımı sunulmuştur. Bu yaklaşım ışığında geliştirilen kriptoğrafik log güvenliği yazılımı, olası adli olaylarda kurumsal ve kişisel anlamda logların değişmediğini ve delil olarak kullanılacak dijital bilgilerin bütünlüğünün korunduğunu garanti eden bir yapı sunmaktadır. Böylece, adli bilişim sürecinde loglara objektif kanıt niteliği kazandırılmıştır.

***Bölüm 7***'de, özgün bir yazılımsal anahtar tasarımı gerçekleştirilerek, honeypot temelli saldırı tespit ve engelleme sistemlerinin, VLAN içeren ağları da izleyebilmesini sağlamak amacıyla merkezi bir kontrol yaklaşımı sunulmuştur. Gerçekleştirilen uygulama ile VLAN ağlarına sahip kurumsal ağlarda her bir VLAN ağının merkezi bir noktadan izlenebilmesi sağlanmıştır. Honeypot sistemler ile STS yapılarının kolaylıkla etkileşim kurabilmesini sağlayan yazılımsal anahtar uygulaması, konfigürasyon, bakım, ağ izleme ve ağ analizi faaliyetlerini kolaylaştırmaktadır.

***Bölüm 8***'de, tez çalışmaları kapsamında gerçekleştirilen ve önceki bölümlerde verilen yeni yaklaşımlar, özel bulut ağlarının güvenliğini sağlamak amacıyla Muxer adı verilen bir uygulama ile bulut sistemlerde kullanılabilir hale getirilmiş ve başarıyla test edilip denenmiştir.

***Bölüm 9***'da, tezin sonuçları irdelenmiş ve orijinal katkılar vurgulanmıştır. Ayrıca gelecek çalışmalar ve öneriler tartışılmıştır.

Bu tez çalışması, Fırat Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi (FÜBAP) tarafından TEKF.15.04 numaralı proje ile desteklenmiştir.

## 1.3 Literatür Taraması ve Değerlendirilmesi

Bilgi, herhangi bir konu ile ilgili belirsizliği azaltan, sürekli gelişen, değişen ve yenilenen bir varlıktır. Bilgi, saygınlık, farkındalık ve ayrıcalık oluşturan bir araç haline geldiğinden dolayı yüksek değere sahiptir. Bu sebeple sahip olunan bu varlık, elektronik ortamlarda her türlü tehdit ve tehlikelere karşı korunmalıdır. İhtiyacımız olmayan bir bilgi bizim için önemli değilken, başkaları için hayati önem taşıyabilir. Yani bilginin değeri, kullanılacağı yere veya duruma göre değişiklik arz edebilir. Bu yüzden bilginin gizliliği ve bütünlüğü korunmalı, önemli bilgiler yetkisiz kişilerin ellerine geçmemelidir. Yaşadığımız dijital bilgi çağında, e-devlet, e-imza, e-ticaret gibi kavramlar ve bunlarla ilgili güvenlik çalışmalarından oldukça sık bahsedilmektedir. Bilginin elektronik ortamlara aktarılmasıyla birlikte, hız ve verimlilik artmış, iş gücü-zaman açısından kazanımlar ve kolaylık sağlanmıştır. Fakat kişisel veya kurumsal açılardan önemli olan bilgilerin, istenmeyen kötü niyetli kişilerin eline geçmesi, maddi ve manevi zararlara yol açabilmektedir. Bu sebeple dijital ortamlardaki bilgi kullanımı artışına paralel olarak, "bilgisayar güvenliği" kavramının da önemi artmıştır. Kurumlar artık giderek daha fazla oranda bilgi güvenliğine verdikleri önemi öncelikli hale getirmektedir [5].

Son yıllarda e-devlet kapsamındaki yazılım projelerinde güvenliğin en üst düzeyde sağlanması ulusal bir amaç haline gelmiştir. Konuyla ilgili olarak birçok hukuki ve teknolojik önlemler geliştirilmiş ve hâlen geliştirilmeye devam edilmektedir. Günümüz bilişim dünyasının en ehemmiyetli alanlarından biri olan bu alanda hızlı bir gelişim söz konusudur [5-8].

Gelişen teknoloji ile birlikte bilişim dünyasındaki ilerleme seviyesi, ülkeler arası ilişkilerde yeni bir güç unsuru haline gelmiştir. Ülkeler bazında askeri, ekonomik vb. güç unsuru olan kavramlar arasına günümüzde "siber uzay", "siber güç" ya da "siber ordu" olarak adlandırılan yeni bir kavram eklenmiştir. Artık fiziksel, sıcak savaş ve saldırılardan farklı olarak siber saldırılar da gerçekleştirilmektedir. Bu sebeple ülkeler artık askeri

ordularının yanı sıra siber ordularını kurmakta ve siber uzay dediğimiz yapıda saldırılar gerçekleştirmektedir.

Dijital ortamlarda saklanan bilginin her türlü saldırı ve tehditlere karşı korunması için güvenlik duvarları, antivirüs yazılımları, saldırı tespit ve engelleme sistemleri gibi araçlar geliştirilmiştir. Günümüzde bilgi ve bilgisayar güvenliğinin öneminin kavranmasıyla geliştirilen araçlardan biri olan STS'ler, saldırılara karşı bilişim sistemlerinde "alarm" niteliği taşıyan yazılım ve donanımlardır. STS'lerin kullanılması ile sistemlere yapılan yetkisiz erişimler ve kötüye kullanımlar tespit edilerek, bunların yol açabileceği zararlar engellenmiş olur. Bilgisayar sistemlerinde STS'lerin kullanılması ile birlikte, sisteme ne tür saldırıların daha çok yapıldığı, sistemdeki mevcut açıklar/zafiyetler ve saldırganlar hakkında daha detaylı bilgiler elde edilebilmektedir.

İlk olarak 1980'de Anderson'un yaptığı çalışmalar sonucunda ortaya çıkan STS'ler, ardından yapılan birçok çalışma ile hızla gelişmesini devam ettirmiştir [7]. Denetim verilerinin STS'lerde kullanılmaya başlanması ile birlikte, denetim verilerinin elde edilmesi ve kullanılması ile ilgili çalışmalar artmıştır. 1988 yılında geliştirilen IDES (Intrusion Detection Expert System-Saldırı Tespiti Uzman Sistemi), o yıla kadar yapılan birçok çalışmayı üzerinde toplaması açısından en önemli STS çalışmalarından biridir [8]. İstatistiksel yaklaşımların dışında yine o yıllarda, kural tabanlı, eşik değeri belirleme (threshold value), durum geçiş diyagramları (state transition diagrams), veri madenciliği gibi metotların da kullanıldığı görülmektedir. Ancak teknolojinin hızlı gelişimi ve saldırganların bu gelişimden faydalanarak eskiye oranla daha az bilgi ve tecrübe sahibi olmalarına rağmen daha etkili ve hızlı saldırılar gerçekleştirmeleri, güvenlik boyutunu dinamikleştirmiş ve sürekliliği zorunlu kılmıştır. Bu nedenle, STS'lerin tarihsel gelişimi sürecinde akıllı STS'lere ihtiyaç duyularak, Yapay Sinir Ağları (YSA), Yapay Bağışıklık Sistemi, Bulanık Mantık gibi akıllı-öğrenen teknikler de kullanılmaya başlanmıştır [8-10]. 1990'ların başında kullanılmaya başlanan ve ön plana çıkan akıllı sistemlerin kullanımı ile STS'lerin başarı oranlarının arttığı gözlenmiştir. Özellikle, önceden bilinmeyen yeni saldırıların tespit edilebilmesi için kullanılan anormallik tespiti yaklaşımında akıllı tekniklerin kullanılması, başarı oranının artmasında en büyük etkenlerden biri olmuştur.

Dilimize bal küpü olarak yerleşen honeypot sistemler de son yıllarda saldırı tespit ve engelleme sistemleri ile birlikte sıklıkla kullanılan güvenlik araçları haline gelmiştir.

Honeypot sistemler, saldırı tespit sistemleri ya da güvenlik duvarları gibi doğrudan özel bir güvenlik probleminin çözümünde kullanılmazlar. Honeypotlar, güvenlik sistemlerinin bir parçası olarak kullanılırlar ve hangi tür problemlere çözüm sunacakları tasarım veya kullanım şekilleriyle bağlantılıdır [11-16]. Bu sebeple diğer bilgi güvenliği araçlarının aksine her problemin çözümüne genel cevap veren bir honeypot sisteminden söz edilemez [15, 16]. Literatürde honeypot sistemler ile saldırı tespit ve engelleme sistemleri (IDPS) gibi güvenlik uygulamalarının birlikte kullanıldığı çeşitli uygulamalar mevcuttur [15-20].

Gökırmak ve diğ. yaptıkları çalışmada, IPv6 ağlarında gerçekleşen saldırıları tespit edebilmek için honeypot temelli bir yazılım geliştirmişlerdir. Servis, işletim sistemi ve ağ seviyesinde izleme yeteneğinde olan bu yazılımla, IPv6 protokolündeki güvenlik sorunlarının saptanabilmesi amaçlanmıştır [15, 16].

Malanik ve diğ. yaptıkları çalışmada, honeypot sistemlerin yerel alan ağlarındaki olası uygulamaları üzerinde durmuşlardır. Yapılan çalışmada, honeypotların bir bilgisayar ağında LAN, İnternet ve DMZ bölgesi olmak üzere üç farklı şekilde konumlandırılabileceği belirtilmiştir. Honeypot sistemlerin STS'ler ile birlikte kullanılmaya başlanmasıyla bu hibrit sistemlerin sıfır-gün kötücül aktivitelerini tespit edebileceği vurgulanmıştır. Yine bu çalışmaya göre ağda tanımlı VLAN ağları bulunması durumunda, honeypot sistemlerin her VLAN içerisinde yapılandırılması gerektiği belirtilmiştir [21].

Riboldi ve diğ. yaptıkları çalışmada, VoIP sistemler üzerinde illegal aktivitelerin izlenebilmesi için düşük etkileşimli bir honeypot sistem kodlamışlardır. 92 gün boyunca performansı incelenen sistem üzerinde SIP protokolü ile ilişkili 3502 olay toplanmıştır. Gerçekleştirdikleri sistemi, VoIP çevrelerinde güvenlik duvarı ve saldırı tespit sistemi şeklinde kullanılabilir olarak yorumlamışlardır [22].

Shukla ve diğ. yaptıkları çalışmada kötücül web URL'lerinin tespiti için bir honeypot sistem geliştirmişlerdir. Python dili ile geliştirilen sistem, istemci tarafında hizmet vermektedir. Geliştirilen sistemde istemci tarafındaki arama motoru örümcekleri ile URL adresleri toplanır. Daha sonra ziyaret ihtiyacı olduğunda ilgili web siteleri ziyaret edilir. Eğer bu URL adresleri kötücül ise veya açıklık içeriyorsa imza temelli saldırı tespit sistemi tarafından bir tetikleyici aktifleştirilir. Böylece zararlı URL adresleri kara listeye alınarak güvenlik sağlanmış olur [23].

Koniaris ve diğ. yaptıkları çalışmada kötücül aktivite ve bağlantıların analizi ve görselleştirilmesi için honeypot sistemlere başvurmuşlardır. Geliştirdikleri uygulamada iki ayrı araştırma honeypotu kurmuşlardır. Bunlardan ilki genellikle kendi kendine yayılım özelliği olan kötücül yazılımları toplamak için, ikincisi ise kötücül aktiviteleri toplayan tuzak sistem olarak tasarlanmıştır [24].

Song Li ve diğ. yaptıkları çalışmada karışık etkileşimli honeypot temelli bir saldırı tespit sistemi mekanizmasının kurulması üzerinde durmuşlardır. Kurdukları sistemin amacını, ağın dayanıklılığını ve güvenliğini arttırmak olarak tanımlamışlardır. Ağ güvenliğinin arttırılması için honeypot sistemin tuzak kapasitesini yükselterek çeşitli incelemeler gerçekleştirmişlerdir [25].

Chawda ve diğ. yaptıkları çalışmada yeni zafiyet ve açıklıkların incelenmesi için dağıtık bir honeypot sistemi önermişlerdir. Geliştirdikleri sistemde zafiyet ve açıklıklara daha fazla maruz kalınması için başlangıç aşaması (önyüz) içerik filtresi olarak düşük etkileşimli honeypot sistemlerini kullanmışlardır [26].

Xiangfeng Suo ve diğ. yaptıkları çalışmada saldırı tespit sistemleri içerisinde honeypot teknolojilerinin nasıl uygulanacağı üzerinde durmuşlardır. Yapılan çalışma, saldırı tespit sistemlerinin sahip olduğu problemleri giderebilmek amacıyla honeypot sistemlerin kullanılabileceğini öneri olarak sunmaktadır [27].

Paul ve diğ. yaptıkları çalışmada bilgisayar ağ güvenliği için honeypot temelli bir imza üreteci geliştirmişlerdir. Geliştirilen sistem özellikle polimorfik solucan saldırılarına karşı koruma amaçlı olarak kullanılmıştır. Geliştirilen sistem, şüpheli trafiği izole edip kötücül trafiği ve solucan saldırıları ile ilgili birçok yararlı bilgiyi toplayabilme yeteneğine sahiptir. İmza temelli sistemlerin yeni saldırıların tespitinde çalışmamasına karşılık önerilen sistem, bilinmeyen solucan saldırıları için imza üretme özelliğine sahiptir [28].

Beham ve diğ. yaptıkları çalışmada sanallaştırma teknolojilerinin avantajlarından yararlanmışlardır. Çalışmada saldırı tespit ve honeypot sistemlerinin iç içe sanallaştırma çevrelerindeki kullanımları incelenmiştir. Çalışmada mevcut iç içe sanallaştırma teknolojileri ile VMI tabanlı saldırı tespit ve honeypot sistemleri karşılaştırmalı bir biçimde incelenmiştir [29].

Liu ve diğ. yaptıkları çalışmada IP traceback tekniğini kullanan honeypot teknolojisi temelli bir saldırı tespit sistemi geliştirmişlerdir. Çalışmada geleneksel saldırı tespit sistemlerinin limitleri ortaya konularak honeypot sistemler üzerinde bir saldırı tespit yapısı önerilmiştir [30].

Auttapon Pomsathit, yaptığı çalışmada dağıtık ağlar üzerinde honeypot sistemler ile saldırı tespit sistemlerinin kullanımını ele almıştır. Çalışmasındaki temel amacı, saldırı tespit sistemi ve honeypotlar ile birlikte kullanılan saldırı tespit sistemlerinin etkinliğinin ölçülmesi olarak açıklamıştır [31].

Jiang ve diğ. yaptıkları çalışmada kurumsal iş ağları için honeypot sistemlerin uygulanmasını işlemişlerdir. Mevcut honeypot sistemlerini inceleyerek saldırı tespit sistemlerinde kullanılan yöntemleri yeni bir honeypot sistem ile birleştirmişlerdir [32].

Mitsuaki Akiyama ve diğ. yaptıkları çalışmada yüksek etkileşimli ve etkin performanslı ölçeklendirilebilir bir istemci honeypotu tasarlamışlardır. Bu sayede derinlemesine analiz ve yakalama gücü hedeflenmiştir [33].

Buvaneswari, M. ve diğ. yaptıkları çalışmada dağıtık hizmet engelleme saldırılarını önlemek için *IHoneycol* adını verdikleri bir yaklaşım önermişlerdir. Böylece DDoS saldırılarının honeypot ve saldırı tespit sistemi temelli yapılarla engellenebileceğini ortaya koymuşlardır [34].

Vishal M. ve diğ. yaptıkları çalışmada SNORT, OSSEC ve Honeyd vb. açık kaynak kodlu sistemleri baz alan ve makine öğrenmesi algoritmalarından da yararlanıp saldırı tahmini yapan bir honeypot sistem üzerinde durmuşlardır [35].

Li L. ve diğ. yaptıkları çalışmada sanal ve fiziksel honeypotların, belirli bir lokasyona yerleştirildiği yerel alan ağında honeypot uygulamaları geliştirmişlerdir. Yapılan çalışmada odaklanılan hedef, yerel alan ağlarında güvenliğin arttırılması için güvenlik duvarı, saldırı tespit sistemleri ve honeypot sistemler gibi çeşitli savunma sistemlerinin kombinasyonunun sağlanması olarak açıklanmıştır [36].

Güncel literatür çalışmalarından da görüldüğü gibi bilişim sistemleri ve ağ güvenliğinin sağlanmasında saldırı tespit sistemleri, saldırı engelleme sistemleri, güvenlik duvarları ya da

honeypotlar tek başlarına yeterli olmamaktadır. Konu ile ilgili güncel çalışmalar incelendiğinde bu gibi güvenlik sistemlerinin birbirleriyle etkileşimli bir şekilde kullanıldığı hibrit yapılar önerildiği görülmektedir.

Bu tez çalışması kapsamında, geliştirilen ve önerilen Honeypot IDPS yaklaşımlarında, literatürdeki öneriler dikkate alınarak honeypotlar ile saldırı tespit ve engelleme sistemleri ortak bir biçimde kullanılmıştır. Böylece sistem başarımı arttırılmış, özellikle anormallik temelli saldırı tespit sistemlerinin dezavantajlarından biri olan yanlış alarm seviyesinin düşürülmesi ve bilinmeyen yeni saldırı örüntülerinin tespit edilebilmesi mümkün olmuştur.

Konu ile ilgili literatür incelendiğinde, STS'lerde; veri toplama, etiketleme, depolama, veri azaltma (filtreleme, özellik seçme ve sınıflandırma), davranış modellerinin belirlenmesi ve sınıflandırılması, kural tabanlı sistemler için kuralların belirlenmesi, raporlama ve sonuç üretme aşamalarında güçlüklerle karşılaşıldığı görülmektedir. STS'ler ile ilgili en büyük problemlerden biri, STS'lerin tasarım ve uygulama aşamalarında kullanılabilecek veri tabanlarının eksikliğinden kaynaklanmaktadır.

Ayrıca, STS tasarımı sırasında araştırmacıların farklı veri tabanları geliştirmesi sonucunda, başarı oranlarının objektif olarak değerlendirilememesinden dolayı, literatürde saldırı veri tabanlarının oluşturulmasına yönelik çalışmalar da yapılmıştır. STS'lerin geliştirilmesi ve test edilmesi için kullanılabilecek saldırı veri tabanlarının;

- Geliştirilmesinin maliyetli ve zor olması,
- Tamamen ayrı bir çalışma zamanı gerektirmesi,
- Geliştirilse bile gerçeğe uygunluğunun kabul edilebilir olmasının sağlanamaması,
- Farklı yaklaşım ve tekniklerle kullanılmaya elverişli ve kolay işlenebilir olmasının sağlanması,
- Güncelliğinin korunması, gibi problemlerle karşılaşılmaktadır [8].

STS'lerde karşılaşılan problemlerden bir başkası ise yanlış alarm (false positive) oranlarının düşürülememesidir. İmza (kötüye kullanım) tespiti yapan STS'lerde bu durum ile pek karşılaşılmazken, anormallik tespiti yapan sistemlerde, gerek davranış ya da kullanıcı profillerinin modellenmesi sırasında karşılaşılan güçlükler, gerekse anormallik tespitinin doğası nedeniyle bu durumla çokça karşılaşılır. Aslında bu problemin en aza indirilmesinin yolunun özenle hazırlanmış veri tabanlarından geçtiğini de belirtmekte fayda vardır [8,9].

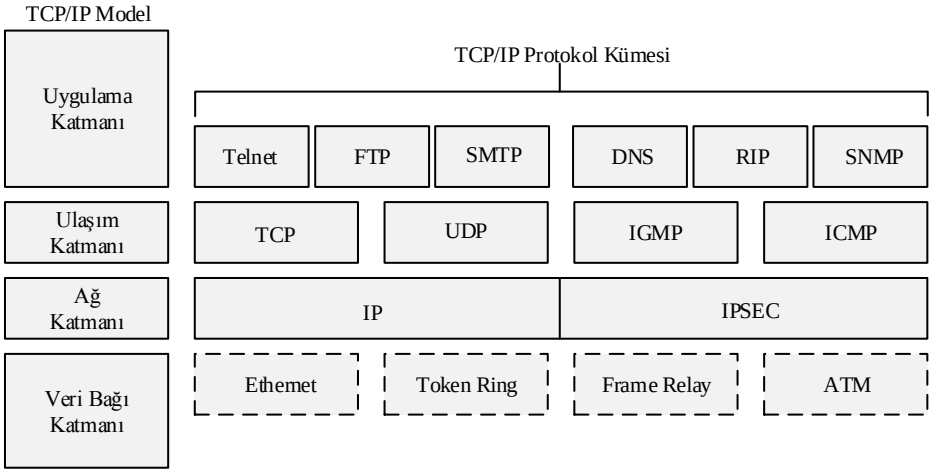
Konu ile ilgili yapılan geçmiş çalışmalar incelendiğinde STS'ler ile ilgili problemlerin ortadan kaldırılması için yeni, gelişmiş STS'lere ihtiyaç duyulduğu belirlenmiştir. Özellikle yanlış alarm durumlarının azaltılması için akıllı-öğrenen sistemler geliştirilmesi bir ihtiyaçtır. Ayrıca STS'ler için güncel saldırı örüntülerini de içeren yeni veri kümelerinin de geliştirilmesi gerektiği açıktır.

# 2. AĞ GÜVENLİĞİ VE SALDIRI TÜRLERİ

## 2.1 Ağ İletişimi ve Protokoller

Protokoller, bilgisayar ağlarında kullanılan tüm aktif ağ cihazlarının birbiri arasında veri iletişimini sağlıklı bir şekilde gerçekleştirmeleri ve veriyi işleyebilmeleri için tanımlanmış kurallardır. Protokollerin tanımında, iletilecek bilginin formatı, iletim şekli, iletim ortamı, bilginin takip edeceği yol vb. gibi bilgiler yer almaktadır. Bu protokoller sayesinde bilgisayar ağlarında, farklı teknolojik donanımlara sahip ağ sistemleri, anlaşabilir, aynı dili konuşabilirler [37].

Ağ ortamında veri iletişimi için kullanılan cihazlar, veriyi genellikle OSI referans modelinin veri bağı katmanı, ağ katmanı ve taşıma katmanlarındaki standartlara ve protokollere göre gönderip almaktadır. Tezin bu bölümünde Şekil 2.1'de verilen, TCP/IP protokol kümesinde yer alan ve en sık kullanılan protokoller incelenmiştir [37]. TCP/IP referans modeli ve protokolleri geliştirildikten sonra, bu protokoldeki bazı noktaların güvenlik açısından istismarından dolayı birçok zafiyet noktası ortaya çıkmıştır [37, 38].




**Şekil 2.1** TCP/IP protokol kümesi

**IPv4 (Internet Protocol Version 4):** İnternet protokolü (IP), İnternet ağının temelini oluşturan TCP/IP protokol kümesinin başlıca protokollerindendir. IP, adresleme ve yönlendirme işlemlerini yapan bağlantısız (connectionless) bir protokoldür. Çünkü veri iletiminden önce alıcı ve gönderici taraf arasında herhangi bir oturum kurulmaz, tarafların anlaşması gerekmez. Bu sebeple verinin iletim garantisi söz konusu değildir. İnternet ortamında bu işlem bir üst katmandaki TCP ile sağlanmaktadır. IP protokolüne göre bilgisayarlar mantıksal olarak adreslenir. Verilen adresler 32 bitten oluşur ve noktalarla

ayrılmış 8'er bitlik oktetler halinde ifade edilir. Örneğin, bir bilgisayarın adresi 192.168.65.123 şeklinde olabilir.

IP, verinin paketler halinde yönlendirilerek iletilmesini sağlar. Oluşturulan veri paketi başlık ve veri kısımlarından oluşur. Başlık kısmında, kaynak bilgisayarın adresi, hedef bilgisayarın adresi ve verinin iletimi için gerekli olan diğer bilgiler tutulur.

IP için istismara açık durumlardan bir tanesi, kaynak IP adresinin farklı gösterilmesiyle karşı tarafın aldatıldığı IP Spoofing (aldatma) işlemidir. Bir bilgisayar ağında o anki iletişimde kullanılmayan IP başlığındaki *Type of Service* gibi bazı alanlar değiştirilerek, asıl verinin yanı sıra paket yapısının içinde farklı verilerin iletimi ile o anki iletişim, veri gizleme amaçlı kullanılabilmektedir [39, 40].

IP paketinin parçalanabilmesi (fragmantasyon), güvenlik açısından riskli durumlara sebep olabilmektedir. Bazı saldırı tespit sistemleri, parçalanmış olarak gelen IP paketinin ilk kısmına bakıp, paket başlık yapısını inceledikten sonra diğer parçalarla ilgilenmeden veriyi iletirler. Bu durumda bir saldırgan, paketi göndermeden önce yardımcı araçlarla paketi parçalayıp, birinci parçadan sonrakilere zararlı komut ve veriler yerleştirerek hedefe gönderebilir [41].

**IPv6 (Internet Protocol Version 6):** IPv6, yeni nesil İnternet yönlendirme protokolüdür. Günümüzde kullanılan IPv4'ün yetersiz kalması ve bazı iyileştirmelerin yapılması ihtiyacına binaen geliştirilmiştir. IPv6 ile beraber IPv4'te bulunan bazı az kullanılan alanlar kaldırılmıştır. IPv6'da dikkate değer ilk değişiklik adresleme alanının genişliğidir. IPv4'de 32 bit olan IP adresleri, IPv6'da 128 bit olmuştur [42]. IPv6 ile gelen yenilikler şöyledir:

- Daha geniş adres uzayı bulunmaktadır.
- Bir ağda birçok hedefe aynı anda veri iletimi (multicasting) yapılabilmektedir.
- Ağ katmanı güvenliği (IPSec protokolü ile) sağlanmıştır.
- Daha basit paket başlığı yapısına sahiptir.
- Geliştirilmiş servis kalitesi ile daha hızlı ses ve video iletimi yapılabilmektedir.
- Daha kolay ve verimli yönlendirme yapılabilmektedir.
- Otomatik adres yapılandırılması mümkündür.

IPv6, standartları RFC 2460 belgesinde tanımlanan, İnternete bağlanacak cihazların adresleme ve iletişimlerini sağlayacak yeni nesil İnternet protokolüdür. IPv6 ile IPv4 yapısına göre birçok yeni özellik getirilmiş, IPv4'ün eksik kaldığı noktalarda çeşitli çözümler üretilmiştir. IPv6 ile gelen bazı özelliklerin açıklamaları aşağıda verilmiştir.

*Genişletilmiş Adres Alanı:* 128 bitlik adres uzunluğu ile IPv4'e göre çok daha fazla sayıda IP adres uzayı sunmaktadır.

*Yeni Güvenlik Özellikleri:* IPv4'te opsiyonel olarak bulunan IPSec desteği IPv6'da bütünleşik olarak gelmektedir. IPv6, güvenlik için ek başlıklar içermektedir. IPv6 yapısında, ara düğümlerde paket yapısının parçalanmasına izin verilmemekte ve yeni başlık yapısı ile ağ üzerinde paketlerin izlenebilmesi kolaylaşmaktadır.

*Sadeleştirilmiş Başlık Yapısı:* IPv6 yapısında, sabit uzunluklu yeni bir başlık yapısı tanımlanmıştır. Ayrıca isteğe bağlı olarak kullanılabilecek *Uzantı Başlıkları* bölümü mevcuttur.

*Gelişmiş Servis Kalitesi:* Ağ trafiğinin daha iyi tanımlanması ve ölçeklendirilmesi mümkündür. IPSec kullanımı, paketlerin önceliklendirilmesini etkilememektedir.

*Otomatik Adres Yapılandırılması*: IPv6 protokolü ile ağ üzerinde herhangi bir adres atama sunucusu olmaksızın, ağa bağlı olan arabirimlerin adres edinebilmeleri mümkündür.

*Genişletilebilirlik:* Genişletilebilir olan uzantı başlıkları bölümü IPv6'ya ek özellikler kazandırabilmektedir. IPv6 protokolünde paket başlığı sabit bir büyüklüğe sahiptir (40 oktet). Ek uzatma başlığı özelliği, protolokolü genişletilebilir yapmaktadır. Bu durum, gelecekte hizmet kalitesinin, güvenliğinin, taşınabilmesinin ve birçok özelliğin, temel protokol düzenlenmeden yapılmasına olanak sağlayacaktır.

**TCP (Transmission Control Protocol–İletim Kontrol Protokolü):** İletim kontrol protokolü, taşıma katmanında yer alır, bağlantılı (connection-oriented) ve güvenli bir iletişim sağlar. Bağlantı temelli iletişim, gönderici ve alıcı taraflar arasında oturum kurulduktan sonra veri iletişimini yapmasını ifade etmektedir. Bu protokolde verinin hedefe ulaşımı garanti edilir.

TCP paketleri (segment), iletilmek için IP paketlerine gömülür. IP, verinin gideceği hedef adresi belirlerken, TCP ise iletilecek verinin hangi servis ve uygulama tarafından kullanılacağını port numarası ile belirler [42].

TCP iletiminde, iki uç arasında üçlü el sıkışma (handshaking) olarak adlandırılan bağlantı kurulumu, *Flags* alanındaki *SYN* ve *ACK* bayraklarının ayarlanması ile yapılır. Öncelikle istemci sunucuya bağlantı kurma isteğini *SYN* işaretli paketi göndererek bildirir, buna karşılık bağlantıyı kabul eden sunucu *SYN* ve *ACK* işaretli bir paketle cevap verir, istemci bu paketi aldıktan sonra *ACK* işaretli bir paket gönderir. Bu iletişim sırasında karşılıklı anlaşma için paketlerin *Sequence Number* ve *ACK Number* alanları da ayarlanıp kullanılır.

TCP bağlantısını sonlandırmak isteyen taraf *FIN* bayrağı ayarlanmış bir paketi karşı tarafa gönderir. Karşı taraf *ACK* işaretli bir paket ile cevap verip, peşinden *FIN* işaretli bir paket gönderir. Bağlantıyı sonlandırmak isteyen taraf son olarak *ACK* işaretli bir paket ile cevap gönderdiğinde bağlantı sonlandırılmış olur.

TCP bağlantı kurulumu sayesinde, TCP tabanlı servislerde ve protokollerde IP Spoofing işlemi yapılamamaktadır. TCP'deki bayraklar ağ güvenliği alanında çokça kullanılmaktadır. TCP tabanlı saldırılardan en çok bilineni, TCP'deki el sıkışma işlemini istismar eden SYN Flood saldırısıdır. Ağ keşfinde, ICMP ping paketlerinin engellendiği ortamlarda TCP'nin SYN, ACK, FIN bayraklarından yararlanılmaktadır [37, 42].

**UDP (User Datagram Protocol–Kullanıcı Veri Bloğu Protokolü):** UDP, TCP'nin de bulunduğu taşıma katmanında yer alır. UDP güvenilir bir veri aktarımı sağlamaz. Yani, TCP'deki gibi verinin iletimini garanti edecek mekanizmalara sahip değildir. Bağlantı kurma, akış denetimi, tekrar iletim gibi özellikler olmadığından ve veri paketi (datagram) başlık boyutu küçük olduğundan veri iletimi daha hızlıdır. UDP, genellikle gerçek zamanlı gönderilecek ses ve video verilerinin iletimi için kullanılır. Çünkü bu gibi verilerin iletiminde hız faktörü daha önemlidir. Ayrıca önemli olmayan, bilgi amaçlı verilerin iletiminde de UDP tercih edilebilmektedir. DNS ve SNMP gibi protokollerin verilerinin iletiminde de kullanılır.

UDP paketleri, alıcıya iletilmek için TCP'de olduğu gibi IP paketlerine eklenip gönderilir. UDP, bağlantı temelli olmadığından, istenildiği kadar UDP verisi bir sunucuya veya bilgisayara gönderilebilir. Böylece, başta UDP Flood saldırısı olmak üzere, UDP tabanlı

saldırılar yapılabilmektedir. Ayrıca bir sunucuda UDP tabanlı (DNS, SNMP gibi) protokoller veya gerçek zamanlı video ve ses akışı kullanılıyorsa, sunucu üzerinde farklı güvenlik önlemleri alınmalıdır.

**ARP (Address Resolution Protocol–Adres Çözümleme Protokolü):** Ağ üzerindeki veri iletiminde hem göndericinin hem alıcının IP adresleri bilinir. Ancak yerel ağda veri iletilirken gideceği cihazın ağ bağdaştırıcısının fiziksel adresinin (MAC adresi) bilinmesi gerekir. ARP protokolü, IP adresini fiziksel adrese dönüştürmek için kullanılır.

Veri gönderecek olan bilgisayar, hedefteki bilgisayarın IP adresini bilir. Ağdaki cihazlara ve bilgisayarlara bu IP adresinin sahip olduğu fiziksel adresi (MAC adresini) soran istek paketleri gönderir. ARP istek mesajlarını alan bilgisayar veya ağ cihazları kendilerini ilgilendiriyorsa bu isteğe cevap verip kendi fiziksel adreslerini bildirirler. ARP isteğinde bulunan makine, verileri gelen bilgideki adrese gönderir [42].

Bilgisayarlar ARP istek mesajlarını gönderip fiziksel adresleri öğrendikçe IP adresi - MAC adresi bilgilerini ARP tablosu adı verilen bir tabloda tutup, daha sonra verilerin gideceği adresi bu tabloya göre otomatik seçerler. ARP tablosu üzerinde yapılabilecek bir değişiklikle yerel ağda *ARP Spoofing* veya *ARP Cache Poisoning* olarak adlandırılan bir saldırı türü mevcuttur.

**ICMP (Internet Control Message Protocol–İnternet Kontrol Mesaj Protokolü):** ICMP, ağın işleyişi ve performansı hakkında yararlı bilgileri toplama ve hata raporlama için kullanılan, TCP/IP'nin işleyişine yardımcı olan kontrol amaçlı bir protokoldür. ICMP veri paketinin başlığında *Type* ve *Code* gibi alanlar vardır. Bu bilgilerle paketin hangi durum bilgisini verdiği anlaşılır.

ICMP'nin sağladığı mesaj hizmetlerinden en önemli üçü şunlardır [43]:

- *Ping:* Başka bir IP adresinin erişilebilirliğini belirlemek için kullanılır.
- *Traceroute:* Bir IP adresine erişirken takip edilen yolu keşfetmek için kullanılır.
- *Destination Unreachable:* Bir yönlendirici, kendisinden sonra herhangi bir yönlendiriciye gitmeyecek olan bir datagram (veri paketi) algıladığında "destination unreachable" (hedef erişilemez) mesajı üretir ve onu datagram kaynağına iletir.

ICMP, bilgi toplama amaçlı kullanıldığından, bazı ağ ortamlarında genellikle kötü amaçlı kullanımlara karşı ICMP paketlerinin iletimine (bazı tipleri hariç) izin verilmemektedir.

**DNS (Domain Name System–Alan Adı Sistemi):** DNS, İnternet uzayını bölümleme, adlandırma ve bölümler arası işleri organize etmek için kullanılan bir sistemdir.

İnternet ağında her bilgisayar benzersiz bir IP adresine sahiptir. Ağ içerisinde iletişim kolaylığı için IP adreslerine karşılık isimler verilir. DNS, 256 karaktere kadar büyüyebilen host isimlerini IP adreslerine çevirmek için kullanılan bir sistemdir [42]. Host isimleri hem bilgisayarın adını verir hem de ait olduğu domaini belirlememize yardımcı olur.

DNS Protokolü, DNS'in çalışması için kullanılan sorguları, bu sorguların işlenmesi için gerekli paket yapılarını tanımlar. DNS protokolü, bir uygulama katmanı protokolüdür. DNS'in, ağ üzerinden veri iletiminde kullandığı taşıma katmanı protokolü UDP'dir. Ancak UDP'nin paket boyutunun yetmediği durumlarda TCP'yi kullanır. DNS sunuculara *DNS Flood* ve *DNS Poisoning* gibi saldırılar yapılabilmektedir.

**SNMP (Simple Network Management Protocol–Basit Ağ Yönetim Protokolü):** SNMP, basit ağ yönetim protokolü anlamına gelen bir uygulama katmanı protokolüdür. TCP/IP protokol kümesi içinde ağ yönetimi için tanımlanmış protokollerden en yaygın olanıdır. SNMP, ağ cihazları arasında yönetimsel bilgi transferi, ağ yöneticilerinin ağ başarımını yönetebilmesi, ağ üzerinde problemlerin bulunup çözülmesi ve ağ genişletme vb. durumların planlanmasını sağlayan bir protokoldür [42].

**FTP (File Transfer Protocol–Dosya Aktarım Protokolü):** FTP, kullanıcıların iki host (bilgisayar veya sunucu-istemci) arasında dosya kopyalamasını sağlayan protokoldür. FTP, ilk geliştirilen İnternet protokollerinden biridir. Bu protokol ile bir bilgisayardan başka bir bilgisayara dosya aktarımı yapılırken o bilgisayar ile eş zamanlı bağlantı kurulur ve protokol ile sağlanan bir dizi komutlar yardımıyla iki bilgisayar arasında dosya alma/gönderme işlemleri yapılır [42]. Kullanıcılar, FTP sunucusuna bağlanırken kimlik doğrulamasından geçerler. FTP ile host üzerindeki donanımdan bağımsız olarak dosya ve klasör listeleme, silme, ekleme, değişik komutları işleme yapılabilmektedir. FTP, dosya transferinin garanti altına alınması için taşıma katmanında TCP'yi kullanır.

**HTTP (Hyper Text Transfer Protocol–Hiper Metin Aktarım Protokolü):** HTTP, HTML (HyperText Markup Language) belgelerinin iletimini sağlamak için tasarlanmış bir protokoldür.

HTTP, istek (request) – cevap (response) faaliyeti tabanlıdır. HTTP bağlantısında, istemci taraf, bir web tarayıcı program kullanarak sunucuyla bağlantı kurup, sunucudan bir talep metodu formunda istekte bulunur. Sunucu, durum bilgisiyle birlikte cevap verir. Sunucunun cevabında hata veya başarı mesaj kodları, sunucu bilgilerini içeren mesajlar ve istenen sayfanın mümkün olan içeriği bulunur.

**SMTP (Simple Mail Transfer Protocol–Basit Posta Aktarım Protokolü):** Basit posta aktarım protokolü, varsayılan olarak TCP'nin 25 numaralı portunu kullanan İnternet üzerinden elektronik posta göndermek için kullanılan bir protokoldür. SMTP, POP3 protokolü ile birlikte e-posta hizmetlerinin bir parçası olarak kullanılmaktadır. SMTP, e-postanın hedef sunucuya ulaşıncaya dek İnternet boyunca aktarılma ve teslim edilme yöntemini denetler.

**DHCP (Dynamic Host Configuration Protocol–Dinamik Bilgisayar Konfigürasyon Protokolü):** DHCP, bir ağdaki makinelere IP adresi, alt ağ maskesi, ağ geçidi bilgilerini ve DNS sunucu bilgilerini otomatik olarak vermek için kullanılan bir protokoldür. Uygulama katmanı protokollerinden olan DHCP, taşıma katmanında UDP'yi kullanır.

Bir ağa bağlı her bilgisayara benzersiz bir IP adresi verilmesi çok zor ve zahmetli olduğundan DHCP, tüm ev ve işyeri ağlarında kullanılır ve sistem yöneticilerine büyük kolaylık sağlar. Temel olarak yerel ağda bir bilgisayar, DHCP sunucusu olarak ayarlanır ve bu sunucu ağdaki bütün bilgisayarlara otomatik olarak IP adresi atar.

DHCP, istek-cevap tabanlı bir protokoldür. Ağa bağlanan bir istemci ilk önce ortama bir keşif paketi gönderir, DHCP sunucusu keşif paketini aldığında istemciye verilebilecek IP numarasını bir paket ile ortama gönderir. Daha sonra istemci istek paketi gönderir, sunucu, istemcinin alabileceği IP numarasını cevap paketiyle döner. Belirlenen IP adresi bir süreliğine istemciye kiralanır, kiralama süresi bitince, istemci ağa bağlı ise adres tekrar tahsis edilir. DHCP sunucudan IP adresinin yanısıra, ağ geçidi ve alt ağ maskesi bilgileri de istemciye aktarılır.

**POP (Post Office Protocol–Posta İletim Protokolü):** POP, protokolünün güncel versiyonu POP3'tür. POP3, iyi bilinen 110 numaralı TCP portunu kullanan ve bir istemcinin yerel makinede, sunucunun da uzak makinede çalışmasını sağlar. SMTP protokolünün karşıtı olan POP3, uzak POP3 sunucusundan e-mail alıp, yerel POP3 istemcisine getirmek ve istemcinin sunucu üzerinde belli komutlar çalıştırarak e-maillerini yönetmesini sağlamakla görevlidir.

## 2.2 Ağ Güvenliği Tehditleri

Yeni teknoloji ve cihazların gelişmesiyle, bilgisayar ağ sistemlerinde alınan güvenlik önlemleri yetersiz kalabilmektedir. Bilgisayar ağları üzerinde birçok farklı saldırı türü bulunmaktadır. Bu bilgisayar ağ saldırı türlerinden literatürde en çok bilinenler incelenmiştir.

### 2.2.1 Paket Koklama

Sniffing (koklama-izleme), ağ paketlerinin takip edilip yakalanması ve analiz edilmesi işlemidir. Koklama işlemi için bilgisayarlar ve ağ cihazları aracılığıyla ağ ortamındaki veri trafiğini görebilecek şekilde fiziksel veya kablosuz bir bağlantı sağlanarak, paket izleyici programlar kullanılır.

Koklama işlemi, kullanım amacına göre saldırı olarak değerlendirilebileceği gibi ağdaki veri trafiğinin takibiyle saldırı tespiti veya verimli veri alışverişinin sağlanması gibi değişik amaçlar için olabilir. Kötü niyetli bir saldırgan, ağ paketlerini yakalayıp analiz ederek şifrelenmeden iletilen gizli, kişisel bilgilere erişebilir. Koklama işlemi ağ düzeyinde en basit saldırılardan birisidir.

Kötü niyetli kişiler, koklayıcı programları şu amaçlarla kullanırlar [44]:

- Kullanıcı adı ve şifre bilgilerini elde etmek
- Bir ağdaki kullanıcıların ağ kullanım örüntülerini keşfetmek
- Kişisel gizlilik derecesi olan bilgileri elde etmek
- VoIP telefon görüşmelerini yakalayıp tekrar oynatmak
- Bir ağın düzenini haritalamak
- Pasif işletim sistemi bilgilerini elde etmek

Koklayıcı programların kullanıldığı makinelerde, ağdan gelen her verinin kabul edilmesi isteniyorsa makinenin ağ arayüzünün promiscuous (her gelen veriyi kabul eden) modu desteklemesi gerekir.

### 2.2.2 Aldatma

Aldatma (spoofing), bir ağ ortamında bilgi alışverişinde bulunurken iletişim bilgilerinin farklı gösterilmesi ile karşı tarafın aldatılması işlemidir. Aldatma işleminde, en basit haliyle kaynak IP adresi, kaynak MAC adresi veya kaynak port numarası farklı gösterilerek karşı taraf aldatılmaktadır.

#### 2.2.2.1 IP Aldatmacası

İletişim halindeki sistemler arası veri transferi yapılırken kaynak IP adresinin farklı gösterilmesi ile hedef cihazın aldatılması şeklinde bir saldırıdır. Genellikle hizmet engelleme (DoS) saldırılarında saldırgan gerçek IP adresini gizleyerek kendisinin tespit edilmesini zorlaştırıp, hatta imkânsız hale getirmek için IP aldatmacası yöntemine başvurur.

TCP/IP protokol kümesindeki protokollerin bazı güvenlik sorunları üst katmanlardaki protokoller ile azaltılabilir veya yok edilebilir. Örneğin, TCP'nin el sıkışma ile bağlantı kurulumu sayesinde, kurulmuş bir bağlantı varsa, IP aldatmaca işlemi her zaman her ağ ortamında rahatça uygulanamaz. Ancak UDP ve DNS ile IP aldatmacası daha kolay uygulanabilir [38].

#### 2.2.2.2 MAC Aldatmacası

MAC, dünya üzerindeki her ağ bağdaştırıcısı cihazda benzersiz olarak bulunan 48 bitlik bir fiziksel adres numarasıdır. MAC aldatmacası (MAC Spoofing) ise, ağ arayüz kartlarının fabrika çıkışlı MAC adreslerini farklı gösterme yöntemidir. Çok farklı sebeplerle MAC aldatmacası yöntemine başvurulabilmektedir. Örneğin, ağda erişim denetiminden geçme, bir bilgisayarın ağdaki varlığını gizleme ya da bir bilgisayarın başka bir bilgisayar gibi davranması, yani ilgili bilgisayarı taklit etmesi gibi işlemler MAC aldatmacası ile mümkündür.

#### 2.2.2.3 ARP Önbellek Zehirlenmesi

ARP önbellek zehirlenmesi (ARP Cache Poisoning), *ARP Poisoning* veya *ARP Spoofing* olarak da anlandırılabilmektedir. Özellikle switch cihazı kullanılan yerel ağlarda, Man In The Middle (MITM, Ortadaki Adam) saldırılarında veya ağ izlemede kullanılır.

Bir yerel alan ağında hub cihazı, bir portuna gelen veriyi diğer tüm portlarına gönderir. Hub'a bağlı olan tüm bilgisayarlar ve cihazlar veriyi görür, ancak sadece hedef bilgisayar onu alıp işleme koyar. Ancak switch kullanılan ağlarda, switch, IP adresi ve MAC adresi

eşleştirmelerinin tutulduğu ARP tablosunu kullanarak kendisine gelen veriyi sadece hedef bilgisayara gönderir.

Adres Çözümleme Protokolü (ARP) ile IP-MAC dönüşümü yapılır. ARP ön bellek zehirlenmesinde, switchlerdeki ARP tablolarındaki IP-MAC eşleştirmeleri üzerinde değişiklikler yapılır. Saldırgan, kurban olarak belirlediği bilgisayara ait IP adresi ile kendi MAC adresini eşleştirir. Bu durumda kurban bilgisayara gidecek olan veriler, Switch tarafından saldırgan bilgisayara gönderilecektir.

#### 2.2.2.4 DNS Zehirlenmesi

DNS zehirlenmesi (DNS Poisoning) işlemi, ARP Poisoning ile benzer yapıdaki bir saldırı türüdür. DNS sunucuları, İnternet ortamındaki alan adlarının IP numaraları ile eşleştirildiği tabloları tutarlar. Saldırganlar, DNS sunucularındaki açıklardan faydalanarak, DNS tablolarında değişiklik yapabilir. Bu saldırı, belirli bir alan ismine tahsis edilmiş IP adreslerinin değiştirilerek, istenen IP adreslerinin DNS tablolarına girilmesi şeklindedir. Böylelikle bir kullanıcı bu alan ismi ile web sunucuya erişmek istediği zaman, saldırganın belirlediği IP adresine sahip sunucuya yönlendirilecektir.

### 2.3 Hizmet Engelleme Saldırıları

Hizmet engelleme (DoS - Denial of Service), İnternet dünyasında en çok karşılaşılan saldırı türüdür. Bu saldırıda, saldırgan hedefindeki sistemi protokol açıklarından yararlanıp, ya sürekli meşgul edip kaynak yetersizliğinden ya da sunucunun ağ bağlantısındaki bant genişliğini doldurup iletişimi tıkamasından dolayı hizmet veremeyecek duruma düşürür.

Dağıtık hizmet engelleme ise (Distributed DoS), hizmet engelleme işleminin çok sayıda zombi bilgisayar kullanarak, organize bir şekilde yapılması işlemidir. Yüksek bant genişliğine ve hizmet kapasitesine sahip sunucular ancak DDoS ile hizmet veremez duruma getirilebilir.

Saldırgan, DDoS saldırılarında IP Spoofing ile kendi IP adresini gizleme ihtiyacı hissedecektir, ancak bu durum üst katmanlardaki bazı protokollerin kullanımında mümkün değildir. Örneğin, SYN Flood saldırısında IP Spoofing mümkün iken, HTTP GET Flood saldırısında kurulmuş bir bağlantının gerekliliğinden dolayı bu mümkün değildir. Bilinen hizmet engelleme saldırıları aşağıda açıklanmıştır.

### 2.3.1 SYN Taşması

SYN taşması (SYN Flood), en kolay gerçekleştirilen hizmet engelleme (DoS) saldırı çeşitlerinden biridir. TCP ile haberleşen iki cihaz arasında bağlantı kurulurken, karşılıklı *SYN* ve *ACK* bayraklarına sahip TCP paketleri, hedef ve kaynaklar tarafından birbirlerine gönderilir. TCP'de el sıkışma olarak adlandırılan bu aşamadaki paket gönderimleri istismar edilerek SYN Flood gerçekleştirilebilir.

SYN taşması saldırısında, saldırgan hedef sunucuya sürekli olarak *SYN* bayraklı TCP paketleri gönderir. Hedef sunucu, kendisine gelen her SYN paketine bağlantı kurulumunu gerçekleştirmek amacıyla *ACK* paketleri ile cevap vermeye çalışır. Bu kadar çok sayıda *SYN* paketi geldiğinde belli bir kapasiteden sonra sunucu artık cevap veremez duruma gelir.

### 2.3.2 ACK/FIN/PUSH Taşması

Bu taşma saldırıları, veri gönderirken TCP başlığındaki *ACK*, *FIN* veya *PUSH* bayraklarının ayarlanıp sunucuya sürekli gönderilmesi ve böylece sunucu kapasitesinin zorlanması şeklindeki saldırılardır. Hedef sunucuda, *SYN* taşması saldırısına karşı tedbir alınmış ise saldırganlar tarafından denenen saldırı türüdür. Ancak engellenmesi kolay ve *SYN* Flood kadar etkili bir saldırı türü değildir.

### 2.3.3 UDP Taşması

UDP taşması, saldırganın hedef sunucunun farklı port numaralarına sürekli olarak UDP paketleri göndermesi şeklinde gerçekleştirilen bir saldırı türüdür.

UDP protokolünde oturum kurulumu olmadığı için açık portlara istenildiği kadar paket gönderilebilir. Bu durumda IP Spoofing yapılması için de bir engel yoktur. Sunucuya UDP paketi geldiğinde, sunucu sonra gelecek veriler için bağlantıyı bir müddet açık tutar. Böylece sunucuda, sürekli bir kaynak tüketimi meydana gelir ve belli bir kapasiteden sonra artık gelen bağlantılara cevap veremez. UDP taşması işleminde, UDP paket boyutunun küçüklüğü, belli bir zamanda gönderilebilecek paket sayısının daha fazla olmasını sağladığından dolayı avantajlıdır.

### 2.3.4 DNS Taşması

DNS, TCP ve UDP protokolleri üzerinde çalışabilen bir sistemdir. DNS protokolü, UDP protokolü üzerinde çalıştığı zaman saldırılara açık olmaktadır. DNS taşması işleminde, UDP üzerinden IP aldatmacası kullanımıyla birlikte DNS istek paketlerinin gönderiminde saldırgan açısından bir zorluk bulunmamaktadır. DNS taşması, saldırgan tarafından DNS

sunucuya sürekli olarak rastgele istek paketleri gönderilip DNS sunucu meşgul edilerek yapılmaktadır.

### 2.3.5 MAC Taşması

Yerel alan ağlarında, bilgisayarlar arasındaki bağlantıyı sağlayan switch, kendisine gelen veriyi sadece ilgili makineye gönderdiğinden, bu tip yapılarda sniffing (dinleme) işlemi yapılamamaktadır. Switch kullanılan ağdaki trafiği dinlemek ARP ön bellek zehirlenmesi ve MAC taşması ile mümkün olmaktadır.

Switch, kendisine bağlı makinelerin MAC adresleri ile IP adreslerini eşleştirip bir tabloda tutar. MAC-IP eşleştirme tablosu, yerel ağdaki makine sayısı arttıkça zamanla dolabilir. Tablo dolduğu zaman switch, bir hub cihazı gibi davranmaya başlar. MAC taşması işlemi, bu durumu istismar eden bir saldırı tipidir. Saldırgan, switch cihazına rastgele sahte MAC adres bilgileri göndererek tablonun dolmasını sağlar ve amacına ulaşmış olur [44].

### 2.3.6 HTTP GET/POST Taşması

HTTP, web sayfalarının sunulması için kullanılan bir uygulama katmanı protokolüdür. HTTP GET/POST taşması saldırısında, web sunucuya sürekli olarak web sayfası istek mesajları gönderilir ve sunucu meşgul edilir.

HTTP'de sayfa isteklerinin gerçekleşmesi için gelen-giden veri miktarı sadece TCP veya UDP kullanılarak iletilen veriden çok daha fazladır. Bundan dolayı HTTP GET/POST taşması saldırısı sunucunun kaynaklarını daha fazla tüketir. Bu saldırı SYN taşması saldırısına kıyasla daha kolay önlenebilmektedir. HTTP, TCP'yi kullandığından, TCP bağlantı kurulumunun gerçekleştirilmesi gerektiğinden, HTTP taşması işlemiyle beraber IP aldatmacası yapılamaz.

## 2.4 Ağ Keşfi

Ağ keşfi, siber bir saldırının gerçekleştirilebilmesi için hedef sistemin mevcut durum analizinin yapılması ve zafiyet arama olarak özetlenebilir. Bu arama ile ağda bulunan hostlar, cihazlar, kullanılan servisler, IP numaraları ve sunuculardaki açık portları tespit etmek mümkündür. Ağ keşfi yöntemleri aşağıda özetlenmiştir.

### 2.4.1 IP Tarama

IP tarama, bir ağdaki açık makineleri tespit etme işlemidir. En basit IP tarama yöntemi Ping Scan'dır. Bu yöntemde, belirlenen IP adresine ICMP echo request paketleri gönderilir.

Eğer ilgili IP adresi kullanımda ise ICMP echo reply paket tipi dönecektir. Bazı sunuculardaki güvenlik duvarları Ping paketlerinin geçişine izin vermemektedir. Bu durumda port tarama işlemlerinde kullanılan teknikler yardımcı olacaktır.

### 2.4.2 Port Tarama

Port tarama, değişik tekniklerle bir sunucunun hangi port numaralarından veri alışverişi yaptığının tespit edilmesi işlemidir. TCP, IP, ICMP, UDP gibi protokollerin bazı özelliklerinden hareketle onlarca farklı teknik ile açık port tespiti mümkündür. Port tarama tekniklerinden bir kısmı aşağıda açıklanmıştır [44, 45].

TCP Connect Scan işleminde, saldırgan, hedef sunucuya *SYN* paketi gönderir, sunucudaki port açıksa *SYN+ACK* sinyali döner, saldırgan *ACK* göndererek bağlantıyı kurmuş olur ve peşinden *RST* göndererek sonlandırır. Bu tür bir tarama kolaylıkla teşhis edilebilir.

TCP SYN Scan işleminde, hedef sunucuya *SYN* bayraklı TCP paketi gönderilir. Paket gönderilen port açıksa, *SYN+ACK* paketi cevap olarak döner ve saldırgan bir *RST* (reset) paketi göndererek bağlantıyı kurmadan iptal eder. Port kapalı ise sunucudan *RST* cevabı gelecektir.

TCP FIN Scan işleminde, saldırgan, hedef sunucuya *FIN* paketi gönderir, taranan port açıksa herhangi bir cevap gelmeyecek, port kapalı ise *RST+ACK* paketi dönecektir.

TCP Xmas Scan işleminde, hedef sunucuya geçersiz bayrak ayarlarıyla *FIN+PSH+URG* paketi gönderilir. Dönecek cevap TCP FIN Scan ile aynıdır. Ancak Microsoft Windows, Cisco IOS, BSD, HP/UX ve IRIX işletim sistemlerinde çalışmaz, çünkü port açık veya kapalı olsun dönen cevap daima *RST* olacaktır.

Null Scan işleminde, hiçbir bayrağı set edilmemiş bir TCP paketi gönderilir, dönecek olan cevap FIN Scan ile aynıdır.

UDP Scan işleminde, hedef makineye UDP paketi gönderilir, port kapalı ise dönen cevap ICMP Post Unreachable olacaktır.

### 2.4.3 İşletim Sistemi Tarama

İşletim sistemi tarama (OS Fingerprinting), hedef sistem üzerinde çalışan işletim sisteminin tespit edilmesi işlemidir. Bu tespit işleminde, ilgili bilgisayarın ağ ortamına

gönderdiği verilerin analiz edilmesi ile işlem yapılır. Bu işlem aktif ve pasif olarak yapılabilmektedir. Aktif modda, saldırgan tarafından hedefteki makineye veri paketleri yollanıp geri dönen cevapların analizi ile tespit işlemi yapılır. Pasif modda ise, hedefteki makine ile iletişim kurulmadan, sadece ağ ortamındaki verilerin kullanılması ile tespit işlemi yapılır. Pasif modda işletim sistemi taraması için kullanılan araç bir sniffer gibi çalışır.

İşletim sistemi taraması için genellikle IP paketindeki *TTL* ve *ID* değerleri, TCP başlığındaki *SYN*, *SYN+ACK* değerleri ve TCP pencere boyutu, DHCP istekleri, ICMP istekleri, HTTP paketleri (genellikle User-Agent alanı) gibi veriler analiz edilir [46].

# 3. BİLGİ GÜVENLİĞİ SİSTEMLERİNİN İNCELENMESİ

## 3.1 Giriş

İnternet'in yaygın ve yoğun kullanımıyla modern dünyada yaşam tarzları ve çalışma şekilleri değişime uğramıştır. Günümüzde, gelişen teknoloji ve yazılımlar yardımı ile artık insanlar evlerinden ayrılmadan İnternet'e bağlı bir bilgisayar ile uluslararası ticaret yapabilmekte, işyerlerini uzaktan denetleyebilmekte, telefon görüşmelerini yapabilmekte, elektronik seçimlere katılabilmektedir. Banka şubesine gidilmeden neredeyse tüm bankacılık işlemleri gerçekleştirilebilmektedir. Teknolojinin bu müthiş gelişiminin yanı sıra, kötü niyetli kişiler de bu sistemleri farklı amaçlarla istismar edebilmektedir. Bu kişiler sistemlere saldırarak, bir bankacılık sistemine zarar vermek, kişisel veya şirket kullanıcı bilgilerini ele geçirmek, kendi namına çıkar veya güç sağlamak, ağ sistemlerini fiziksel zarara uğratmak gibi işlemleri yapabilmektedir. Kısacası ağ teknolojileri ve bilişim sistemlerine olan bireysel ve toplumsal bağımlılık veya bu sistemlerin kullanım oranı arttıkça bu sistemlerde meydana gelebilecek sızmalara/saldırılara karşı duyarlılık da artmaktadır. Bu saldırılar ile sistemler, kişiler veya tüzel kişilikler maddi-manevi zarar görebilmekte, itibar kaybı yaşayabilmektedir.

Bilişim sistemleri güvenliğini tehdit eden saldırılar ve bu saldırıların türleri tarihsel süreç içerisinde yeni teknolojilere paralel olarak artmıştır. Bunun temel sebebi, sistemlerin inşa edilmeleri sırasında bir takım güvenlik açıklarına sahip olmaları bir başka ifadeyle sistemler geliştirilirken güvenlik unsurlarının göz ardı edilmesidir.

Bilişim sistem güvenliğinin sağlanması için çeşitli savunma mekanizmaları da geliştirilmekte ve gerek yazılımsal gerekse donanımsal olarak birçok korunma mekanizması kullanılmaktadır. Burada temel amaç, olası bir saldırı durumunda sistem güvenliğini sağlamak, veri kayıplarını önlemek, sistem erişilebilirliğini en üst düzeyde tutabilmektir. Bilişim sistemlerinde kullanılan savunma mekanizmalarından birisi de 'Saldırı Tespit Sistemleri'dir. STS'ler, bilişim sistemlerinde olayları analiz ederek zararlı davranışları tespit etmeye yönelik sistemlerdir. STS'lerde temel olarak kötüye kullanım ve anormallik tespiti şeklinde iki yaklaşım bulunur. Kötüye kullanım yaklaşımında, iyi bilinen saldırı örüntüleri kullanılarak saldırı imzaları ile eşleşen olgular saldırı olarak belirlenir. Anormallik tespiti yaklaşımında ise beklenen normal kullanım profillerinden sapma gösteren etkinlikler anormal durum olarak değerlendirilir. Anormallik tespiti yaklaşımında, kötüye kullanım yaklaşımından avantajlı bir yapı vardır. Çünkü bu yaklaşımda bilinmeyen saldırıların tespiti

de mümkündür. Bu bölümde bilgi güvenliği ve saldırı tespit sistemleri ile ilgili temel kavramlar, konu ile ilgili literatür bilgilerinden yararlanılarak verilmektedir. Böylece, saldırı, saldırgan, saldırı tespit sistemleri, saldırı tespit sistemlerinin türleri ve saldırı tespit sistemlerinde kullanılan yöntemler detaylı bir biçimde incelenmiştir.

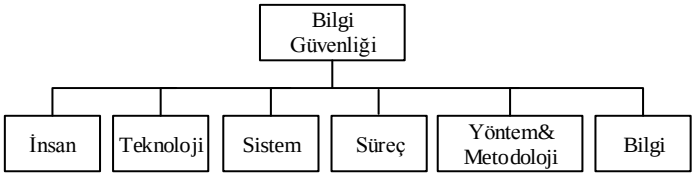
## 3.2 Bilgi Güvenliği

Bu kısımda, bilgi güvenliği temel ilkeleri, bilgi iletimindeki unsurlar ve tehditler, bilgi güvenliği yazılımları, saldırı tespit ve engelleme sistemleri, mevcut saldırı tespit araçları ve saldırı tespit sistemlerinde kullanılan veri kümeleri konularında detaylı bilgiler verilmektedir.

### 3.2.1 Bilgi Güvenliği Temel İlkeleri

Bilgi güvenliği, kurumsal ve kişisel anlamda önem ifade eden bilgilerin yetkisiz erişimlere karşı korunmasıdır. Bilgilerin dijital arşivlerde güvenli bir biçimde saklanması, ihtiyaç duyulduğunda sadece yetkili kişiler tarafından erişilebilir ve değiştirilebilir olması, bilgi güvenliğinin temel unsurlarını ifade eder. Bilgi güvenliği, güvenlik gereksiniminin oldukça yükseldiği günümüz koşullarında bilişim sistemleri açısından önemi artan ve içerisinde birçok olguyu barındıran bir süreçtir.

Bilgi güvenliği, Şekil 3.1'de gösterildiği gibi içerisinde insan, teknoloji, yöntem, metodoloji ve bilgi kavramlarını barındıran kapsamlı bir süreçtir. Bilişim sistemleri ve özellikle İnternet teknolojilerinin yaygınlaşmasıyla birlikte bu sistemlerdeki açıklıklardan kaynaklanan güvenlik olayları da artmaya başlamıştır. Ortaya çıkabilecek olası yeni açıklıkların fark edilmesi ve olası zararın önlenebilmesi, maksimum düzeyde güvenliğin sağlanabilmesi amacıyla birçok araç geliştirilmiştir [6].

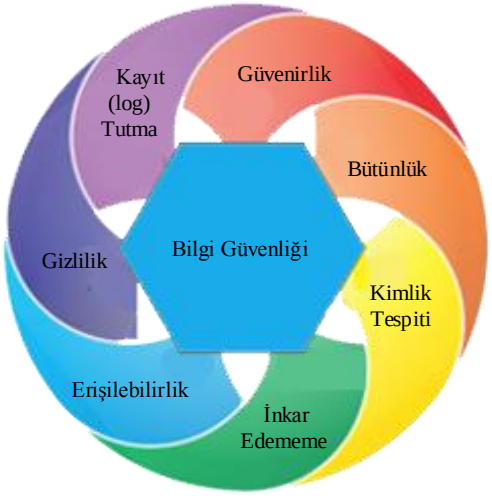



**Şekil 3.1** Bilgi güvenliği süreci

Bilgisayar ağları kullanılarak gerçekleştirilebilecek kötücül amaçlı aktiviteler, bilinçsiz kullanımlar sonucu oluşabilmektedir. Ancak sistemin açıklıklarından faydalanarak sisteme bilerek zarar vermek isteyen saldırı faaliyetlerinin oranı daha fazladır. Özellikle web teknolojilerinin gelişip yaygınlaşması sonucu saldırıların sayıları ve türleri de artmıştır [6]. Birçok farklı türü olan saldırılara karşı, kimlik doğrulama, yetkilendirme, antivirüs

programları gibi güvenlik çözümleri geliştirilmeye çalışılmıştır. Bilinen ilk saldırılar, parola tahmini, basit kod yürütme gibi etki ve olasılığı düşük saldırılar olmasına rağmen gelişen teknoloji ile beraber saldırı karmaşıklığı ve etkisi artmıştır. Buna rağmen saldırıların gerçekleştirilebilmesi için gereken bilgi düzeyi düşmüştür. İnternet üzerinden kontrolsüz bir şekilde yayılan bilgiler ve otonom araçlar bunun en temel sebebidir.

Statik veya dinamik ortamlarda saklanan ve ihtiyaç duyulması halinde iletilen bilginin güvenliğinden söz edebilmek için, bilgi güvenliği temel ilkelerinin sağlanması gerekmektedir. Bilgi güvenliğinin temel ilkeleri gizlilik, bütünlük ve erişilebilirlik olmak üzere üç tanedir. Bunların yanı sıra bilgi güvenliğinin kayıt (log) tutma, kimlik tespiti, güvenirlik ve inkâr edememe ilkeleri de vardır. Bilgi güvenliği temel ilkeleri Şekil 3.2'de gösterilmektedir.




**Şekil 3.2** Bilgi güvenliği temel ilkeleri [47-49]

#### 3.2.1.1 Gizlilik

Bilginin sadece yetkisi ve erişim izni olan kişiler tarafından erişilebilir olması, izni olmayan kişilerin erişimlerinin engellenmesidir [47]. Gizlilik, statik ortamlarda (disk, teyp, cd, dvd vb.) durağan veya ağ üzerinde bir göndericiden bir alıcıya gönderilen dinamik ortamdaki veriler için sağlanmak zorundadır [48]. Bilgi gizliliğinin sağlanması için kriptografi ve steganografi yöntemlerinden yararlanılmaktadır.

#### 3.2.1.2 Bütünlük

Bilginin kaynağından bir bütün olarak hedefine ulaştırılması ile bütünlük ilkesi sağlanır [47]. Bilgi, iletişim sürecinde değiştirilmemeli, üzerine yeni veriler eklenmemeli, bir kısmı ya da tamamı tekrar edilmemeli ve sırası değiştirilmemelidir. Bütünlük ilkesi ayrıca, bilginin

yalnızca yetkili kişiler tarafından değiştirilebilir olmasını garanti altına almaktadır. Bilginin bütünlüğünün sağlanması için *hash* fonksiyonları ile verinin özeti alınabilir.

#### 3.2.1.3 Erişilebilirlik

Sistem erişilebilirliği, ihtiyaç duyulması durumunda bilgiye erişilmesini garanti altına almaktadır. Bilgi güvenliğinin üç temel unsuru (gizlilik, bütünlük, erişilebilirlik) sağlanmadıkça bilgi güvenliğinden söz edilemez. Bilgi sürekli bir şekilde erişilebilir halde bulunmalıdır. Bilgisayar yazılımlarındaki hatalar, sistemin yanlış, bilinçsiz ve eğitimsiz personel tarafından kullanılması veya konfigüre edilmesi, doğal felaketler gibi faktörler de sistem sürekliliğini etkileyebilir [47]. Sisteme erişilebilirliğin sürekli sağlanması için fiziksel önlemler alınmalı, güvenlik duvarları, saldırı tespit sistemleri, antivirüs yazılımları kurulmalıdır.

#### 3.2.1.4 Kayıt Tutma

Log dosyaları, elektronik ortamda gerçekleşen olayların daha sonra analiz edilmek üzere kayıt altına alındığı yapılardır [49]. Sistem üzerinde gerçekleştirilmiş saldırılarda, saldırganın kimliğinin tespit edilmesinde, log dosyaları önemli rol oynar. Log analizleri sonucunda elde edilen bulguların inkâr edilemez olması için log kayıtlarının da bilgi sistemleri güvenliğinin ilkelerinden gizlilik, bütünlük ve erişilebilirlik niteliklerini kaybetmemesi önemlidir. Saldırganların kimliklerinin tespit edilmesinde, dış servis olarak hizmet veren kurumların, belirlenen yöntemlerle log kayıtlarını tutmaları gerekliliği yasal zorunluluk haline getirilmiştir.

#### 3.2.1.5 Kimlik Tespiti

Bilgi sistemlerinden hizmet alan alıcının, iddia ettiği kişi olduğundan emin olunmasıdır [49]. Kimlik tespitini sağlamak için bilgi sistemlerinde parola erişimli oturumlar kullanılabilir. Parolanın yanı sıra daha spesifik durumlarda akıllı kartlar, bir seferlik parola, token veya biyometrik teknolojiler de kullanılabilmektedir.

#### 3.2.1.6 Güvenirlik

Bilgisayar sistemlerinden beklenen davranış ile elde edilen sonuçlar arasındaki tutarlılık durumudur. Başka bir deyiş ile güvenirlik, sistemden ne yapmasını bekliyorsak, sistemin kendisinden beklenileni yapması ve her çalıştırıldığında da aynı şekilde davranması olarak tanımlanabilir.

#### 3.2.1.7 İnkâr Edememe

Gönderici ile alıcı arasında ortaya çıkabilecek anlaşmazlıkların en aza indirilmesini sağlamaya yardımcı olmaktadır. İnkâr edememe ilkesi, sayısal imza teknikleriyle sağlanır [50]. Bu ilkenin, özellikle gerçek zamanlı işlemler gerçekleştiren bankacılık ve finans bilgi sistemlerinde önemi daha büyüktür.

### 3.2.2 Bilgi İletimindeki Unsurlar ve Tehditler

Tehdit, "bir sistemin veya kurumun zarar görmesine neden olan istenmeyen bir olayın arkasındaki gizli neden" olarak tanımlanabilir. Her tehditin bir kaynağı ve bu kaynağın yararlandığı sistemdeki bir "güvenlik boşluğu" vardır. "Sistemi neye karşı korumalıyım?" sorusuna verilecek cevap bir sisteme yönelik olan tehditlerin belirlenmesine yardımcı olmaktadır [48]. Tehditler, insan kaynaklı tehditler, fiziksel tehditler, yazılım kaynaklı tehditler, güvenlik boşlukları, eğitim ve bilinç eksikliğinden kaynaklanan tehditler olarak sınıflandırılabilir.

**İnsan Kaynaklı Tehditler:**

İnsan kaynaklı tehditleri kendi içinde iki alt gruba ayırabiliriz:

*Kötü niyet olmayan davranışlar sonucu oluşanlar:* Bir kullanıcının, bilinçsiz, bilgisiz ve yeterli eğitime sahip olmadan sistemi kullanması sonucu, sistemde ortaya çıkma olasılığı olan aksaklıklardır.

*Kötü niyetli davranışlar sonucu oluşanlar:* Sisteme zarar verme amacıyla, sisteme yönelik olarak saldırganlar tarafından yapılacak tüm kötü niyetli davranışlardan kaynaklanan tehditlerdir. Bu tür tehditlerde saldırganlar, tehdit kaynağı ve sistemde bulunan güvenlik boşluklarından yararlanırlar [51].

**Fiziksel Tehditler:**

Deprem, yangın, su baskını, sel, ani sıcaklık değişimleri, toprak kayması, çığ düşmesi bu tür tehditlere örnek olarak verilebilir. Bu tür tehditler genellikle önceden tespit edilemezler, doğal olaylar olmaları sebebiyle genellikle engellenemezler.

Tehditin geliş yönüne göre de sınıflandırma yapılabilir. Buna göre iç tehditler, kurum içinden kuruma yönelik yapılabilecek saldırılar, dış tehditler ise kurum dışından kuruma yönelik olarak yapılabilecek saldırılar olarak tanımlanır. Son yıllarda yapılan araştırmalara göre [48,51], güvenlik ihlallerinin % 67'si iç tehditler sonucunda ortaya çıkmaktadır.

**Yazılımsal Tehditler:**

Donanımlar, yazılımlar (işletim sistemi, programlar, uygulamalar) olmadan çalışamazlar. Yazılımlar, bilgisayarlarda çalışmasıyla birlikte çok çeşitli tehditlerle karşı karşıya kalabilmektedir. Yazılımlar, kötü amaçlı olarak tahrip edilebilir, değiştirilebilir, silinebilir veya kaza ile değişikliğe maruz kalabilir. Yazılımlardaki değişiklikler kolayca fark edilemezler. Örneğin, bir programı daha önceden yaptığı her şeyi yapacak ve bunun yanında saldırganın istediği ekstra işlemleri de yapacak şekilde değiştirmek mümkündür [51]. Bu şekilde fonksiyonel işlevlerinin yanı sıra casus işlevler de kazanan yazılımlar, bilgi güvenliği açısından tehlike oluşturmaktadır.

**Korunmasızlık:**

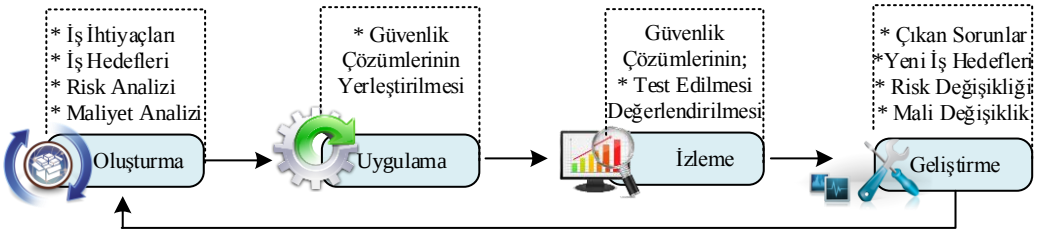
Yazılım veya donanımdan kaynaklanan zafiyetler, güvenlik boşluğu olarak tanımlanabilir. Bir güvenlik boşluğu sayesinde saldırgan, sistemdeki bilgisayarlara veya bilgisayar ağı üzerindeki kaynaklara yetkisiz olarak erişebilir [51]. Sistemde bulunan mevcut açıkları kapatmak için donanım ve yazılım güncelleştirmelerinin takip edilmesi önemlidir. Sistem üzerindeki açıklıkların ve olası yeni tür saldırıların tespit edilmesi, oluşabilecek güvenlik boşlukları ve korunmasızlık durumlarının giderilmesi bu tez çalışmasının amaçlarındandır.

**Eğitim ve Bilinç:**

Bilgi sistemleri güvenliğinde en zayıf halkalardan birisi insan faktörüdür. Bilgi sistemleri kullanıcılarının, bilgi güvenliği konusunda eğitimlerle bilinçlendirilmesi, onların bir güvenlik boşluğu oluşturmalarını ve kurum açısından risk oluşturacak bir etken olmaları olasılığını en aza indirecektir [52]. Bu sebeple kurum ve kuruluşlar, bilgi sistemlerinin birer unsuru olan çalışanlarının farkındalığını arttırmak için eğitimler düzenlemelidir. Ayrıca özellikle güncel hususlarda çalışanlar duyurular ile bilgilendirilmeli, böylece bilinç düzeyi arttırılmalıdır.

**Güvenlik Politikaları:**

Bilgi güvenliği politikası, kurum, kuruluş veya organizasyonlara bilgi güvenliğinin uygulanması için gerekli olan önlemlerin alınmasını sağlayan kurumsal etkinliklerdir [53]. Güvenlik politikaları, organizasyona özgü, değişebilir özel kanunlar olarak da düşünülebilir. Her organizasyon içerisinde genel olarak Şekil 3.3'te gösterilen süreç işlemektedir.




**Şekil 3.3** Güvenlik politikası yaşam döngüsü [48]

Temel olarak Şekil 3.3'te verilen aşamalardan oluşan güvenlik politikası yaşam döngüsü, bir yazılımın yaşam döngüsüne benzetilebilir. Buna göre güvenlik politikaları; oluşturulur, kurum içerisinde uygulanır, uygulanırken test edilip değerlendirilir ve bu testler sonucundaki raporlara göre yeniden geliştirilebilir. Böylelikle değişebilecek ihtiyaçlara veya yeni risklere göre sürekli olarak güncellenmelidir.

**Bilgi Güvenliğinin Sınıflandırılması:**

Bilgi güvenliğinin sınıflandırılmasında çeşitli yaklaşımlar bulunmaktadır. Bu yaklaşımlara göre bilgi güvenliği üç temel kategoride değerlendirilmektedir. Bunlar [55];

- Mantıksal güvenlik
- Çevre güvenliği
- Fiziksel güvenlik

*Mantıksal Güvenlik:* Bilgi sisteminin iletişim ağları vasıtası ile maruz kalabileceği tehditleri kapsamaktadır.

*Fiziksel Güvenlik:* Bilgi sistemlerini barındıran fiziksel altyapının güvenliğini tarif etmektedir. Fiziksel güvenliğin kapsamına sunucu ve istemci donanımları, sistem odası, sistem odasının bulunduğu bina, güç hatları gibi bileşenler girmektedir.

*Çevre Güvenliği:* Fiziksel güvenlikle birlikte düşünülebilir. Ayrıldığı nokta ise bilgi sistemini barındıran bina veya kampüs alanının sınırlarında alınacak güvenlik önlemleridir [56].

Bir bilgi sistemi, katmanlı olarak düşünülürse yapılan sınıflandırmaların hangi katmanlara denk geldiği Tablo 3.1'de görülebilir. Burada en alt düzeyden en üst düzeye kadar, kurumsal işleyiş açısından sahip olunan değerler, temel olarak üç düzeyde ele alınmıştır. Buna göre, çalışanlar, kurumun sahip olduğu tesisler ve fiziksel bütün donanımlar, fiziksel güvenlik katmanı olarak değerlendirilmektedir. Kurumsal altyapıyı oluşturan, iç ağ, İnternet, işletim sistemleri ise altyapı güvenliği katmanı olarak değerlendirilmektedir. Benzer şekilde kurum

tarafından kullanılan yazılım sistemleri, veri tabanı vb. diğer hizmetler ise uygulama güvenliği katmanı olarak değerlendirilmektedir.

**Tablo 3.1** Bilgi sistemi katmanları [57]

| | |
|---|---|
| Bilgi Sistemi Uygulamaları, Hizmetler | Uygulama Güvenliği |
| Veri tabanları | |
| Hazır Yazılımlar | |
| İşletim Sistemleri | Altyapı Güvenliği |
| İç Network | |
| İnternet | |
| Fiziksel Ekipman, Personel, Tesis | Fiziksel Güvenlik |

### 3.2.3 Bilgi Güvenliği Yazılımları

Bilgi güvenliğinin sağlanması için çok çeşitli donanım ve yazılımlar kullanılmaktadır. Kişisel ya da kurumsal ihtiyaca göre bilgi teknolojileri yöneticileri, bu araçları temin ederek sistem üzerinde sürekliliğini ve güncelliğini sağlamak durumundadır. Bilgi güvenliği yazılımları, kurumsal ve kişisel anlamda öneme sahip kritik bilgilerin korunması açısından elzem araçlardır. Bu araçların birbirinin yerine kullanılamayacakları ve birbirini tamamlayıcı yapılar olduğu unutulmamalıdır [56]. Güvenlik ihtiyaç ve gereksinimlerine göre bu araçlar hibrit bir biçimde kullanılırlar.

#### 3.2.3.1 Güvenlik Duvarları

Güvenlik duvarları (firewall), temel ağ güvenliği sistemlerinden biri olup statik izleme kabiliyeti sayesinde üzerinden geçen trafik için belirlenen kurallara göre erişim denetimi yapmak amacıyla kullanılırlar. Güvenlik duvarları, sahip oldukları kural tablosu yardımı ile istenmeyen yere doğru giden veri geçişini engelleyebilmektedirler. Ağ içerisinde ana segmentler arasında kullanılabilmeleri sayesinde, segmentler arasında erişim kuralları uygulanabilmektedir. Güvenlik duvarları, dış dünya ile olan bağlantıyı sağladığından ve kullanıcıların şifreli bir şekilde bağlantı kurmalarına olanak verdiğinden verinin gizliliğinin korunmasına yardımcı olmaktadır.

#### 3.2.3.2 Saldırı Tespit ve Engelleme Sistemleri

STS'ler, saldırıları trafik üzerinde önceden belirlenmiş saldırı imzalarına uyan trafiği raporlayarak, kritik durumlarda sistem yöneticisini uyarabilir. Saldırı engelleme sistemleri gibi ağ trafiğine müdahale etme özellikleri yoktur. Sonraki bölümde detayları ile birlikte ele alınacaktır.

Saldırı engelleme sistemleri, korunmak istenen ağ segmentlerinin bağlantıları üstüne konularak zararlı trafiğin kesilmesini sağlar. Saldırı engelleme sistemleri, trafik üzerinde önceden belirlenmiş saldırı imzalarına uyan trafiği ararlar ve bulduklarında, paket düşürme, TCP bağlantısını sonlandırma gibi eylemlerde bulunabilirler. Bu özelliklere ek olarak servis dışı bırakma saldırılarına karşı (DoS, DDoS), istatistiksel ve manuel verilmiş sınırları işleterek koruma sağlayabilirler.

#### 3.2.3.3 Web Uygulama Güvenlik Duvarı

Web uygulama güvenlik duvarları, saldırı engelleme sistemlerine benzer bir görev üstlenirler. Web hizmetlerinin çok yaygın kullanılması sebebi ile üretilen bu sistemler, web hizmetlerine ve web sunucularına gelebilecek saldırıları önleyecek trafik imzaları bulundururlar. Bu özelliklerine ek olarak yazılım geliştirilirken önlem alınmamış konularda ek koruma getirebilirler.

#### 3.2.3.4 Veri Tabanı Güvenlik Duvarı

Günümüzde hacimleri oldukça artan ve sayısız sorgunun çalıştırıldığı veri tabanlarına gelen sorguların incelenmesi ve olası kötücül faaliyetlerin önlemlerinin alınması gerekmektedir. Veri tabanı güvenlik duvarları bu amaçla kullanılmaktadır. Veri tabanı güvenlik duvarları, kullanıcıların genel davranışlarını öğrenerek bunların dışına çıkılması halinde uyarı üretebilen sistemlerdir. Veri tabanı güvenlik duvarı ile web güvenlik duvarlarının birlikte kullanılması sayesinde sistem kullanıcılarının web üzerinde gerçekleştirdiği eylemlerin izdüşümleri takip edilebilmektedir.

#### 3.2.3.5 E-posta Güvenliği

Kurum ve kuruluşların sistemlerine dışarıdan gelebilecek SPAM ve zararlı kod içeren e-postaların engellenmesi amacıyla kullanılırlar. E-posta güvenliği (gateway), ağ üzerinde İnternet erişimi olan bir mail sunucu (Mail Transfer Agent, MTA) görevi ile de kullanılabilmektedir. Ayrıca mail sunucular üzerinde çalışabilen çeşitleri de mevcuttur.

#### 3.2.3.6 Yük Dengeleyici

Bilişim sistemlerinde işlem hacmi oldukça kritik seviyelere ulaşabilmektedir. Yük dengeleyiciler, artan trafiği, yoğun istek gelen sunucular arasında yükü paylaştırarak rahatlatırlar. Örneğin söz konusu bir web sunucu ise yük dengeleyiciler, SSL'i kendi üzerinde sonlandırır ve diğer sunucuları şifreleme yükünden kurtarıp performans artışı sağlayabilmektedir.

#### 3.2.3.7 URL Filtresi ve Antivirüs

Bilgisayar ağlarında çalışan istemci makinelerin İnternet erişimlerinin düzenlenmesi için kullanılmaktadırlar. Kurum politakaları gereği erişim yasağı bulunan sitelerin engellenmesi gibi kritik işlemleri gerçekleştiren yapılardır. Bu eylemlerin yanı sıra bu yazılımlar eğer vekil sunucu olarak çalışıyorlarsa ağ üzerindeki gelen trafiği kontrol edip zararlı yazılım analizi de yapabilmektedirler.

#### 3.2.3.8 Web Cache Vekil Sunucu

Web Cache vekil sunucuları, URL filtreleri ile birlikte kullanılabildikleri gibi tek başlarına da kullanılabilmektedir. İnternet üzerinden aynı dosyanın birden çok kez indirilmesi durumunun önüne geçilmesi amacıyla kullanılmaktadır. Web Cache vekil sunucuları, indirilen dosyaları kendi üzerlerinde tutarak İnternet bant genişliği açısından tasarruf sağlamış olurlar.

#### 3.2.3.9 Transparan İçerik Yönlendiriciler

Transparan içerik yönlendiriciler, kompleks ağ yapılarında ağa bağlı istemcilerin URL filtrelemesi gibi trafik yönlendirmesi gerektiği durumlarda, kullanıcı tarafında herhangi bir ayarlama yapılmadan işlemin gerçekleştirilmesini sağlayabilirler. Böylece kullanım kolaylığı sağlanmış ve ayarların eksik yapılması durumu ortadan kalkmış olur. Ayrıca kullanıcıların istenilen vekil sunucuları kullanması garanti edilmiş olur.

#### 3.2.3.10 Zafiyet Tarama Sistemleri

Bu yazılımlar, sistem üzerindeki açıklıkların tespit edilmesi amacıyla kullanılan yapılardır. Zafiyet tarayıcı sistemler, işletim sistemlerindeki ve işletim sistemi üzerinde çalışan uygulamalar içindeki açıklıkları otomatik taramalar ile bulabilmektedir. Bu sistemler ayrıca yama eksiklikleri ve kurum/kuruluş güvenlik politikalarına aykırı durumları tespit edebilme yeteneğine de sahiptirler.

#### 3.2.3.11 Risk Analiz Yönetim Sistemleri

Risk analiz yönetim sistemleri, ağ üzerinde bulunan zafiyet tarama sistemlerinden sistem açıklıklarını; güvenlik duvarı, anahtar ve yönlendirici gibi ağ aktif cihazlarından da yapılandırma ayarlarını toplayıp bir ağ modeli oluşturabilen yapılardır. Oluşturulan bu ağ modeli ile risk analizleri gerçekleştirilir ve bu riskler önceliklerine göre derecelendirilir. Böylece sınırlı iş gücü kaynaklarının nerede ve hangi durumda ilk önce kullanılması gerektiği ve en çok riske sahip olan sistemler gibi kritik bilgiler elde edilebilmektedir.

#### 3.2.3.12 Kayıt Toplama ve Korelasyon Sistemi

Ağ üzerinde birçok cihaz ve sistem kendi üzerinde meydana gelen aktiviteleri barındıran olay kayıtlarını tutmaktadır. Bu kayıtlar, her sistem üzerinde ayrı ayrı formatlarda bulunabildiğinden bunların ilişkilendirilmesi çok zor olmaktadır. SIEM (Security Information and Event Management) adı verilen bu kayıt toplama ve korelasyon sistemleri sayesinde dağıtık halde bulunan bu kayıtlar, merkezi bir yerde toplanıp ilişkili hale getirilebilmektedir. Yazılan mantıksal kurallar sayesinde gerçek zamanlı korelasyon yapılabilir ve böylece normalde tespit edilemeyen güvenlik olayları tespit edilebilir.

#### 3.2.3.13 Ağ Erişim Kontrolü

Ağ erişim kontrolü sistemleri, kurum politikalarına uymayan sistemlerin ağa dahil olmalarını engellemek amacı ile kullanılır. Bu sayede yabancı sistemlerin ve güvenlik durumu uygun olmayan sistemlerin iç ağı tehdit etmesi önlenir.

#### 3.2.3.14 Sıfır Gün Zararlı Yazılım Tespit Sistemi

Kötüye kullanım tespiti mantığı ile çalışan imza tabanlı saldırı tespit sistemleri (antivirüs vb.), kendi imza veri tabanlarında olmayan zararlı yazılımları yakalayamamaktadır. Bu durum, özellikle günümüzde artan bir hacime sahip kötücül yazılımların varlığı düşünüldüğünde oldukça kötü sonuçlar doğurabilmektedir. Bugün ortaya çıkartılan yeni bir kötücül yazılımın, ilgili tespit sisteminin imza veri tabanında yer alması uzun süreler alabilmektedir. Arada geçen süre zarfında sistemler yeni çıkartılan kötücül yazılımlara karşı savunmasız kalmaktadır. Sıfırıncı gün zararlı yazılım tespit sistemi, şüpheli gördüğü yazılımları ağ seviyesinde yakalayıp test sistemlerinde çalıştırırlar [58]. Bu testler sonucunda imzasız olarak zararlı olduğu tespit edilen yazılımların engellenmesi sağlanmaktadır.

#### 3.2.3.15 Ağ İzleme ve Performans Analiz Sistemi

Bu sistemler, ağ üzerinde ağ trafiğini izleyip, uygulamalar ve ağ performansı hakkında bilgi toplayan yapılardır. Ağ izleme ve performans analiz sistemlerinin kullanım amacı, performans kaybı durumlarında sistem yöneticisini bilgilendirmek ve sorunun kaynağı hakkında detaylı bir raporlama sağlamaktır [59].

#### 3.2.3.16 Veri Kaçakları Önleme Sistemi

Kurumsal açıdan önem arz eden hassas verilerin izinsiz olarak kurum dışına çıkartılmasına engel olan sistemlerdir. İstemci ve ağ düzeyinde çalışan farklı modelleri

mevcuttur. İstemci üzerinde çalışan sistemlerde, taşınabilir cihazlardan oluşabilecek veri kaçaklarının önlenmesi için aygıtsal kontrol yapan bileşenler bulunmaktadır.

#### 3.2.3.17 Ağ Tabanlı Adli Bilişim Sistemi

Bu sistemler pasif olarak ağ trafiğini yakalayarak trafik üzerinde derin paket incelemesi yapabilme imkânı sağlarlar [60]. Bu sistemler, bilişim sistemlerindeki bilgilerin mahkemede suçluluğun veya suçsuzluğun ispatında kullanılmak üzere incelenmesini sağlamaktadır. Ağ tabanlı adli bişim sistemleri, sayısal verilerin elde edilmesi, korunması ve analiz edilmesi işlemlerinin kanıtın gereklerine uygun olarak adli makamlara sunulması aşamasına kadar uygulanması sürecini yönetir.

#### 3.2.3.18 Tek Yönlü Veri Transfer Cihazları

İnternet erişimi olmayan ağlara veri transferi yapıldığı zaman dış dünyaya veri sızmasını engellemek amacı ile kullanılan cihazlardır. Donanım tabanlı tek yönlü veri transfer cihazları, kullanılan protokolü her iki tarafa da çalışırmış gibi gösterip donanım üzerinde bir yön dışında ters tarafa veri iletişimini fiziksel olarak engelleyen yapılardır [61].

#### 3.2.3.19 İstemci Güvenlik Ürünleri

İstemci tarafında çalışan güvenlik ürünleri, ağ seviyesinde çalışan araçlara destek niteliğindeki yapılar olup ek bir katman gibi görev yaparlar. İstemci güvenlik ürünlerine örnek olarak, antivirüs, saldırı engelleme sistemleri, veri kaçakları önleme yazılımları ve disk şifreleme yazılımları örnek olarak verilebilir [56].

## 3.3 Saldırı Tespit Sistemleri

*Saldırılar*, sezgisel olarak bir bilgi sistemi içerisinde, sistemin güvenlik politikalarını ihlal eden etkinliklerdir. *Saldırı tespiti* ise saldırıların türünü, şiddetini, ihlal ettiği zafiyet vb. durumları tespit etmeye, tanımlamaya yönelik olarak geliştirilen işlemlerdir [7]. *Saldırgan*, bilişim sistemlerindeki zafiyetlerden yararlanarak sistem üzerindeki emellerini gerçekleştirmek üzere saldırıyı planlayan ve gerçekleştiren kişidir. *Saldırı tespit sistemleri* ise, saldırıları tespit etmeye yönelik olarak geliştirilen işlemleri bünyesinde barındıran yazılımsal veya donanımsal çözümlerdir. Anderson'un [7], konunun temelini oluşturan ve 1980 yılında bir rapor olarak sunduğu bu tanımdan günümüze kadar literatürde birçok tanım yapılmış ve konuyla ilgili olarak birçok yazılım gerçekleştirilmiştir. Bu tez çalışması, tüm bu yapıların analizine odaklanmaktadır.

Bilişim teknolojilerinin hızlı gelişimi sonucu, elektronik ortamların kullanım oranının gün geçtikçe artması ve sağladığı kolaylıkların yanında, bu ortamlarda saklanan bilgilerin güvenliğinin sağlanması da çok önemli bir ihtiyaç haline gelmiştir. Bilgi ve bilgi sistemlerinin korunmasındaki temel amaç, saldırganlara ve saldırılara karşı önlem alarak, bilginin mahremiyetinin korunmasıdır. Teknolojik ilerlemelerle beraber her yeni teknoloji için bazı açıklıklar olabilmekte ve bu da bilişim sistemlerine yönelik sürekli ve yeni tehditleri beraberinde getirmektedir. Bu sistemlerdeki açıklık ve zayıflıklar olduğu sürece saldırıları tespit etmek sistem güvenliğinin sağlanması açısından oldukça önemlidir [8].

Bilgiyi korurken, var olan sistemlerin sürekliliğinin sağlanması hayati önem taşımaktadır. Bilginin erişilebilirliği, bilgi güvenliğinin temel kavramlarından biridir. Bu sürekliliği korumak için, yapılan saldırılara karşı alınan önlemlerin de güncelliğini koruması gerekmektedir. Bu durum da değişen saldırı ve yöntemlerin bilinmesi ve var olan sisteme adapte edilmesi ile sağlanabilir. Yeni saldırı yöntem ve tekniklerine göre yeni savunma mekanizmaları geliştirilmeli ve sistem koruması sağlanmalıdır. Bilişim sistemlerinde bu hızlı değişim sebebiyle saldırı, saldırgan, saldırı tespiti gibi temel kavramların ne anlama geldiği çok iyi bilinmelidir. Bu nedenle güvenlik konusunda yapılan çalışmalarda saldırının birçok tanımı yapılmıştır. Anderson'a göre bir saldırı, izin almadan bilgiye ulaşım, bilgiyi değiştirme ve sistemi kullanılmaz veya güvenilmez hale getirmektir [7].

Günümüzde ise saldırı, "bilginin mahremiyetini, bütünlüğünü ve erişilebilirliğini tehlikeye atabilecek girişimlerin kümesi" olarak tanımlanmaktadır [8]. Saldırı ifadesi ile sızma kavramı birçok çalışmada benzer olguları karşılamaktadır. Bu tezde bu iki kavram için de "saldırı" ifadesi kullanılmıştır. Yapılan çalışmalarda saldırı olarak nitelendirilen bazı olgu örnekleri şöyle verilmiştir [8, 9, 69];

- Uzaktaki bir e-posta sunucusunun kök dizin erişim uzlaşmasının sağlanması
- Bir web sunucunun yapısının bozulması
- Şifrelerin tahmin edilmesi ve çözülmesi
- Kredi kartı numaralarını içeren bir veri tabanının kopyalanması
- Bordro kayıtları, hastaya ait özel bilgiler gibi hassas verilere yetkisiz kişilerce erişilmesi
- Kullanıcı adı ve şifrelerin elde edilmesi için bir iş istasyonu üzerinde paket koklayıcı vb. kötücül yazılımların koşturulması
- Anonim bir FTP sunucu yapısındaki yetkilendirme izni hatasını kullanarak sunucudaki yazılım veya mp3'lerin dağıtılması

- Güvensiz bir modem üzerinden dahili bilgisayar ağına erişim hakkı sağlanması
- Bir yönetici kılığında yardım masasını arayarak yeni bir şifrenin alınması
- Keylogger benzeri çeşitli yapılarla şifrelerin elde edilmesi
- Kötücül yazılımlarla, sistem üzerinde denetimlerin gerçekleştirilmesi
- Sosyal mühendislik yöntemleri ile istenilen bilgilerin elde edilmesi.

Saldırı tespiti, bir bilgisayar sisteminde veya ağda meydana gelen olayları izleyerek, bilginin mahremiyetini, bütünlüğünü ve erişilebilirliğini bozmak ya da sistemin güvenlik mekanizmalarını aşmak için yapılan hareketleri analiz etme ve tanımlama işlemidir [8].

Saldırgan, en temel ifadeyle sisteme sızmaya çalışan veya saldırı işini gerçekleştiren kişiye verilen addır. Saldırganlar, sistemle olan ilişkilerine göre 3'e ayrılmaktadır [7, 9]. Bunlar;

**Gerçeği Gizleyen (The Masquerader):** Sistemi kullanmaya yetkisi olmayan, ancak sistemde yetkili bir kullanıcının hesabını kullanarak sisteme sızan kişidir.

**Yasal Kullanıcı (The Legitimate User):** Sistem kaynaklarına erişebilen, ama sistem üzerindeki haklarını kötüye kullanan yasal bir kullanıcı.

**Gizli Kullanıcı (The Clandestine User):** Sistemde yönetici haklarını ele geçiren kişi. Tespit edilmesi ve yakalanması en zor saldırgan tipidir. Çünkü sistem üzerinde sahip olduğu yönetici hakları sayesinde kontrollerden kurtulur.

En genel tanımıyla saldırı tespit sistemleri ise, bilgi güvenliği sistemlerinde, bilginin korunmasını ve erişilebilir haldeki sürekliliğinin sağlanması için sisteme yapılabilecek olası saldırıları tespit ve tanımaya yönelik olarak geliştirilmiş sistemlerdir. Literatürde yer alan bazı STS tanımları aşağıda verilmiştir.

- STS, bilgi güvenliğinin sağlanmasında bizlere yardımcı olan sistemlerdir [9].
- STS, bir ağ veya bir bilgisayara karşı yapılan her türlü saldırının tespit edilmesi ve saldırganın bertaraf edilmesi için geliştirilmiş sistemlerin bütününe denir [10].
- STS, gerçek zamanlı, sistem bütünlüğünü, kullanılabilirliğini ve gizliliğini tehdit eden aktiviteleri ayırt ederek bildiren sistemlerdir [62].
- STS, sistemleri yetkisiz kullanma ya da yetkilerini aşan işlemleri yapma girişiminde bulunan kişi veya programları tespit etme çalışmasıdır [63].
- STS, son yıllarda ortaya çıkmış yeni bir ağ teknolojisidir. Yazılım ve donanım kombinasyonundan oluşur. Firewall'ın yetersiz kaldığı durumlarda gerekli koruma ölçütlerini sağlar ve saldırılara karşı etkin bir koruma gücüne sahiptir [64].

- STS, bilgisayar sistemlerindeki sızmalara karşı, bu sızmaları tespit etmeye çalışan ve sistem izolasyonunu sağlayan yeni bir ağ teknolojisidir [65].
- STS, bir ağda düşman (hostile) ve sömürücü (exploit) aktivitelere karşı koruyucu sistemdir [66].
- STS'ler bilişim sistemlerine yapılan yetkisiz erişimleri tespit etmek için kullanılan yazılım araçlarıdır. STS'ler kötü niyetli ağ trafiği ve bilgisayar kullanımını tespit etme yeteneğine sahiptir. Bir STS, olası güvenlik açıklarını belirleyebilmek için bilgisayar veya ağ içerisinde değişik alanlardan bilgileri toplar ve analiz eder. STS'ler, güvenlik duvarının statik izleme kabiliyetini tamamlayan dinamik izleme elemanlarıdır [67].

Literatürdeki bu tanımları dikkate aldığımızda, STS'leri, bilginin elektronik ortamlarda taşınırken, işlenirken veya depolanırken oluşabilecek tehdit ve tehlikelerin ortadan kaldırılması amacıyla, bilgiye yetkisiz erişim veya bilgiyi kötüye kullanım gibi girişimleri tespit edebilme ve bu tespiti sistem güvenliğinden sorumlu kişilere iletebilme özelliğine sahip yazılımsal ya da donanımsal güvenlik araçları olarak tanımlayabiliriz. STS'ler, ağ cihazlarını izleyerek trafiği kontrol edip anormal davranışları ve kötüye kullanımı tespit edebilme yeteneğine sahiptirler.

Güvenlikle ilgili çalışmaların temeli 1970'lere varmaktadır. Buna karşın, STS kavramı ise ilk olarak bir önceki kısımda temel tanımlar yapılırken de yararlanılan Anderson'un "Bilgisayar Güvenliği Tehdit Gözetleme ve İzleme" (Computer Security Threat Monitoring and Survaillance) makalesi ile 1980'de ortaya atılmıştır [7]. Bu çalışma, STS'lerin tanımlanması ve tanınması açısından büyük bir öneme sahiptir.

İlk nesil STS'ler, basit bilgisayar sistemleri üzerine düşünülmüştür. İkinci nesilde ise günümüzde STS'lerin vazgeçilmezi olan "denetleme izi (audit trail)" kavramı ve veri tabanı mantığının bilgi güvenliği alanındaki önemi ortaya çıkmıştır. Sonraki çalışmalarda, güvenlik konusundaki çalışmalara yardımcı olmak amacıyla günlük denetleme verilerinin otomatik araçlar ile elde edilmesi konusunda çalışmalar yapılmıştır [8]. 1985'ten itibaren bu amaçla geliştirilen projelerde denetleme verileri üzerinde özenle durulmuş ve istatistiksel yaklaşımlar temel alınarak saldırı tespit modelleri geliştirilmiştir.

IDES (Intrusion Detection Expert System), bu çalışmalardan biri olup Denning tarafından geliştirilmeye başlanan ve 1988-1992 yılları arasında yapılan çalışmaların birçoğunu

üzerinde barındıran bir sistemdir. Daha sonra ismi NIDES (Next-Generation Intrusion Detection Systems) olarak değişen bu sistem, ikinci nesil STS'lerin en eski ve en bilinen çalışmalarından biridir. IDES, saldırı senaryolarının kural kümelerinin çıkarılmasıyla tasarlanmaya başlanan kural tabanlı bir STS'dir. Denning 1987 yılında yayınladığı çalışmasında üç farklı istatistiksel model tanımlamıştır. Bu modeller;

- Kullanıcının belirli aralıklarla bir işlemi tekrar etmesine izin veren ve eşik değerine göre anormallik durumlarını tespit eden model,
- İstatistiksel momentlerin bilindiği varsayılarak tespit edilen sapmalar ile anormallik olduğunu tespit eden model ve
- Anormalliklerin tek olaya değil bir diziye bağlı olduğu Markov modeli olarak verilmektedir [8, 68].

STS'lerin geliştirilmesinde günümüze kadar istatistiksel yöntemlerin dışında, kural tabanlı (rule based), eşik değeri belirleme (threshold value), durum geçiş diyagramları (state transition diagrams), ensemble teknikleri, yapay sinir ağları (artificial neural networks), veri madenciliği (data mining), metin madenciliği (text mining), karar ağaçları (decision trees), destek vektör makinaları (support vector machines), yapay bağışıklık sistemi (artificial immune system), bulanık mantık (fuzzy logic) gibi farklı birçok yaklaşım uygulanmıştır [8, 9]. Bu yaklaşımlara, "STS'lerde Kullanılan Yöntemler" başlıklı kısımda ayrıntılı olarak değinilmiştir.

### 3.3.1 Bazı Saldırı Türleri ve Özellikleri

Bu bölümde bilişim sistemlerine yönelik olarak gerçekleştirilebilen çeşitli saldırı karakteristikleri, türleri ve bu saldırıların özellikleri anlatılmaktadır. Bilişim sistemleri günümüzde, kamu ve özel sektörde özellikle; bankacılık, ticaret, eğitim, ulaşım gibi birçok alanda yaygın olarak kullanılmaktadır. Bu yaygın kullanım insanlara birçok açıdan yarar sağlamasına rağmen birtakım önlemleri almayı da zorunlu kılmaktadır. Çünkü teknoloji geliştikçe saldırı türleri de artmaktadır. Hatta hiçbir bilgi düzeyi gerektirmeyen, bir sohbet programının kullanım kolaylığına sahip saldırı araçları ile çocuk yaştaki kişiler bile otomatik olarak bazı saldırıları gerçekleştirebilmektedir. Yapılan saldırıların neden gerçekleştirildiği hususunda da çok çeşitli sebepler karşımıza çıkmaktadır. Bu sebepler arasında [62, 69, 70];

- Politik nedenler,
- Meydan okuma,
- Web sitelerinden intikam alma isteği,

- Vatanperverlik/Milliyetçilik düşüncesi,
- Eğlence amaçlı,
- İtibarsızlaştırmak,
- Terörist propaganda yapmak,
- Kin,
- Merak,
- Maddi kazanç elde etme isteği,
- Ün kazanma isteği,

gibi birçok neden sıralanabilir. Bu saldırıların zaman içerisinde artmasının sebepleri olarak da şunlar sayılabilir;

- Globalleşme,
- İstemci-sunucu mimarisine yönelme,
- Bilgisayar korsanlarının hızlı öğrenmesi,
- Geliştirilen otomatik saldırı araçları ile kolayca saldırı yapılabilmesi,
- Geliştirilen yeni yazılım ve donanım sistemlerinin zayıflıklarından kaynaklanan güvenlik açıkları [9].

Bilgisayar ağ sistemlerinde ve ağ iletişiminde normal ve sistem tarafından istenen bilgi alışverişine yönelik olarak yapılan saldırılar dört karakteristik yapıya sahip olabilmektedirler. Bunlar; gizli dinleme (intercept), kesme (interruption), değiştirme (modification) ve üretim/imalat (fabrication) olarak karşımıza çıkmaktadır. Şekil 3.4'te bu saldırı karakteristikleri gösterilmektedir. Burada ağ trafiğinin normal seyri kaynak (source) makine ve hedef (destination) makine arasındadır. Saldırgan ise bu iletişime farklı şekillerde dahil olan bir başka makine veya aygıttır. Bu saldırı karakteristiklerini kısaca değerlendirecek olursak [70];

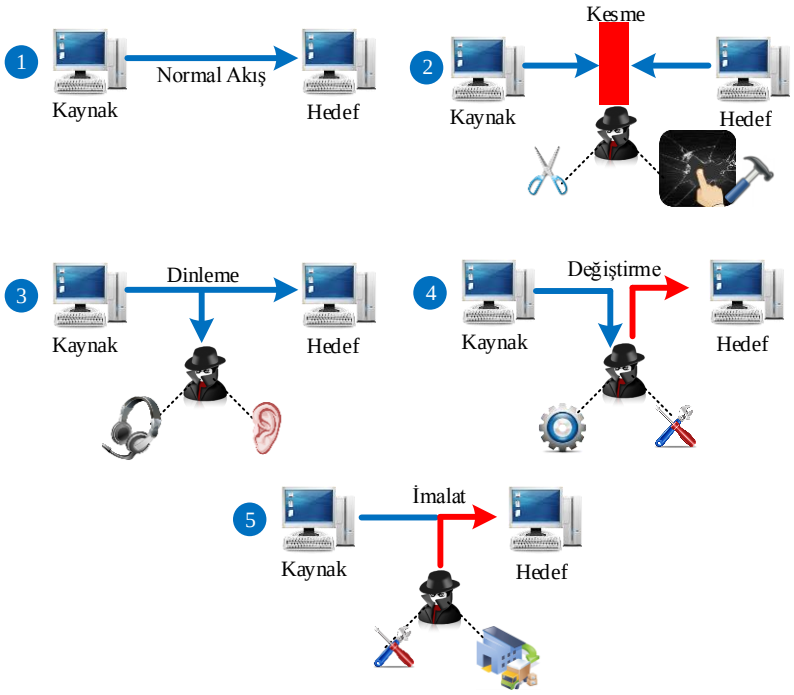
**Yarıda kesme:** Kaynak ve hedef arasındaki bilgi akışına direk olarak engel olunup iletişim koparılır. Bu yapıda sistemin değerli varlıkları yok edilir, erişilemez ya da elde edilemez hale getirilebilir. Amaç iletişimin engellenmesidir. Bu sebeple bu saldırı karakteristiği fiziksel bir müdahale sonucu da olabilmektedir. Bu saldırı tipinin hedeflediği güvenlik unsuru, bilgi güvenliği kavramı olarak kullanılabilirlik (availability) özelliğidir. Donanımsal yıkım, iletişim hatlarına fiziksel zarar verme, iletişim ortamında gürültü yayma, rota (routing) şaşırtma, program ve dosya silme, hizmet aksattırma (DoS), çeşitli kötücül

yazılımlarla iletişim halindeki cihazlara zarar verme şeklinde gerçekleştirilebilir. Etkin ve kolay bir çözüm yöntemi yoktur.

**Gizli dinleme**: Diğer saldırı karakteristiklerinin aksine pasif saldırı olarak nitelendirilir. Diğer saldırı türleri ise aktif saldırı olarak isimlendirilmektedir. Bu saldırı türünde sisteme erişim yetkisi bulunmayan saldırgan, sistemi gizli bir biçimde dinleyerek kaynak ve hedef arasındaki bilgiyi elde eder. Bu saldırı türü, güvenlik unsurlarından gizlilik (confidentiality) unsurunun ihlaline yönelik bir durum doğurur. Hat izleme, paket yakalama ve sistemle uzlaşma gibi yaklaşımlarla gerçekleştirilir. Bu saldırı türü için şifreleme/şifre çözme yöntemiyle çözüm aranmaktadır.

**Değiştirme:** Kaynak ile hedef arasında iletimdeki verinin elde edilerek bu veri üzerinde istenilen değişikliğin yapılması ve değiştirilmiş verinin kaynaktan gönderiliyormuş gibi hedefe gönderilmesidir. Burada saldırgan, sistem üzerinde erişim hakkı kazanmakla kalmamakta, iletişimin değerli bir unsurunu da değiştirmektedir. Bu saldırı türü, güvenlik unsurlarından bütünlük (integrity) kavramının ihlaline yönelik bir saldırıdır. Saldırı, veri tabanı kayıtlarını değiştirme, iletişimdeki yavaşlama ve gecikmelerden yararlanma, donanımsal değişiklikler yapma şeklinde işlemler gerçekleştirebilmektedir. Bu saldırı türü için, iletilecek her paket için dijital imza kullanımı ile çözüm aranmaktadır.

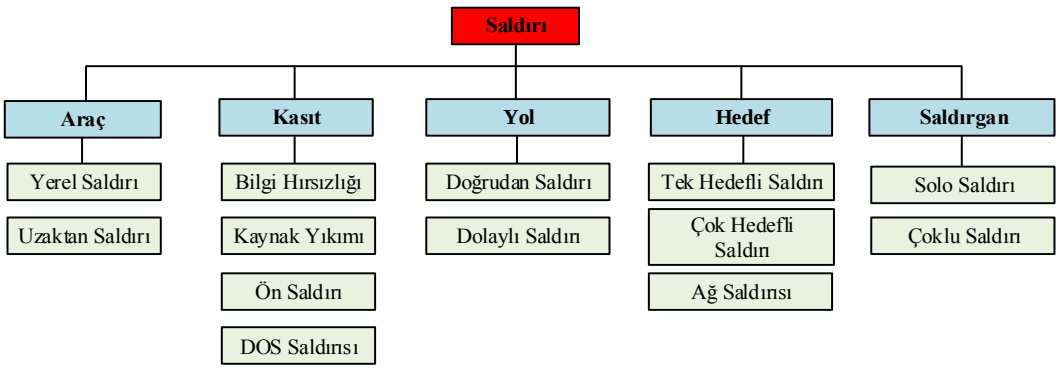
**İmalat:** Sisteme yetkisiz kişiler tarafından sızılarak kaynaktan geliyormuşçasına hedefe yapay veriler gönderilmesidir. Burada veri tabanına kayıt ekleme, IP gizleme yöntemiyle ağ iletişim ortamına yeni ağ paketi ekleme, geçersiz e-posta ve domain isimleri kullanarak sistemi kötüye kullanma, sosyal mühendislik yöntemleriyle sistemin geçerli kullanıcılarını yanlış işlemlere yöneltme, sistem erişim haklarını kötüye kullanarak geçersiz veya sahte veri üretimi şeklinde işlemler yürütülebilir. Bu saldırı türü için kimlik doğrulama metodu çözüm olarak önerilmektedir.




**Şekil 3.4** Saldırı karakteristikleri [70]

Şekil 3.4'te, (1) normal, (2) yarıda kesme, (3) gizli dinleme, (4) değiştirme, (5) imalat karakteristiklerini temsil etmektedir.

Saldırılar farklı kategorilerde sınıflandırılabilirler. Bu kategorilendirmede saldırgan sayısı, kullanılan yol, hedef türü, saldırı amacı ve kullanılan araçlar birer parametre olabilir. Saldırıların genel bir tasnifi Şekil 3.5'te verilmiştir.




**Şekil 3.5** Saldırıların sınıflandırılması [70]

Bilgisayar sistemlerine yönelik olarak gerçekleştirilebilecek birçok farklı saldırı olgusu vardır. Bunlar, kaynak kod istismarı (code exploit), gizli dinleme, hizmet aksattırma (DoS), arka kapılar (backdoor), sosyal mühendislik, dolaylı saldırılar, kriptografik saldırılar, yönetici hesabı ile oturum açma (R2L), kullanıcı hesabının yönetici hesabına yükseltilmesi (U2R), bilgi tarama (probe&scan), tahrip etme (vandalism), özel veri hırsızlığı, doğrudan

erişim saldırıları, dolandırıcılık ve suistimal, log dosyalarının silinmesi ya da değiştirilmesi ve güvenlik mimarisinin değiştirilmesi saldırılarıdır [70].

#### 3.3.1.1 Kaynak Kod İstismarı

Sistemde kullanılan (işletim sisteminin kullandığı sistem programları dahil) tüm yazılımlarda var olabilecek arabellek taşması (buffer overflow), CGI betikleme (scripting) hataları ve şifreleme hataları gibi yazılım kusurları, bir bilgisayar sisteminin kontrolünün ele geçirilmesine ya da o bilgisayarın beklenmedik bir şekilde çalışmasına neden olabilmektedir. Bu durum, son yıllara kadar gözden kaçırılan bir unsur olarak, birçok saldırıya açık kapı bırakan bir korunmasızlıktır [70]. Kod istismarı, bir yazılım ya da donanım sistemi üzerinde hata, aksaklık ya da güvenlik açığı oluşumlarından yararlanan, bir yazılım parçası, bir veri yığını veya komut dizinidir.

#### 3.3.1.2 Gizli Dinleme

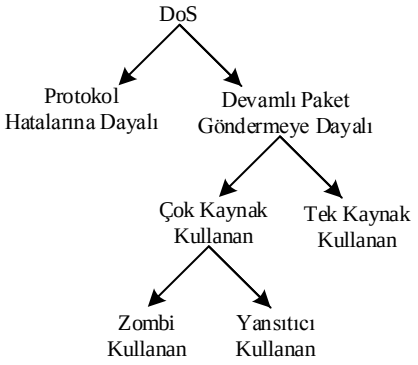
Bilgisayar ağlarında veri iletişimi esnasında kaynak ve hedef arasında iletilen verinin kötü niyetli kişiler tarafından elde edilmesidir. Bu saldırı türünde elde edilen veri hedefe değiştirilmiş bir şekilde bile iletilebileceğinden oldukça tehlikeli bir saldırı türüdür. Herhangi bir ağa dahil olmayan bir bilgisayar, donanımsal parçalarından yayılan elektrik yayılım takip edilerek gizlice dinlenebilmektedir. Geliştirilen donanımsal cihazların bu tür dinlemelere olanak vermemesi için *Tempest* adı verilen bir standarda uygun üretilmeleri gerekmektedir [70].

#### 3.3.1.3 Hizmet Aksattırma

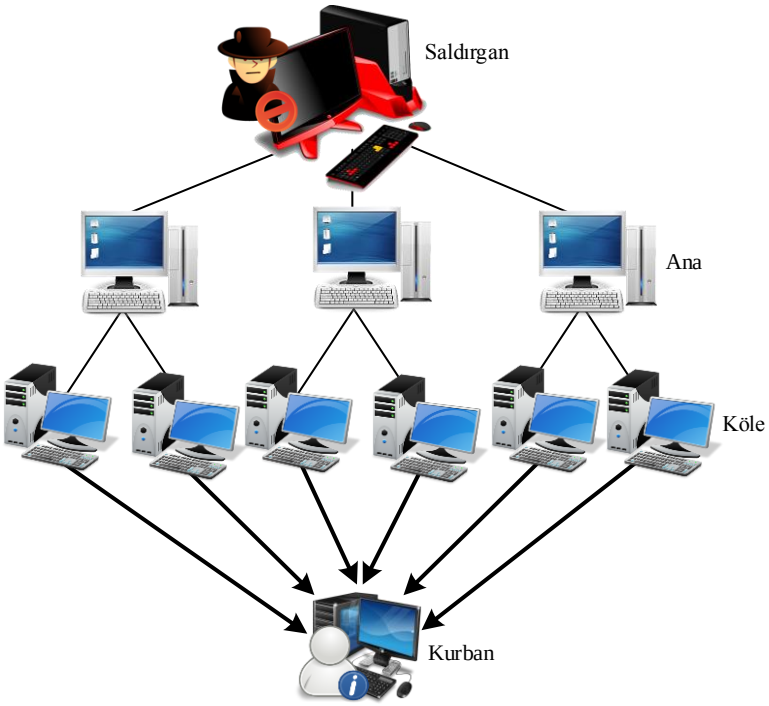
Bir sunucuya çok fazla sayıda istekte bulunmak suretiyle sunucunun iş göremez hale gelmesini amaçlayan bir saldırı türüdür. Bu haliyle hizmet aksattırma (DoS) saldırıları, yetkisiz erişim veya sistem kontrolünü ele geçirmeye yönelik olarak yapılan saldırılardan farklıdır. Bu saldırı türündeki tek amaç, bilgisayar, sunucu veya ağın kaldırabileceğinden çok daha fazla yük bindirilerek sistemin kullanılmaz hale getirilmesidir.

DoS saldırıları kendi içinde gruplara ayrılmaktadır. “the ping of death” (ölümüne ping) saldırısı protokol hatalarından yararlanılarak yapılan bir saldırı türüdür. Bu saldırı türünde bir tek büyük boyutlu ICMP eko mesajı gönderilerek sistemin cevap verememesi sağlanmış olur [10].

Diğer bir DoS saldırı türü ise TCP SYN paketinin içerisine kaynak ve varış adresi aynı makine olan bir paket gönderilmesiyle olur. Bunlar tek paketle ya da az paketle gerçekleştirilebilen, protokollerin açıklarını kullanan saldırılardır. Bu durum, Şekil 3.6'da gösterilmektedir. Diğer gruptaki saldırılar devamlı istekte bulunulmasına dayanır. Hem sunucu makine hem de ağ meşgul edilir. Örneğin bu, bir sunucuya devamlı bağlantı isteği yapmak olabilir. Saldırı, tek bir makine kaynaklı olabileceği gibi ağ üzerinden ele geçirilen birçok makine ile de yapılabilir. Bu tip makinelere zombi denir. Eğer zombi kullanılmıyorsa yansıtıcı da kullanılabilir. Yansıtıcı herhangi bir sunucu olabilir. Şekil 3.7, dağıtık hizmet aksattırma saldırısının senaryosunu göstermektedir.




**Şekil 3.6** DoS saldırı türleri




**Şekil 3.7** Dağıtık hizmet engelleme saldırısı

#### 3.3.1.4 Arka Kapılar

Kötü amaçlı yazılım türlerinden biri olan arka kapılar (backdoor), sistemlere uzaktan erişmeyi mümkün kılmak üzere oluşturulmuş yapılardır. Arka kapıların sıradan incelemelerle tespit edilme olanağı yoktur. Arka kapılar, kurulu bir program şeklinde olabilir veya bilinen bir programın içerisine programın üreticisi tarafından gizli bir şekilde yerleştirilmiş olabilir. Arka kapı saldırılarında truva atı programları sıklıkla kullanılmaktadır. Arka kapılar, bilgisayar üzerinde normal kimlik kanıtlama süreçlerini atlamayı veya kurulan bu yapıdan haberdar olan kişiye o bilgisayara uzaktan erişmeyi sağlayan yöntemlerdir. Bir sisteme sızmak için oldukça zahmetli bir çaba harcayan korsanlar, daha sonra aynı sisteme erişmek için daha kolay bir yolu sisteme eklemek isterler. En sık karşılaşılan arka kapı yöntemi, hedef sistemde, dinleme ajanı iliştirilmiş bir kapıyı (port) açık tutmaktır. Bu açıdan bakıldığında, bu tür bir açığa maruz kalınmadığından emin olmak için, sistemde mevcut bulunan bütün kapılar, 1'den 65535'e kadar, iki kere (bir kez TCP bir kez de UDP için) taranmalıdır. Hem arka kapılar hem de truva atları, hedef sisteme sızmaya yarayan kötü amaçlı yazılımlar olup; Truva atı, faydalı bir program gibi gözükürken; arka kapı, sadece sisteme erişimi sağlayan gizli yapılardır.

Arka kapılar kimi zaman, sistemi geliştiren programcı tarafından test edilen sisteme erişmek amacıyla kullanılan fakat daha sonra unutulan açıklar olarak karşımıza çıkmaktadır. Bu durumun farkına varan kötü niyetli kişiler, bu yapıları kullanabilmektedirler. Hatta bu tip arka kapılar bazen programcı tarafından kasten bırakılabilmektedir. Arka kapı konusunda en ünlü iddialardan biri Microsoft'un, Windows işletim sisteminin bütün sürümlerinde NSA (American National Security Agency) için bir arka kapı yerleştirdiği iddiasıdır. Bu iddianın ortaya atılmasının sebeplerinden biri Microsoft'un bütün sürümlerinde bulunan CryptoAPI yapısında, _NSAKey adına ilave bir giriş anahtarının bulunmasıdır [70, 72].

#### 3.3.1.5 Sosyal Mühendislik

Sosyal mühendislik veya bazen de toplum mühendisliği olarak karşımıza çıkan bu kavram, bilgi sistemleri güvenliği açısından önemli kavramlardan biridir. Bilgi güvenliği sistemlerinde en önemli unsurlardan birisi de insan faktörüdür. Sistemleri kullanan ya da yönetenler sonuçta bilgi düzeyi farklı seviyelerde olabilen kişilerdir. Bilgi sistemlerinde karşılaşılan birçok olay, insan faktöründen kaynaklanmaktadır. Bilerek veya bilmeyerek sebep olunabilecek en küçük bir hata çok vahim sonuçlar doğurabilir. İşte sosyal mühendislik, kötü niyetli saldırganın ilgilendiği sistem üzerinde, sistemde kayıtlı bulunan, sistemi yöneten veya kullanan kullanıcılar üzerinde bir takım psikolojik ve sosyal taktikler

kullanarak, sisteme erişmek için gerekli bilgileri elde etme yöntemlerine verilen genel bir isimdir. Bu duruma en tipik örnek telefon kullanımı ile kullanıcı adı ve şifre bilgilerini elde etmektir. Saldırgan, sıradan sistem kullanıcısıymış gibi sistem yöneticisinden sisteme giriş bilgilerini elde edebilir. Sosyal mühendislik konusunda birçok yöntem düşünülebilir ve tüm bu yöntem ve taktiklerden korunmanın en temel yolu, sistem kullanıcılarının sürekli olarak eğitilmesi ve yönetici ve kullanıcıların sistem güvenliği politikalarını uygulamasıdır [70].

Bilgi sistemlerinde meydana gelen birçok hata olgusu insan kaynaklıdır. Kurum içi kullanıcılar aynı zamanda sisteme en çok hasar veren kişilerdir. Bu durum, kurum içi bilgi güvenliği politikalarının ve eğitimlerin önemini vurgulamak açısından önemlidir. Bir başka durum da kurum veya şirket çalışanlarının işten ayrılmaları sonrasındaki durumdur. İşten ayrılan veya çıkartılan kişinin süreç sonrasında ayrıldığı oluşuma zarar vermek için saldırma olasılığına karşı ayrıca güvenlik önlemleri alınmalıdır.

#### 3.3.1.6 Dolaylı Saldırılar

Dolaylı saldırılar üçüncü parti bir bilgisayar tarafından başlatılan saldırılardır. Değişik türden saldırıları kapsamaktadır. Dolaylı saldırılarda zombi bilgisayar olarak adlandırılan başka birinin bilgisayarının kullanılması, gerçek saldırıyı yapanı bulmayı zorlaştırmaktadır.

#### 3.3.1.7 Kriptografik Saldırılar

Kriptografik saldırılar, şifreli bilgilerin şifrelerinin elde edilmesine yönelik olarak gerçekleştirilen saldırılardır. Bu saldırılar birçok kriptoanaliz metoduyla gerçekleştirilebilir. Bu metotlar arasında, sözlük saldırısı (dictionary attack), kaba kuvvet saldırısı (brute force attack), ortadaki adam saldırısı (man in the middle attack), sadece şifreli metin (chiper-text only), bilinen düz metin (known plaintext), seçilen düz metin veya şifreli metin (chosen plaintext, chipertext), adaptif seçili düz metin (adaptive chosen plaintext), ilişkili anahtar saldırısı (related key attack) gibi saldırıları saymak mümkündür [70]. Bu saldırılar aktif saldırı grubuna girerler. Saldırılar veya saldırılarda kullanılan araçlar, teknik açıdan gittikçe karmaşıklaşırken, bu saldırıyı gerçekleştirecek saldırganın sistem üzerinde ihtiyaç duyduğu bilgi seviyesi de o derecede azalmaktadır.

*Sadece Şifreli Metin Saldırısı:* Saldırganın mesaj içeriğini bilmediği ve yalnızca şifreli metinler kullanarak gerçekleştirilen bir saldırı türüdür.

*Bilinen Düz Metin Saldırısı:* Saldırgan şifreli metnin bazı kısımlarını kullanarak düz metin hakkında fikir sahibi olabilir. Bu saldırı türünde, saldırgan metni tahmin edebilir veya bölebilir. Bu adımdan sonra bu bilgiyi kullanarak şifreli metni çözebilir.

*Kaba Kuvvet Saldırısı:* Şifreli metnin bulunmasına yönelik bütün ihtimallerin denenmesi şeklindeki saldırı türüdür. En basit ve en uzun çözüm yoluna sahip kriptografik saldırı türüdür.

*Sözlük Saldırısı:* Bu saldırı türünde saldırgan, çeşitli sözcüklerden oluşan bir kelime listesini bir sözlük yapısında tutar. Saldırı yapılırken bu sözlükteki kelimeler şifreli metin ile karşılaştırılarak mesaj elde edilmeye çalışılır.

#### 3.3.1.8 Yönetici Hesabı ile Yerel Oturum Açma

Sistem üzerinde kullanıcı haklarına sahip olunmadığı halde, sisteme bazı paketler gönderilerek misafir ya da başka bir kullanıcı olarak sisteme erişim hakkı kazanılması şeklinde bir saldırı türüdür. Bilinen türleri aşağıdaki gibi özetlenebilir [9].

**Guest:** Bazı sistemlerde Guest hesabının şifresi boş bırakılmış veya tahmin edilebilecek basitlikte bir şifre verilmiş olabilir. Bu saldırı türünde bu şifre tahmin edilmeye çalışılır.

**Dictionary:** Kullanıcılar sistem üzerinde genellikle şifre olarak doğum tarihleri, isimleri, belirli kimlik, telefon vb. numaraları kullanmaktadırlar. Bu tür şifreler kötü niyetli kişiler tarafından tahmin edilebilir bir yapı arz eder. Bu saldırıda belirli bir sözlük yapısı ile şifreler tahmin edilmeye çalışılır.

**Imap:** Burada tampon bellek taşma hatası (Buffer overflow) verdirilerek hedef sisteme yönetici (root) olarak erişim yetkisi elde edilmeye çalışılır.

**Sendmail:** Sendmail posta sunucusunun güvenlik açıklarından yararlanarak bir mesaj gönderimi ile yönetici hakları üzerinden bazı komutlar işletilebilmektedir.

**Xlock:** Saldırgan tarafından kullanıcının kullandığı X terminalinin benzeri bir yapı gerçekleştirilir ve kullanıcı normal olarak sisteme girdiğini düşünürken giriş aşamasında şifreler elde edilir.

#### 3.3.1.9 Kullanıcı Hesabının Yönetici Hesabına Yükseltilmesi

Sistem üzerinde erişim hakkı olan fakat yönetici olmayan bir kişinin yönetici haklarını elde etmesi şeklindeki bir saldırı türüdür. Bilinen türleri aşağıdaki gibi özetlenebilir [9].

**Eject:** Tampon bellek taşması ile yönetici haklarına erişilmesidir.

**Fdformat:** Solaris İşletim Sistemi'nin 2.5 versiyonunda *PCMCIA* kartlarını biçimlendirmeye yarayan programdır. Bu saldırıda tampon bellek taşması ile yönetici haklarına sahip olunur.

**Perl:** Perl'deki bazı güvenlik açıklarının kullanılması yoluyla yönetici haklarının elde edilmesidir.

**Xterm:** Tampon bellek taşması yoluyla yönetici haklarının elde edilmesidir.

#### 3.3.1.10 Bilgi Tarama

Bilgi tarama (Probe/Scan) kabaca bir bilgi güvenliği sisteminin erişimcileri açısından taranması ve böylece çeşitli bilgilerin elde edilmesidir. Bilgi tarama ile ağdaki bilgisayar sayısı, sunucu veya herhangi bir bilgisayarın IP adresi, aktif portları, işletim sistemi, sistemdeki kullanıcı isimleri veya kullanıcılar ile ilgili bilgiler öğrenilebilmektedir. Bilinen bilgi tarama saldırı türleri aşağıdaki gibidir [9].

**Ipsweep:** Bir ağ yapısındaki bilgisayarların tespit edilmesine yönelik olarak bir saldırı türüdür. En çok bilinen ve en popüler olan yol, ağda bulunan tüm adreslere ICMP paketi gönderip bu paketlere cevap veren bilgisayarların belirlenmesi şeklindedir.

Bunun dışında *Nmap* gibi bir bilgisayardaki aktif olan portları bulan ve *Mscan*, *Saint* ve *Satan* gibi bilinen açıkları taramaya yarayan türleri de vardır.

Port tarama, ağ seviyesi saldırıların başında gelmektedir ve sızma testlerinde ilk adımlardan biridir (bilgi toplama aşaması). Sızma testleri, sistemin açıklıklarının kontrolü açısından yapılması yararlı olan ve açıklıkların belirlenmesine yönelik faaliyetlerdir. Port tarama yapmadan diğer aşamalara geçilmesi genellikle sızma testlerinin eksik kalmasıyla sonuçlanmaktadır. Zira bir güvenlik zafiyetinin olması için öncelikle hizmet veren bir servisin olması gerekmektedir, hizmet veren bir servisin bulunması için de TCP/IP temelli ağlarda port kavramı devreye girmektedir.

Port tarama işlemi gürültü çıkarttığı için çok rahatlıkla STS'ler tarafından yakalanabilmektedir. Port tarama işlemlerinde hedef sisteme yakalanmadan tarama işlemini gerçekleştirmek için çeşitli yöntemler bulunmaktadır. Bunlardan biri de *Decoy Scanning* olarak literatüre geçen tuzak sistemler aracılığıyla port tarama yapmaktır. Tuzak sistemler kullanarak tarama yapma, hedef sisteme port tarama için gönderilen paketlerin eş zamanlı olarak ve farklı ip adreslerden gönderilerek hedef şaşırtma amaçlı kullanılmasıdır.

Tuzak sistemler aracılığıyla tarama yapabilmek için tarama yapan sistemin *IP spoofing* yapabilir durumda olması gerekir. Sadece sistemin *IP spoofing* yapabilmesi yetmez, önündeki router, firewall vb. gibi sistemlerin de bu spoof edilmiş paketleri geçiriyor olması gerekir. URPF, antispoof gibi korumalar olmaması gerekir.

*Decoy scanning*, TCP için üçlü el sıkışma gerektirmeyecek tarama türlerinde ve UDP, ICMP tarama tiplerinde geçerlidir. Sahte IP adreslerinden TCP üçlü el sıkışma tamamlanamayacağı için versiyon tarama gibi türlerde sahte IP kullanılamaz. Tuzak sistemler aracılığıyla port taraması *Nmap* kullanılarak yapılabilmektedir [73].

#### 3.3.1.11 Tahrip Etme

Tahrip etme, vandalizm olarak bilinen ve bir kişiye, kuruma ya da kamuya ait bir mala, araca ya da ürüne zarar vermeye yönelik olarak gerçekleştirilen eylemdir. Bilgi güvenliği sistemleri açısından web sayfa yapısının değiştirilmesi, dosyaların silinmesi, hard diskin formatlanması şeklinde veya sistemin fiziksel zarara uğratılması şeklindeki saldırılar bu kategoriye girmektedir.

#### 3.3.1.12 Özel Veri Hırsızlığı

Kişisel PC, laptop, hard disk, tablet vb. özel aygıtlardaki verilerin kopyalanmasına yönelik girişimlerdir. Gelişen teknoloji ile birlikte artan cihaz sayısına paralel olarak özel veri hırsızlığının da arttığı tespit edilmiştir. Bu sebeple bu konuda da gereken hassasiyet gösterilmelidir.

#### 3.3.1.13 Dolandırıcılık ve Suistimal

Teknolojik gelişmelere paralel olarak hırsızlık veya dolandırıcılık olayları da farklı boyutlar kazanmıştır. Dolandırıcılık, kritik dosyaların değiştirilmesi, kredi kartı numara ve şifre bilgilerinin çalınması, banka hesap bilgilerinin çalınması vb. yöntemlerle haksız maddi kazanç elde etmeye yönelik eylemlerdir.

#### 3.3.1.14 Doğrudan Erişim Saldırıları

Dolaylı saldırıların aksine bu saldırı türünde saldırganın sistem üzerinde fiziksel olarak erişim hakkı söz konusudur. Bilgisayar sistemine doğrudan erişim yetkisi olan kişilerin yaptığı saldırılar bu kategoride değerlendirilir. Bu erişim hakkını elinde bulunduran saldırgan, ilgili sistem üzerindeki işletim sisteminde kendisi için bir kullanıcı belirlemek gibi daha sonra kullanabileceği birtakım değişiklikler yapabilir. Sisteme solucan, keylogger ve gizli dinleme cihazları kurabilir. Ayrıca doğrudan erişim hakkına sahip olan saldırgan, CD-

ROM, DVD-ROM, HDD vb. yedekleme birimlerini, bellek kartları, kameralar, ses sistemleri, cep telefonu, bluetooth/kablosuz erişimi olan cihazları kullanarak, çeşitli bilgileri kendi tarafına kopyalayabilir.

#### 3.3.1.15 Kütük Dosyalarının Silinmesi / Değiştirilmesi

Kütük (Log) dosyaları sistem, kernel, servisler ve uygulamalar hakkında bilgiler içeren metin tabanlı dosyalardır. Farklı bilgiler için farklı kütük dosyaları mevcuttur. Kütük dosyaları sistem üzerindeki erişimlerin ve isteklerin zaman, erişimci bilgisi vb. bilgilerini kayıt altına alır. Kütük dosyaları sistemde olabilecek problemleri giderebilmek için birçok kolaylık sağlar. Örneğin, sisteme izinsiz giriş yapmaya çalışanların tespitine veya işletim sisteminin yüklenmesi sırasında oluşan bir sorunun tespitine olanak sağlar.

Sistem kütük bilgilerini silmek ya da değiştirmek, olası bir suç durumunda delillerin ortadan kaldırılmasını amaçlayan yöntemlerdir.

#### 3.3.1.16 Güvenlik Mimarisinin Değiştirilmesi

Bilgi güvenliğinin sağlanabilmesi açısından çeşitli standartlar ortaya konulmuştur. Bu konuda uluslararası bir standart olan ISO/IEC 27001, Bilgi Güvenliği Yönetim Sistemi (BGYS) kurmayı planlayan kurum ya da kuruluşlara Bilgi Güvenliği Politikası'nın yazılması konusunda yol gösteren bilgiler içermekte ve bu politikada bulunması gerekli olan unsurları örneklerle açıklamaktadır. UEKAE BGYS-0005 ise ülkemizde kullanılan bilgi güvenliği politikası oluşturma kılavuzudur.

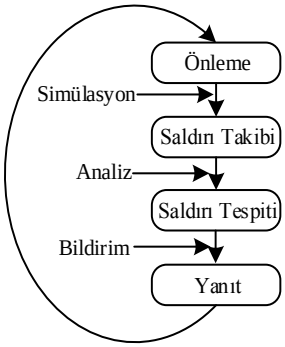
Bir sistem üzerinde güvenlik politikalarının değiştirilmesine yönelik olarak gerçekleştirilen her eylem bir saldırı olarak nitelenebilir. Örneğin güvenlik duvarı ya da sunucu konfigürasyonlarını değiştirmeye yönelik olarak gerçekleştirilen saldırılar bu kategoride değerlendirilebilir.

### 3.3.2 Saldırı Tespit Sistemlerinin Yapısı ve Sınıflandırılması

Saldırı tespiti, tarihi oldukça eski bir olgudur. Eski zamanlarda şehirleri korumak için tepelere kaleler ve bu kalelerin içlerine ise yüksek ve sağlam kuleler yapılırdı. Bu yapılar sembolik bir güç göstergesi olmalarının yanında şehir savunmasında açık bir görüş sağlaması, olası saldırıların erken fark edilmesi ve saldırı durumlarında zaman kazandırma gibi birçok açıdan önemli olagelmiştir.

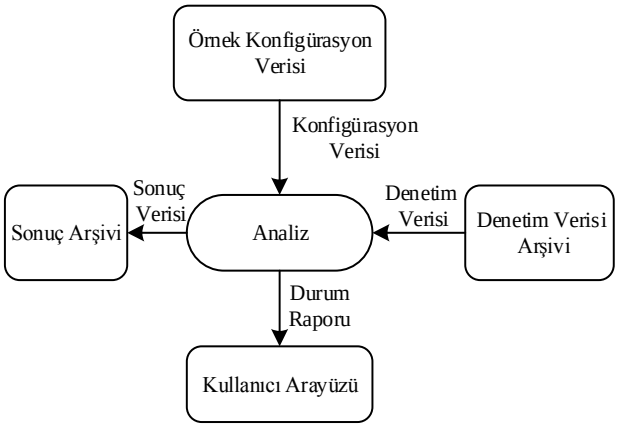
Günümüz dünyasında, bilgisayar teknolojisinin gelişmesi ile saldırılar farklı bir boyut kazanmıştır. Artık klasik fiziksel saldırı ve savunma faaliyetleri bilgi sistemleri dünyasında siber savaş aktiviteleri olarak karşımıza çıkmaktadır. Çok daha profesyonel

şekilde, mesafeden bağımsız olarak kilometrelerce uzaktan, beklenmedik bir zamanda ve fark edilmeden siber uzay üzerinden sistemlere çeşitli saldırılar yapılabilmektedir. Ancak yine bilgisayar teknolojisi sayesinde, bilgisayarlarda meydana gelen her olayın kaydı (log) tutulduğundan, saldırıların ne şekilde yapıldığı tespit edilebilmekte ve bunlara karşı önlem alınabilmektedir. Şekil 3.8, bu tespit işleminin nasıl yapıldığına dair bir fonksiyonel yapı sunmaktadır. Bilgi güvenliği sürekli olarak devam etmesi gereken kritik bir süreçtir. Bu sebeple sistemler üzerinde, saldırı takibi, saldırı analizi, yanıt/önleme işlemleri devamlı olarak yapılmalıdır.




**Şekil 3.8** Saldırı tespit sistemlerinin fonksiyonel yapısı

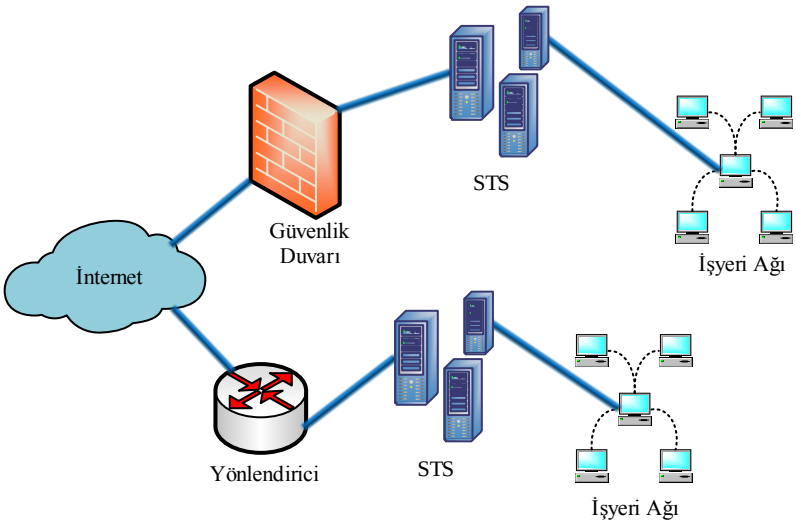
Şekil 3.9, STS'lerin temel yapısını, Şekil 3.10 ise STS'ler için ağ üzerinde genel bir yerleşim modeli sunmaktadır.




**Şekil 3.9** Saldırı tespit sistemlerinin temel yapısı

STS'ler için genel olarak, örnek konfigürasyon verisi, denetim verisi arşivi, kullanıcı arayüzü, sonuç arşivi ve analiz modüllerini içeren bir yapı önerilmektedir. Burada denetim verisi kullanıldığı varsayılmıştır. STS'nin türüne göre log kayıtları veya ağ paketleri de kullanılabilmektedir. Temel olarak arşivden alınan denetim verileri, örnek konfigürasyon verisi ile karşılaştırılarak analiz işlemi gerçekleştirilir. Bu analiz ile elde edilen sonuçlar

sonuç arşivine gönderilmekte ve aynı zamanda bir durum raporu olarak kullanıcı arayüzü ile sunulabilmektedir.




**Şekil 3.10** Saldırı tespit sistemlerinin ağlarda genel yerleşimi

STS'lerde genel olarak "Kötüye Kullanım" ve "Anormallik Tespiti" olmak üzere temelde iki tür yaklaşım bulunmaktadır. Literatürde bu iki yaklaşımı kullanan çeşitli çalışmalar bulunmakla birlikte her iki yaklaşımı birlikte kullanan sistemler üzerinde çalışmalar devam etmektedir [8, 9].

Anormallik tespiti olarak adlandırılan yaklaşım, sistemdeki kullanıcı davranışlarını analiz ederek modellerken, kötüye kullanım (imza) tespitine dayanan yaklaşım ise, saldırganların davranışlarını modeller. Bu iki yaklaşımı kullanan STS'lerin değerlendirilmesi aşamasında bazı metrikler kullanılmaktadır. Bu tez çalışmasında da kullanılan bu metrikler ve tanımları aşağıda verilmiştir.

**Doğru pozitif (True Positives):** Doğru olarak sınıflandırılan saldırılar için kullanılır.

**Doğru negatif (True Negatives):** Doğru olarak sınıflandırılan saldırı olmayan, iyi tür aktiviteler.

**Yanlış pozitif (False Positives):** Yanlışlıkla saldırı olarak sınıflandırılan, gerçekte saldırı olmayan, iyi tür aktiviteler.

**Yanlış negatif (False Negatives):** Yanlışlıkla iyi tür, saldırı olmayan aktiviteler olarak sınıflandırılan, gerçekte saldırı olan aktiviteler.

### 3.3.2.1 Anormallik Tespiti

Anormallik tespiti (anomaly detection) yaklaşımı, temel olarak sistemde meydana gelen anormal olayları, normal olaylardan ayırt etme mantığı ile çalışmaktadır. Bir bilgi güvenliği sistemi içerisinde yer alan STS için anormallik, normal aktivitelerden herhangi bir

sapma durumuna karşılık gelir. Sistemin normal davranışı, uzun süren bir analiz sonucunda elde edilebilir. Sistemdeki kullanıcı veya kullanıcı gruplarının davranış profillerinin ortaya çıkartılması anormallik tespiti için en temel iştir. Normal davranış profili belirlendikten sonra, farklılık gösteren davranışlar saldırı olarak tespit edilir. Anormallik tespitinde saldırıların doğru tespit edilmesi, normal davranış profilinin ne kadar doğru belirlendiğiyle orantılıdır [8].

Anormallik tespiti yapan sistemlere öğrenen sistemler de diyebiliriz. Çünkü sistemdeki normal faaliyetler sürekli güncellenir. Sistemdeki bütün faaliyetler sürekli izlenir ve tespit edilen normal faaliyetler ile kıyaslanır. Anormal bir faaliyet tespit edilmesi durumunda bu aktivite, bir saldırı olarak kabul edilir. Anormallik tespitinde bütün sızma faaliyetlerinin mutlaka alışılmışın dışında gerçekleşeceği varsayılır. Örneğin, sistem üzerindeki istatistiklere bakarak normal kullanım modelleri (CPU kullanımı, çalışma saatleri vb.) belirlenir ve bu normal kullanım dışındakiler saldırı kabul edilir [9]. Bu yöntemin zorluğu, normal sistem özelliklerinin belirlenmesindeki zorluklardır. Bu yöntemde yanlış veya yetersiz modelleme nedeniyle normal işlemler yanlışlıkla saldırı olarak kabul edilebilir. Bu yöntemin avantajı ise yeni saldırıların tespit edilebilmesidir. Anormal davranışların belirlenmesinde genellikle aşağıda belirtilen profiller kullanılır [9].

**Bireysel Profiller:** Bir kullanıcının yapması beklenen genel faaliyetleri kapsar. Bu faaliyetler, çalışma saatleri ve çalışma biçimi gibi bireysel aktivite örüntüleridir.

**Grup Profiller:** Bir gruptaki kullanıcıların profilidir. Kullanıcıların çalışma biçimi, kullandıkları kaynaklar, tarihsel aktiviteleri kapsar.

**Kaynak Profili:** Uygulamalar, kullanıcı hesapları, iletişim portları gibi kaynakların nasıl kullanıldığı gözlemlenir.

**Diğer Profiller:** Çalıştırılabilir programların sistem kaynaklarını nasıl kullandığı gözlemlenir.

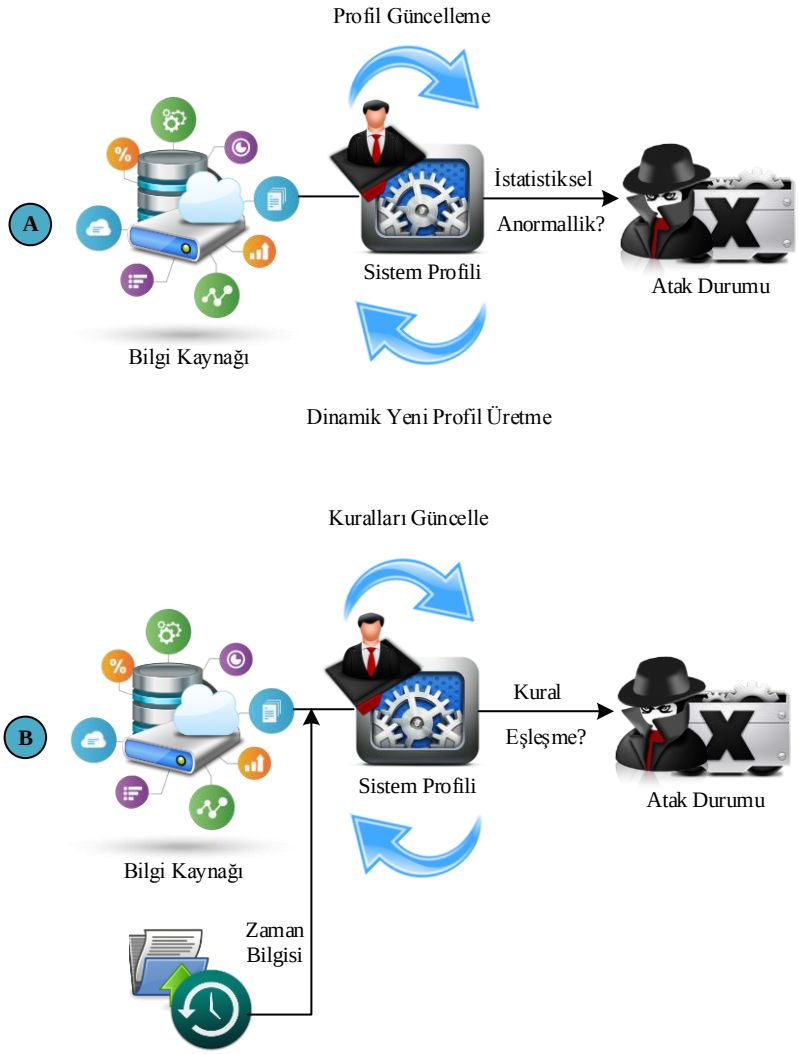
Anormallik tespiti yaklaşımı için üç temel teknikten söz edilebilir. Bunlar ***istatistiksel anormallik tespiti***, ***veri madenciliği*** yöntemlerine bağlı teknikler, ***yapay zekâ*** yöntemlerine bağlı teknikler olarak verilebilir [9].

**İstatistiksel Anormallik Tespiti:** Bu yöntemde her kullanıcı için iki tane profil tutulur. Bir tanesi sistemde saklanan ve her kullanıcının izlenmesi sonucunda elde edilen profil, diğeri ise aynı kullanıcının anlık profilidir. Gelen paketlere, bilgisayardaki log kayıtlarına vb. bakılarak anlık profil belirli aralıklarla güncellenir ve sistemde saklanan profil ile karşılaştırılır. Eğer fark önceden belirlenen eşik değerinden fazla ise sistem bunu saldırı

kabul eder ve alarm verir. Bu yöntemin avantajı, saldırılar hakkında önceden bilgisi olmasına gerek yoktur ve yeni saldırıları tanıyabilir. Dezavantajı ise, yetenekli saldırganlar tarafından sistemin anormal bir davranışını normal olarak tanıyacak şekilde eğitilebilmesidir.

**Veri Madenciliği:** Bu teknikte veri girdi olarak alınır ve çıplak gözle kolaylıkla görülemeyecek bağıntılar ortaya çıkarılmaya çalışılır. Örneğin, veri madenciliği ile doğru ağ trafiğinin sınırlarının tanımlanması, analizcinin saldırıyı normal ağ trafiğinden ayırt etmesine yardımcı olur.

**Yapay Zekâ:** Yapay zekâ teknikleri ile öğrenme, bir sistemin belirli bir olayı öğrenmesi ve zaman içinde kendini güncelleyebilmesi yeteneği olarak tanımlanır. Diğer bir ifadeyle sistem yeni olaylara karşı verdiği tepkiyi değiştirebilmektedir. Şekil 3.11, kullanılan yöntem açısından STS türlerini göstermektedir.




**Şekil 3.11** STS'lerde farklı yaklaşımlar a) Anormallik tespiti b) Kötüye kullanım tespiti

Anormallik tespiti yaklaşımının kötüye kullanım tespiti yaklaşımına göre avantajı, önceden bilinmeyen saldırıların da tespit edilebilmesidir. Dezavantajı ise yanlış alarm (false positive) yani gerçekte saldırı olmayan davranışların da saldırı olarak belirlenmesi oranının yüksek olmasıdır. Anormallik tespitinde istatistiksel yöntemler, yapay zekâ teknikleri, yapay sinir ağları, veri madenciliği, yapay bağışıklık sistemi gibi birçok farklı teknik kullanılabilir [8, 9].

#### 3.3.2.2 Kötüye Kullanım Tespiti

Kötüye kullanım tespiti (misuse detection) yaklaşımında, saldırganların davranış modellerinin çıkarılması esastır. Saldırının imzası olarak da nitelendirilen bu modeller, daha önce karşılaşılmış saldırıların analiz edilerek kendine has karakteristik özelliklerinin çıkarılması ile elde edilir. Saldırılara yönelik imza veri tabanları oluşturulduktan sonra, bu imzalarla eşleşen hareketler saldırı olarak tespit edilir.

Kötüye kullanım tespiti yaklaşımının anormallik tespitine göre avantajı, imzası bilinen her saldırının tespit edilebilmesi ve yanlış pozitif üretmemesidir. Dezavantajı ise, imzası bilinmeyen saldırıların tespit edilememesi ve bu nedenle yanlış negatif (saldırı olan bir davranışın saldırı olarak tespit edilmemesi) oranının yüksek olabilmesidir. Yanlış negatif oranının azaltılması için imza veri tabanının yeni saldırı imzaları ile güncellenebiliyor olması gerekir [8].

Kötüye kullanım yönteminde, bilinen saldırı örüntüleri tanımlanır ve bu tanımlar ile mevcut aktiviteler karşılaştırılarak saldırı olup olmadığına karar verilir. Örneğin, "şifre tahmin etme" saldırısı için; "eğer 2 dakika içerisinde 4 adet başarısız giriş denemesi olursa, bu bir saldırıdır" şeklinde kural tanımlanabilir [9]. Bu yöntemin en büyük avantajı, bilinen saldırı modellerinin sisteme tanıtılabilmesidir. Böylece bilinen saldırılar tam ve doğru olarak tespit edilebilmektedir. Dezavantajı ise sisteme tanıtılmamış olan yeni saldırıları tanıyamamasıdır.

Kötüye kullanım sistemindeki temel mantık, saldırıların belirli bir örüntü şeklinde tanımlanmasıdır. Anormallik tespiti yönteminde, bilinmeyen kötü davranışlar tespit edilmeye çalışılırken, kötüye kullanım yönteminde bilinen kötü davranışlar tanımlanmaya çalışılır. Kötüye kullanım tespitinde yararlanılan yöntemler 4 gruba ayrılabilir [9] ;

**Uzman Sistemler:** Kodlama bilgisi olan uzmanlar tarafından eğer-ise (if-then) şeklinde kurallar tanımlanarak saldırılar tespit edilmeye çalışılır.

**Model Tabanlı Muhakeme Sistemleri:** Kötüye kullanım olduğunda kötüye kullanım modellerini beraber kullanarak doğru sonuç elde edilmeye çalışılır.

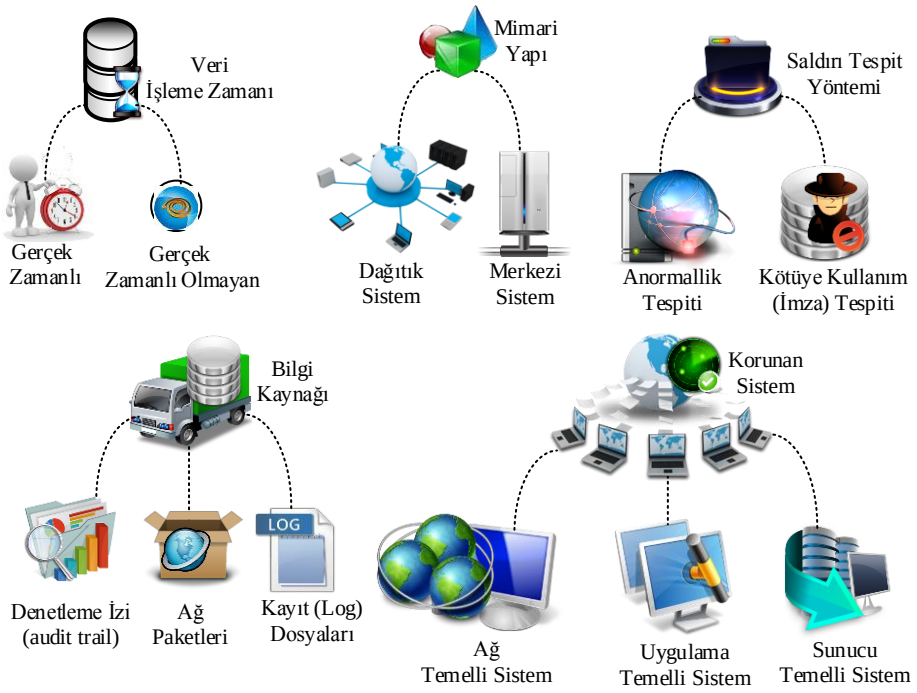
**Durum Geçiş Analizi:** Saldırıları, izlenen sistemdeki bir dizi durum geçişi olarak modeller. **Klavye Kullanım Analizi:** Kullanıcının klavye kullanma hızı kontrol edilerek, başka bir kullanıcının bilgisayarı kullanması durumundaki saldırı tespit edilmeye çalışılır. Bu yöntem sadece klavye kullanımını test ettiğinden kötü niyetli yazılımlar ile yapılan saldırıları tespit edemez. STS'ler çok çeşitli seviyede veri toplayıp bunlar üzerinde işlem yapabilirler. Genel olarak toplanan veriler şunlardır [75];

- Ağ verisi (router ya da firewall gibi cihazlardan sağlanan veri),
- İşletim sistemi komut satırı,
- İşletim sistemi sistem çağrıları,
- Uygulama içi bilgiler,
- İletilen tüm karakterler,
- Tuş basmaları.

#### 3.3.2.3 Saldırı Tespit Sistemlerinin Sınıflandırılması

STS'ler birçok farklı özelliğe göre sınıflandırmaya tabi tutulabilirler. Genel olarak, *veri işleme zamanı*, *mimari yapıları*, *saldırı tespiti yaparken kullandıkları yöntemler*, *bilgi kaynağı* ve *korunan sistem* sınıflandırmada temel alınan bazı özelliklerdir. STS'ler günümüze kadar farklı birçok kritere göre sınıflandırılmıştır. Bunlardan en çok bilinen sınıflandırma türü saldırı tespit yöntemine göre olup, anormallik tespiti ve kötüye kullanım tespiti olarak ikiye ayrıldığı yukarıda verilmişti. STS'ler için genel bir sınıflandırma Şekil 3.12'de verilmiştir. Burada en çok kullanılan sınıflandırma yöntemlerine göre bir kategorizasyon yapılmıştır.

Bir saldırı tespit sistemi, Şekil 3.12'de verilen her bir kategoride farklı bir özelliği ile yer alabilir [9]. Bu durum, buradaki tasnifin yapıldığı özelliğe bağlıdır. Örneğin; 1988'de geliştirilen IDES adlı STS, veri işleme zamanına göre gerçek zamanlı, mimari yapısına göre merkezi sistemli, kullandığı bilgi kaynağına göre sunucu tabanlı, kullandığı saldırı tespit yöntemine göre anormallik tespiti yaklaşımı, koruduğu sisteme göre ise sunucu temelli STS sınıfına dahildir. Eğer belirleyici olarak kullanılan sınıflandırma özniteliği sadece saldırı tespitinde kullanılan yaklaşım ise IDES, anormallik tespiti sınıfında yer alır. Bu durumda diğer kriterlerden bahsedilmez. Ama tüm bu özellikler, bir STS'nin karakteristik yapısını belirlemesi açısından önemlidir [9].




**Şekil 3.12** Saldırı tespit sistemlerinin sınıflandırılması

### Veri İşleme Zamanına Göre STS:

STS'ler için veri işleme zamanı, sınıflandırmada kullanılan kriterlerden biri olup izlenen olaylar ve olayların değerlendirilmesi arasında geçen zamanı ifade eder. Bu sınıflandırma türüne göre STS'ler, "gerçek zamanlı" ve "gerçek zamanlı olmayan" şeklinde ikiye ayrılmaktadır.

*Gerçek zamanlı STS yapılarında*; veri, iletişimin sağlandığı gerçek zaman diliminde analiz edilir ve eğer bir saldırı tespit edilirse saldırıya cevap olarak gereken işlemler gerçekleştirilir. Bilgi akışının yoğun olduğu büyük ağlarda/sistemlerde uygulanması zor ve maliyeti daha yüksektir. Ancak aktif olarak gerçek zamanlı cevap üretilmesi gereken kritik öneme sahip sistemlerde kullanılması zorunlu olan yapılardır. Piyasada hali hazırda bulunan birçok ticari STS uygulaması bu yapıdadır.

*Gerçek zamanlı olmayan sistemlerde*; alınan veriler depolanarak, analiz edilmesi için ilgili STS'ye gönderilir. Bu yapıda anlık, gerçek zamanlı bir saldırı tespitinin yapılması söz konusu değildir. Çünkü saldırı analizinin yapıldığı zaman dilimi ile olası saldırının gerçekleştirildiği zaman dilimi farklıdır. Bu sistemlerde belirlenen tehditler ve olası saldırılar, sistem kullanıcısını uyaracak biçimde alarmlar üretir. Saldırının profilinin çıkarılması ve bu tür saldırılara bir daha maruz kalınmaması için, geçmişe yönelik depolanan

veriler üzerinde yapılan analizlerle sonuçlar veya saldırılar elde edilir. Bu yapıdaki bir STS, sistemin belirlenen güvenlik açıklarını kapatmak için kullanılır [9].

**Mimari Yapılarına Göre STS:**

STS'lerin mimari yapısı, ağ üzerinde sistemde aktif olarak rol alacak olan fonksiyonel bileşenlerin birbirlerine göre nasıl konumlandırıldıkları ile alakalı bir durumdur. Bir bilgi güvenliği ya da bilgisayar ağ sisteminde temel fonksiyonel bileşenler; izlenen sistem, analizin yapıldığı sunucu ile çevresi, ağ cihazları ve problemler için izlenen hedef olarak verilebilir.

STS'ler mimari yapı açısından, "merkezi sistem" ve "dağıtık sistem" olmak üzere ikiye ayrılmaktadır. Merkezi sistem mimari yapısında; tüm izleme, tespit ve raporlama işlemleri bir merkezde kontrol edilir. Bilinen STS'lerin büyük bir kısmı bu kategoriye girer. Merkezi kontrol için fiziksel yakınlık önemli bir etken değildir.

Dağıtık sistemlerde ise tüm saldırı analizi ve ağ izleme işlemleri, analiz noktasında yapılan ajan (agent) tabanlı yaklaşım ile yapılır. Dağıtık STS yapısı genellikle çok geniş olan ağlarda kullanılır. Mesela, bir kurumun farklı şehirlerde yer alan bilgisayar ağ sistemlerinin tümünü kapsamına alan STS'ler bu şekildedir [9].

**Bilgi Kaynağına Göre STS:**

STS'ler, saldırıların tespitine yönelik olarak bilgi kaynaklarını analiz ve karşılaştırma yapmak için kullanırlar. Saldırı tespiti için temelde kullanılabilecek bilgi kaynakları, bilgisayar veya ağ paketlerinin dinlenmesinden elde edilebildiği gibi, kullanıcıların davranış örüntülerinden de elde edilebilir. STS açısından kullanılabilecek bilgi kaynakları "denetleme izi", "ağ paketleri" ve "uygulama kayıt (log)" dosyalarıdır.

Denetim (hesap, günlük) izi; bilgi ve iletişim güvenliği için, sistem aktivitelerinin zamana göre sıralanmış şeklidir. Bu yapı, sistem üzerinde gerçekleşen olayların sıralaması bozulduğunda veya bu olaylarda değişiklikler meydana geldiğinde, sistemin yeniden yapılandırılmasında ve buradaki test işlemlerinin gerçekleştirilmesinde kullanılır. STS'ler denetleme izini, daha çok sistemde tanımlı bulunan kullanıcıların veya grupların hareket-davranış örüntülerini çıkarmak için kullanır. Günlük yapılan işler ve bu işlere yönelik olarak sistemdeki yetkiler göz önünde bulundurularak kullanıcı profilleri oluşturulur. Çeşitli analizlerle bu profillerin doğrulukları ispatlanır. STS'ler bu belirlenen profillere aykırı her türlü hareketi saldırı olarak algılar [8].

STS'ler açısından bir başka bilgi kaynağı yapısı olan ağ paketleri; koklayıcılar (sniffers) tarafından ağ trafiğinin dinlenmesiyle elde edilir. Bu günlükler daha çok hizmet aksattırma (DoS) saldırılarını tespit etmekte kullanılır. Ağ paketlerinin dinlenmesiyle elde edilen bilgiler sayesinde, sunucu tabanlı STS'lerden farklı olarak, ağ katmanında gerçekleşen saldırı olaylarını da tespit etmek mümkün olabilmektedir.

Uygulama log dosyaları; uygulama katmanında gerçekleşebilecek saldırıları tespit etmekte kullanılabilir. Bu veri kaynağı, diğer iki kaynaktan daha kolay elde edilebilmektedir ancak saldırı tespit oranı daha düşüktür. Ayrıca gerçek zamanlı bir saldırı tespiti yapılması gerekiyorsa, eş zamanlı olarak uygulama log dosyalarının hem oluşturulup hem analiz edilmesi gibi bir yapı pratikte pek mümkün olamayacağından bu bilgi kaynağı kullanılmaz. Daha çok saldırı ile analiz aşamasının farklı olduğu, gerçek zamanlı olmayan sistemlerde log kayıtlarının analizleri yapılır. Bilgi kaynağı olarak uygulama kayıt dosyalarını kullanan bir STS, web tabanlı saldırıları tespit etmek için geliştirilen bir yazılım olan WebWatcher'ın uygulama log dosyalarını kullanabilir [8].

**Saldırı Tespit Yöntemine Göre STS:**

STS'lerde, saldırı tespit yöntemi olarak anormallik tespiti ve kötüye kullanım tespiti olmak üzere iki farklı yaklaşım kullanılmaktadır. Bu yaklaşımlar, 3.8.1 ve 3.8.2 bölümlerinde detaylı bir şekilde anlatılmaktadır.

**Korunan Sisteme Göre STS:**

STS'ler, korudukları sistemin yapısına göre de sınıflandırılabilmektedir. Bu sınıflandırma yöntemine göre STS'ler üç grupta toplanabilir. Bunlar ağ, sunucu ve uygulama temelli STS yapılarıdır. STS'ler koruma hacmi anlamında küçük ya da büyük bir alanı kontrol edebilirler. Küçük bir alanı kontrol eden STS'ler genel olarak "sunucu temelli", büyük bir alanı kontrol eden STS'ler ise "ağ temelli" olarak bilinirler.

*Ağ temelli STS'ler*, ağ trafiğini izleyerek sisteme karşı olabilecek saldırıları tespit etmek amacıyla kullanılmaktadır. Bu sistemler, ağ paketlerini analiz ederek saldırıları tespit ederler. Ticari STS'ler genellikle ağ temellidir. Ağ temelli STS yapıları, güvenlik duvarları ile karıştırılmamalıdır. Güvenlik duvarları sadece belirli kural tanımlamalarına göre çeşitli servis/sistem erişimlerini denetler. Paketlerin içeriklerini herhangi bir kontrole tabi tutmaz. Ancak STS'ler ağ paketlerinin içeriğini kontrol eder. Bu yapısıyla ağ temelli STS'ler özellikle ağ iletişimindeki zafiyetlerin kullanılmasıyla gerçekleştirilen SYN Flood ve TCP port taraması gibi saldırıların tespitinde kullanılmaktadır [8, 9].

*Sunucu temelli STS'ler*, bir bilgisayar üzerindeki saldırıların tespitine yönelik olarak çalışır. İlgili bilgisayar üzerindeki sistem ve olay günlükleri, işletim sistemi hesap izleri gibi verilerin toplanıp analiz edilmesi ile kötücül davranışlar ortaya çıkarılabilir. Eklenen yeni bir kayıtta yapılan işlemle ilgili olarak "saldırı" veya "normal aktivite" şeklinde bir karara varılır. Bu kararın alınması aşamasında STS yapısındaki imza veri tabanı kullanılır. Sunucu temelli STS'ler, bir sunucu bilgisayar üzerinde çalıştığı için üzerinde çalıştığı ağ yapısının tamamını görüp analiz edemez. Ancak ağ temelli sistemlerden hızlıdır ve bunların göremediği bazı saldırıları tespit edebilmektedir. R2L ve U2R saldırıları buna örnek olarak verilebilir.

*Uygulama temelli STS'ler* ise, bazı kaynaklarda sunucu temelli STS'lerin bir alt kümesi ve daha küçük bir versiyonu olarak değerlendirilmektedir [8]. Bir sunucu bilgisayar üzerinde bir uygulama yazılımı şeklinde yer alırlar. Olayları bu uygulama yazılımı ile analiz eden uygulama temelli STS'ler, bilgi kaynağı olarak ilgili uygulamanın oluşturduğu günlük-log kayıtlarını kullanırlar.

### 3.3.3 Saldırı Tespit Sistemlerinde Kullanılan Yöntemler

Saldırıları tespit etmek amacıyla çok farklı yöntemler kullanılmaktadır. Literatürde günümüze kadar gerçekleştirilmiş uygulamalarda kulanılan yöntemlerden bazıları aşağıda sunulmuştur.

**Metin madenciliği (Text Mining):** Metin madenciliği, metinlerin yarı otomatik veya otomatik olarak işlenmesi süreçlerini içeren bir yöntemdir. Kelimelerle ilgilenen metin madenciliği, metinlere bağlı olarak yapının belirlenmesi, incelenmesi ve metinden bilgi çıkarılması işlemlerini içerir. Metin madenciliği çalışmalarında, log dosyalarına taşınarak saklanan çok büyük boyutlarda milyonlarca kelimenin ham veri olarak ele alınıp çeşitli tekniklerle analiz edilmesi ve sonucunda da çeşitli ilginç ve gerekli bilgiler elde edilmesi amaçlanmaktadır. Örneğin saldırı tespiti yaklaşımı olarak metin madenciliği yöntemlerinden olan metin sınıflandırma ile web uygulamalarına yapılan yetkisiz erişimler tespit edilebilmektedir [9].

**Ensemble yöntemi (Ensemble Method):** Boosting Algoritmaları, toplama yöntemi (ensemble method) adı verilen ve literatürde makine öğrenmesi konuları arasında değerlendirilen algoritmalardır. Toplama yöntemi algoritmalarının temeli, güçsüz birçok öğrenme algoritmasının birleştirilerek daha güçlü bir öğrenme algoritmasının

oluşturulmasına dayanmaktadır. Burada güçsüz algoritma olarak genellikle karar ağaçları seçilir. Bunun sebebi ise kolay oluşturulabilmesi ve hızlı türetilebilmesidir. Boosting Algoritması, diğer güçsüz sınıflandırıcılar ile çalışan Ensemble algoritmalarına benzemektedir. Burada da güçsüz çok sayıda sınıflandırma yöntemi kullanılarak güçlü bir sınıflandırıcı yapısı oluşturulur. Ancak burada güçlü sınıflandırıcı oluşturulmaya çalışılırken, modelin daha önce yanlış sınıflandırdığı örnekleri daha iyi sınıflandırması için her güçsüz sınıflandırıcıya bir ağırlık atanmaktadır [76]. Chebrolu ve arkadaşları STS'lerde önemli özelliklerin seçilerek doğru bir sınıflandırıcı yapılabilmesi için Ensemble yöntemlerinden Naive Bayes ve CART yöntemlerini birlikte kullanarak hibrit sistem geliştirmişlerdir [9, 149].

**Bağışık Sistemler (Immune Systems):** Yapay bağışık sistemler, son yıllarda popüler alanlardan biri olmuştur. Özellikle saldırı tespit sistemleri, robotik, kontrol, optimizasyon, anormallik tespiti gibi bir çok alanda başarılı sonuçlara ulaşılmasını sağlamıştır. Yapay bağışık sistemler, canlılardaki bağışıklık sistemi yapısının matematiksel bir modellemeyle çeşitli alanlardaki problemlerde kullanılması temeline dayanan bir yapay zekâ yöntemidir. Aickelin and Greensmith, anormallik tespiti için bağışıklık sistemlerinde kullanılan DCA ve TLR Algoritmalarının kullanılabileceğini belirlemişlerdir [9, 150].

**Veri Madenciliği (Data Mining):** Veri tabanındaki büyük boyutlu verilerden ilginç-saklı örüntüleri ortaya çıkarmak için yapılan analizlerdir. Örüntüleri ve veriler arasındaki ilişkileri bularak yeni kurallar çıkarmak için kullanılır. Veri madenciliği yöntemleriyle hesap izleri, günlük kayıtları vb. birçok materyal kullanılabilir. Normal aktivitelerden sapma gösterenler tespit edilebilir [8].

**Kural Tabanlı (Rule Based) Sistemler:** Bu yapıdaki STS'ler, ağ trafiğini inceleyip çeşitli kurallar oluşturur ve saldırı analizi aşamasında bu kural veri tabanında belirlenmiş olan kurallara göre aktiviteleri değerlendirirler.

**Açıklayıcı İstatistikler (Descriptive Statistics):** Bu tip STS'ler, kullanıcı veya sistemin davranışsal örüntüsünü izleyerek bir model oluştururlar. Çeşitli değişkenler kullanılarak oluşturulan bu istatistiki model, sistem üzerindeki davranışların sürekli olarak ölçülmesiyle elde edilir. Bu yapıdaki değişkenlerden bazıları; kullanılan disk alanı, oturum girişi, oturum kapatma, belirli bir zaman aralığında erişilen dosya sayısı, kullanılan hafıza alanı, kullanılan

işlemci zamanı ve belirli bir zaman aralığındaki ağ bağlantı sayısı olabilir. Buradaki bu değişkenlerin basit istatistiklerle kullanıcı profillerini oluşturmasıyla normal profillerin dışında kalan aktiviteler, saldırı olarak tespit edilebilmektedir. İstatistiksel yöntem kullanımının dezavantajlarından biri, saldırganın sistem üzerindeki bu istatistikleri öğrenerek ona göre davranış sergileyebilmesidir [8].

**Eşik Değeri Tespiti (Threshold Value):** Bu yöntemde model oluşturulurken belirli olayların tekrarlama sıklığı / sayısı ve belirli zaman aralıkları dikkate alınmaktadır. Eşik değer tespitinde çeşitli özellikler baz alınır. Örneğin yanlış giriş sayısı, giriş/çıkış sayısı, hata sayısı, sistem kaynaklarına erişim ve bunları kullanma sayısı gibi konular eşik değerler ile tespit edilebilir. Bu yöntemde, belirlenen eşik değerlerin aşılması durumları ise saldırı olarak değerlendirilmektedir. Eşik değer yaklaşımı tek başına yeterli değildir ancak büyük STS yapılarında alt bileşen olarak kullanılabilmektedir [8].

**Durum Geçiş Analizi (State Transition Analysis):** Bu yöntem, bir işin gerçekleştirilmesi için birbirini izleyen durumların bir sırası olduğu varsayımına dayanır. Bu sıraya göre bir durum değişim serisi oluşturulur. Böylelikle sızma olarak nitelendirilebilecek olayların senaryosu çıkarıldıktan sonra anahtar hareketler, imza hareketler olarak tanımlanır. Bir saldırının tamamlanması için gerekli olan en küçük hareket kümesi olarak tanımlanan imza hareketler, bu yapının temelini oluşturur.

**Uzman Sistemler (Expert System):** Uzman sistemler, belirli bir alanda o alanın tamamına hâkim, ilgili bilgilerle donatılmış ve ilgili alana yönelik problemlere o alandaki uzman bir kişinin getirdiği gibi çözümler getirebilen, olabilecek yeni durumlar karşısında yeni ve mantıklı çıkarımlar yapabilecek sistemler olarak tanımlanabilir. Bilgi güvenliği alanında STS'lerle ilgili ilk sistemler kural-tabanlı uzman sistemler olarak ortaya çıkmıştır [8, 151].

**Örüntü Eşleştirme (Pattern Matching):** Bu yöntemde, sistem tarafından karşılaşılmaması gereken olay/durum daha önceden tanımlanmıştır. Burada sistemin yapısına göre kötücül olarak değerlendirilebilecek herhangi bir aktivite tanımı sistemde yer aldığından o olay gerçekleştirildiğinde saldırı uyarısı verilebilmektedir. Örneğin "sistem dosyasını sil" komutu görüldüğünde basit bir şekilde bunun saldırı olduğu tespit edilmektedir [8, 151]. Tablo 3.2'de STS'lerde kullanılan tekniklerin temel özelliklerine göre karşılaştırılması verilmektedir [8, 9].

**Tablo 3.2** STS'lerde kullanılan tekniklerin karşılaştırılması

| Teknik | Tespit Yaklaşımı | Bilgi Kaynağı | Bilinen Saldırı | Bilinmeyen Saldırı | Başarım |
|---|---|---|---|---|---|
| Metin Madenciliği | Anormallik | Denetim verisi, Bilgi tabanı, Log kayıtları | Evet | Evet | Orta |
| Ensemble | Anormallik | STS Bilgi Tabanı | Evet | Evet | Orta |
| Bağışık Sistemler | Anormallik | Denetim Verisi, Bilgi Tabanı | Evet | Evet | Yüksek |
| Veri Madenciliği | Anormallik | Denetim Verisi, Bilgi Tabanı | Evet | Evet | Orta |
| İstatistiksel | Anormallik | Denetim Verisi, Kullanıcı Profili | Evet | Evet | Orta |
| Durum Geçiş Analizi | Kötüye Kullanım | Denetim Kayıtları, Bilinen Saldırı Durum Geçiş Diyagramları | Evet | Hayır | Yüksek |
| Örüntü Eşleştirme | Kötüye Kullanım | Denetim Kayıtları, Saldırı İmzaları | Evet | Hayır | Yüksek |
| Dosya Kontrol | Anormallik | Sistem Dosyaları | Evet | Evet | Yüksek |
| Uzman Sistemler / Kural Tabanlı | Kötüye Kullanım | Kural Bilgi Tabanı | Evet | Hayır | Yüksek |
| Protokol Analizi | Anormallik | Denetim Kayıtları, Protokolün Kullanım Modeli | Evet | Evet | Düşük |
| Tuş Vuruşu İzleme | Kötüye Kullanım | Bilinen Saldırıların Bilgi Tabanı, Tuş Vuruşları | Evet | Hayır | Yüksek |

## 3.4 Saldırı Engelleme Sistemleri

Saldırı engelleme sistemleri, bilişim sistemlerine yönelik gerçekleştirilebilecek kötü niyetli saldırıların, henüz gerçekleştirilirken tespit edilip önlenebilmesi için gerçekleştirilen yazılımsal veya donanımsal araçlardır.

Teknolojik gelişmelere paralel olarak, bilişim sistemlerine yönelik geliştirilen saldırılar da çeşitlenerek artmaktadır. Dünya üzerinde bilinen ve daha önceden kaydedilmiş saldırı tipleri saldırı veri tabanlarında toplanmaktadır. Kullandığımız STS'ler, bu veri tabanlarını sürekli olarak güncel tutar ve kurumsal ya da kişisel bilgisayar sistemlerimize gelebilecek olası saldırıları sürekli izleyebilmemizi sağlar. STS'ler sadece analiz ve izleme sistemleridir. Herhangi bir saldırı engelleme özelliği içermemektedir.

STS sistemleri ile aynı teknoloji ile çalışan SES sistemlerinde ek olarak saldırı engelleme/önleme özelliği de bulunmaktadır. Bu özellik sayesinde ek bir cihaza gerek duyulmadan saldırı engelleme yapılabilmektedir. Genel olarak STS-SES çözümleri güvenlik duvarı adı verilen donanımsal/yazılımsal sistemler ile birlikte kullanılmaktadır. Bu sayede herhangi bir performans sorunu yaşamadan STS'ler ihtiyaç duyulan trafiği izleyip analiz

ederler. Analiz edilen ağ trafiğinin aşırı miktarda artması durumunda STS'ler aşırı kaynak tüketir ve kötü sonuçlar verebilirler. Bu sebeple STS yazılım/donanımları da sürekli olarak takip edilmelidir.

STS-SES yapılarını içeren sistemlerde bu hizmeti veren cihaz ya da yazılımlar küçük sayılabilecek zaman aralıklarında veri tabanlarını güncelleyerek çalışmaktadır. Dolayısıyla sistemlere gelebilecek saldırılara karşı en güvenli şekilde engelleme ve koruma sağlamış olurlar. STS hizmeti kullanmayan, üzerinde sadece 1 adet web sitesi çalışan normal bir sunucu günde ortalama oran olarak 1200-1500 arasında saldırı almaktadır. Bu saldırılar, planlı veya plansız olarak dünya üzerindeki sistemlerden gelmektedir. Sunucu üzerinde daha fazla web sitesi çalışıyor ise bu saldırı oranları katlanarak artmaktadır. Barındırma hizmeti sunan büyük sistemler için günlük saldırı alarm sayısı yüz milyonlar ile ifade edilmektedir [77].

### 3.4.1 Saldırı Engelleme Sistemlerinin Özellikleri ve Türleri

Saldırı tespit ve engelleme sistemleri, gerçekleştirmeleri gereken işlevler itibariyle oldukça önemli yapılardır. Bu sistemler genel olarak aşağıda maddeler halinde verilmiş özellikleri içerirler [78].

- İmza tabanlı çalışma yapısına ek olarak davranış analizi ve protokol anormallik tespiti yöntemleri desteği bulunmaktadır.
- Durum korumalı (stateful) izleme yeteneği vardır.
- IP ağları üzerinde saldırı tespit ya da paket izleme yeteneği vardır.
- Paket izleme yaparken değişik formlarda kayıt imkânı bulunmaktadır.
- Protokol analizi, içerik arama ve denetleme ayrıca birçok saldırı türünü tespit edebilme yeteneği vardır.
- Portscan tespit kabiliyeti bulunmaktadır.
- Gerçek zamanlı saldırı tespitini gösterebilme yeteneği vardır.
- Raporlamayı farklı uygulamalar üzerine yapabilme özelliği bulunmaktadır.
- Sürekli olarak güncellenen kural veri tabanı bulunmaktadır.
- 5000'den fazla saldırı imzası ve kuralı mevcuttur.
- Kural tabanı üzerine kuruma özel ekleme yapabilme esnekliği bulunmaktadır.
- Üçüncü parti ürünler ile entegrasyon (Örneğin; Güvenlik duvarı ile otomatik önlem alabilme) mümkündür.

- Birden fazla CPU platformu ya da işletim sistemi üzerinde çalışabilme yeteneği vardır.

Saldırı engelleme sistemleri temel olarak iki ana grupta incelenebilir. Bunlar; ağ tabanlı saldırı engelleme ve host tabanlı saldırı engelleme sistemleridir. Ağ tabanlı saldırı engelleme sistemleri, genel olarak bir ağ sistemine yönelik olarak gerçekleştirilen saldırıların tespit edilip engellenmesi temeline dayanır. Host tabanlı saldırı engelleme sistemleri ise bir sunucu ya da tekil bir bilgisayara karşı gerçekleştirilen saldırıların tespit edilip engellenmesi temeline dayanır. Bu sistemlerin bazı temel özellikleri aşağıda özetlenmiştir [78].

***Ağ Tabanlı Saldırı Engelleme Sistemleri:***

- Performans kaybı olmaksızın gerçek zamanlı saldırı engelleme özelliği bulunmaktadır.
- Hizmet aksattırma saldırısı (DoS) koruması bulunmaktadır.
- Protokol anormalliği koruması ile bilinmeyen saldırıları engelleme özelliği vardır.
- SSL ile şifrelenmiş trafiğin içerisinde saldırı tarayabilme özelliği vardır.
- İstenmeyen protokollerin durdurulması (Msn, Icq, KaZaa, eDonkey vb.) mümkündür.

***Host Tabanlı Saldırı Engelleme Sistemleri:***

- Performans kaybı olmaksızın gerçek zamanlı saldırı engelleme özelliği vardır.
- Buffer Overrun koruması ile bilinmeyen saldırıları engellenme özelliği vardır.
- Lokal saldırıları engelleme özelliği vardır.
- Registry ya da web sayfaları gibi kritik kaynakların korunması mümkündür.
- Saldırganlar tarafından sistemin ele geçirilmesini engelleme özelliği vardır.
- SQL, IIS ve Apache gibi web sunucuları için özel koruma desteği vardır.

### 3.4.2 Saldırı Engelleme Sistemlerinde Kullanılan Yöntemler

Saldırı tespit ve saldırı engelleme sistemleri, günümüzde artık genel olarak birbiriyle ilişkili bir biçimde algılanmakta ve birlikte kullanılmaktadırlar. Literatürde birçok tanım bu sistemleri IDPS (Intrusion Detection and Prevention System) olarak tanımlamaktadır. Bu sebeple bölüm 3.9'da anlatılan yöntemler SES yapılarında da geçerlidir. Saldırı engelleme sistemlerinin çoğu üç tür algılama yönteminden birini kullanır. Bunlar;

- İmza temelli algılama (Signature-Based Detection),
- İstatistiksel anormallik temelli algılama (Statistical Anomaly-Based Detection),

- Durumsal protokol analizi temelli algılama (Stateful Protocol Analysis Detection)'dır.

İmza temelli algılama yönteminde, saldırı örüntülerinin bilinen saldırı imzaları daha önce tanımlanmış olup network paketlerinin incelenmesi sonucunda bu saldırı imzaları ile eşleşen kayıtların saldırı olarak tanımlanması esası vardır.

İstatistiksel anormallik temelli algılama yönteminde, ağ aktivitelerinin davranışsal modelleri oluşturulmuştur. Genellikle kullanılan bant genişliği, kullanılan portlar ve cihazlar bazında ağın tespit edilmiş normal aktivitelerinin dışında kalan herhangi bir davranış saldırı olarak nitelendirilebilir.

Durumsal protokol analizi temelli algılama yönteminde ise, iyi huylu aktivitelerin genel kabul görmüş tanımları önceden belirlenmiştir. Önceden belirlenmiş bu profiller gözlenen olayların durum protokolleri ile karşılaştırılarak bir karara varılır [80].

### 3.4.3 Güncel Saldırı Tespit ve Engelleme Araçlarının İncelenmesi

STS'lerin tarihi oldukça eskiye dayanmaktadır. 1986 yılında Denning tarafından yayınlanan makale [68] ile genel hatları ortaya konulan STS'ler, somut olarak 1988 yılından itibaren geliştirilmeye başlanmıştır. O tarihten bugüne kadar hem akademik hem de ticari kullanıma sunulan birçok STS yapısı mevcuttur. STS'ler bilgi ve bilgisayar güvenliğindeki yerini sağlamlaştırmaya başlaması ile birlikte, bu konuda yapılan çalışmalar da artmıştır. Konu ile ilgili olan araştırmacılar, farklı ortamlarda, farklı sistemler üzerinde ve farklı tekniklerle birçok STS yazılımı geliştirmişlerdir [8].

Haystack, MIDAS, IDES, W&S, ComputerWatch, NSM, NADIR, Hyperview, DIDS, USTAT, IDIOT, Janus, EMERALD, Bro, Ripper, Snort, Snortsam, Ossec, Snorby, PHPIDS, .NetIDS, Kfsensor, Selks ve Suricata geliştirilmiş STS yapılarından en önemli olanlarıdır. Tablo 3.3'te verilen ve STS'lerin gelişiminde önemli yer tutan bazı çalışmalar [8, 81, 82] bu bölümde alt başlıklarla kısaca açıklanmaktadır.

**Haystack:** Haystack saldırı tespit sistemi, 1988 yılında Amerikan Hava Kuvvetleri'nde kullanılan, OS/1100 işletim sistemini çalıştıran çok kullanıcılı Unisys 1100/60 ana bilgisayarları için tasarlanmış bir saldırı tespit sistemidir. Haystack aslında 6 farklı tür saldırının tespiti için geliştirilmiştir [8, 153]. Bu saldırılar şöyledir;

- Kırma girişimleri (attempted break-ins)
- Kılık değiştirilen saldırılar (disguised intrusion)
- Güvenlik kontrol sistemine sızma (penetration of the security control system)
- Sızma (leakage)

- Hizmet aksattırma (denial of service)
- Kötü niyetli kullanım (malicious use)

Haystack sistemi *Unisys* ve *Zenith* olmak üzere iki platforma ayrılmıştır. Unisys (Sperry) çalışma sistemi veri toplamayı denetlemekten sorumludur. Haystack'in Unisys parçası daha sonra bu denetleme takibini, birleşmiş denetleme takibine dönüştürür.

**MIDAS:** MIDAS (Multics Intrusion Detection and Alerting System), 1988'de NCSC (National Computer Security Center) tarafından, Bilgisayar Bilimleri laboratuvarı ve SRI (Stanford Araştırma Enstitüsü) işbirliği ile NCSC'nin ağa bağlanmış ana bilgisayarı Honeywell DPS-8/70 için saldırı tespiti sağlamak üzere geliştirilmiştir [8, 151]. MIDAS, sezgisel saldırı tespiti konsepti çevresinde geliştirilmiş uzman sistem tabanlı bir STS'dir.

**Tablo 3.3** Saldırı tespit sistemlerinin karşılaştırılması

| Yıl | STS | Saldırı Tespiti | Gerçek | Veri | Mimari yapı |
|---|---|---|---|---|---|
| 1988 | Haystack | Hibrit | Hayır | Sunucu | Merkezi |
| 1988 | MIDAS | Hibrit | Evet | Sunucu | Merkezi |
| 1988 | IDES | Anormallik | Evet | Sunucu | Merkezi |
| 1989 | W&S | Anormallik | Evet | Sunucu | Merkezi |
| 1990 | Comp-Watch | Anormallik | Hayır | Sunucu | Merkezi |
| 1990 | NSM | Hibrit | Evet | Ağ | Merkezi |
| 1991 | NADIR | İmza | Hayır | Sunucu | Merkezi |
| 1992 | Hyperview | Hibrit | Evet | Sunucu | Merkezi |
| 1992 | DIDS | Hibrit | Evet | Hibrit | Dağıtık |
| 1992 | ASAX | İmza | Evet | Sunucu | Merkezi |
| 1993 | USTAT | İmza | Evet | Sunucu | Merkezi |
| 1994 | DPEM | İmza | Evet | Sunucu | Dağıtık |
| 1994 | IDIOT | İmza | Evet | Sunucu | Merkezi |
| 1995 | NIDES | Hibrit | Evet | Sunucu | Merkezi |
| 1996 | GrIDS | Hibrit | Hayır | Hibrit | Dağıtık |
| 1996 | CSM | İmza | Evet | Sunucu | Dağıtık |
| 1996 | JANUS | İmza | Evet | Sunucu | Merkezi |
| 1997 | JiNao | Hibrit | Evet | Sunucu | Dağıtık |
| 1997 | EMERALD | Hibrit | Evet | Hibrit | Dağıtık |
| 1998 | Bro | İmza | Evet | Ağ | Merkezi |
| 1997 | Anzen flight jacket | Hibrit | Evet | Ağ | Hibrit |
| 1998 | Centrax security suite | Hibrit | Evet | Hibrit | Dağıtık |
| 1991 | Computer misuse detection system | Hibrit | Evet | Sunucu | Dağıtık |
| 1998 | Cross-site for security | İmza | Evet | Ağ | Dağıtık |
| 1999 | Cybercop monitör | İmza | Evet | Sunucu | Dağıtık |
| 1992 | Intruder alert | İmza | Evet | Hibrit | Dağıtık |
| 1996 | Kane security monitör | İmza | Evet | Sunucu | Dağıtık |
| 1997 | Netprowler(nee id-trak) | İmza | Evet | Sunucu | Dağıtık |
| 1996 | Netranger | İmza | Evet | Ağ | Dağıtık |
| 1996 | Realsecure | İmza | Evet | Ağ | Dağıtık |
| 1997 | Securenet pro | İmza | Evet | Ağ | Dağıtık |
| 1997 | Sessionwall-3 | İmza | Evet | Ağ | Dağıtık |
| 1998 | Smartwatch | İmza | Evet | Sunucu | Dağıtık |
| 1998 | Stake out | İmza | Evet | Ağ | Hibrit |
| 1998 | Tripwire | İmza | Hayır | Sunucu | Merkezi |
| 2000 | FIRE | Anormallik | Hayır | Ağ | Merkezi |
| 1998 | Snort | Hibrit | Evet | Ağ | Hibrit |
| 2014 | Selks | Hibrit | Evet | Ağ | Merkezi |
| 2009 | Suricata | Hibrit | Evet | Ağ | Merkezi |
| 2006 | Ossec | Hibrit | Evet | Sunucu | Dağıtık |
| 2003 | KfSensor | İmza, Kural Tabanlı | Evet | Sunucu | Merkezi |
| 2007 | .NetIDS | İmza, Kural Tabanlı | Evet | Sunucu | Merkezi |
| 2008 | PhpIDS | Hibrit | Evet | Sunucu | Merkezi |
| 2010 | Snorby | Hibrit | Evet | Ağ | Merkezi |

**IDES:** IDES (Intrusion Detection Expert System), Denning ve Neuman'ın 1985'li yıllarda yaptıkları çalışmalar ile gündeme gelmiştir. IDES için gereksinimleri ve tasarım modelini

ortaya koyan bu temel çalışmanın ardından, IDES üzerinde yapılan çalışmalar birçok araştırmacının da katkıları ile 1987-1992 yılları arasında yoğun olarak devam etmiştir. Bu nedenle IDES, bu yıllarda gerçekleştirilen birçok çalışmayı içermektedir.

IDES prototipi, denetleme verilerinde raporlanan kullanıcı davranışını, kullanıcının geçmiş hareket profiline göre normal olarak belirler. Zamanla değişebilen kullanıcı davranışlarına karşılık hareket profilleri her gün yenilenir. IDES, bir uzman sistem bileşeni içermektedir, bundan dolayı önceki saldırılardan edinilen bilgilere dayanan özel davranış örüntülerini, bilinen sistem açıklarını veya kuruluma özel güvenlik politikalarını tanımlayan kurallar içerir [8, 155].

IDES'in ilk orijinal versiyonu tek bir sunucu için tasarlanmıştır. Daha sonra yapılan çalışmalarda tekrar dizayn edilerek, farklı birçok hedef sistem için kullanılabilir hale getirilmiştir. IDES'in birçok versiyonu bulunmaktadır ve son olarak NIDES (Next-Generation Intrusion Detection Expert System) adını almıştır.

**W&S:** W&S (Wisdom and Sense–Hikmet ve His), Los Alamos Ulusal Laboratuvarları'nda geliştirilmiş olan bir STS'dir. W&S anormallik tabanlı bir çalışma yapısına sahiptir. Bu STS için ilk geliştirme çalışmaları 1984'de başlamış olmasına rağmen ilk olarak 1989 yılında sunulmuştur. Hikmet kısmı (W), incelenen sistem üzerindeki geçmişe dair denetleme iz (audit) ve verilerini inceleyerek normal davranış örüntüsünü oluşturması anlamına gelmektedir. His kısmı (S) ise W kısmında belirlenen davranışsal örüntülerin kural tabanı haline getirilip uzman sisteme verildikten sonra anormal davranışların yakalanması anlamına gelmektedir [8, 156].

**NSM:** NSM (Network Security Monitor) saldırı tespit sistemi, yukarıda anlatılan IDES sistemi gibi çeşitli düzenlemelerden geçmiştir. Bu STS'nin geliştirilmesi 1990-1994 yılları arasında olmuştur. NSM, bir bilgisayar ağını dinleyerek, ağın kullanımıyla ilgili bir profil geliştirir ve geçerli kullanımı geliştirdiği bu profil ile karşılaştırır. Elde edilen veri beklenen bağlantı verisiyle karşılaştırılır ve beklenen aralıkta çıkmayan her veri anormal olarak belirlenmiş olur [8, 151].

**NADIR:** NADIR (Network Anomaly Detection and Intrusion Reporter) saldırı tespit sistemi, 1991-1993 yılları arasında Los Alamos Ulusal Laboratuvarları'nda (LANL-Los Alamos National Laboratory) bilgisayar ağları mühendisliği grubu tarafından geliştirilmiştir. NADIR, uzman sistem tabanlı bir saldırı tespit yapısı kullanmaktadır. NADIR, sistem

üzerindeki kullanıcılar hakkında haftalık istatistikler tutar ve her hafta profil analizinin sonucunda detaylı raporlar oluşturur. Daha sonra bu istatistikleri uzman sistem yapısındaki kurallarıyla karşılaştırır [8, 157].

**Hyperview:** Hyperview saldırı tespit sistemi diğer STS yapılarından farklıdır. Bu STS, 1992 yılında geliştirilmiştir ve temelde iki kısımdan oluşmaktadır. Bu kısımlar izleyen ve öğrenen mantığıyla çalışan iki kısımdır. Bunlardan ilki sistemdeki davranışları izleyen ve bunları sınıflayan bir uzman sistem yapısı, ikincisi ise ilk kısımdan öğrendikleri ile eğitilen yapay sinir ağlarını içeren bölümdür. Dolayısıyla sistem yeni saldırı örüntülerini de bu öğrenen YSA yapısı ile tanıyabilmektedir [8, 151].

**USTAT:** Bu saldırı tespit sistemi 1993-1995 yılları arasında Unix temelli sistemlerde kullanılmak üzere geliştirilmiştir. USTAT (State Transition Analysis Technique for Unix Systems), durum geçiş analizi yöntemi kullanan bir saldırı tespit mekanizmasına sahiptir. Bu STS'de, eğer bir davranış saldırı için tanımlı bulunan durum geçişlerini yapıyorsa saldırı olarak sınıflandırılmaktadır [8, 158].

**DIDS:** DIDS (Distributed Intrusion Detection System), 1992 yılında geliştirilmiş olan ve dağıtık mimari yapılı saldırı tespit sistemlerini kapsamaktadır. Bu tip dağıtık mimari yapılı sistemlerde genel olarak bir ağın değişik noktalarından veriler toplanır ve bu veriler bir merkezde analiz edilir.

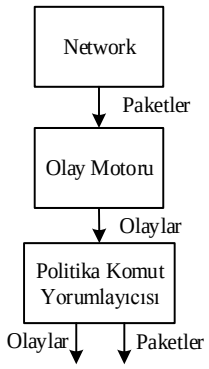
**IDIOT:** IDIOT (Intrusion Detection In Our Time) saldırı tespit sistemi, 1994-1996 yılları arasında CERIAS (Center for Education and Research in Information Assurance and Security)'da Kumar tarafından geliştirilmiştir [8, 151]. Kumar'ın tasarımı olan IDIOT, saldırı yöntemlerinin eşleştirme ve geçici karakteristiklerin karmaşıklığına dayalı olan bir sınıflandırma yöntemidir. Bu saldırı tespit sistemi, örüntü tanıma için petri-net'lerin kullanımına dayanmaktadır.

**JANUS:** 1994-1996'da Berkeley'de Wagner tarafından tez çalışması sırasında geliştirilmiş bir saldırı tespit sistemidir [8, 151].

**EMERALD:** EMERALD (Event Monitoring Enabling Responses to Anomalous Live Disturbances) saldırı tespit sistemi, 1997-1998 yıllarında geliştirilmiş ölçeklenebilir, dağıtık

mimari yapılı bir STS'dir. Bu alanda önemli çalışmalardan birisi olup, pek çok çalışmada kaynak olarak alınan bir saldırı tespit sistemidir [8, 81].

**BRO:** Bro saldırı tespit sistemi, 1998'de Vern Paxson tarafından geliştirilmiş bir STS'dir. Gerçek zamanda ağ trafiğini pasif olarak gözlemleyerek saldırı tespiti yapan bu sistem, ağ güvenliği izleyicisi olarak tanımlanmaktadır (Network Security Monitor). Birçok değişik özelliği bünyesinde barındırması açısından sıkça kaynak verilen saldırı tespit sistemlerinden biridir [8, 81, 82]. Bu STS'yi diğer saldırı tespit sistemlerinden ayıran temel özelliği, performansa dayalı analizler ve ağ sorunlarının çözümü için de destek sunmasıdır. Bro, bu özelliği ile ağ trafiği analiz aracı olarak nitelendirilebilir. Zaman içerisinde işlevsellik kazanmış olan Bro en güçlü STS yapıları arasına girmiştir. Geliştirilmesine ICSI'da (International Computer Science Institute) devam edilmektedir. Bro, iki temel bileşenden oluşur. Bu bileşenler, "Olay Motoru (event engine)" ve "Politika Komut Yorumlayıcısı (policy script interpreter)" 'dır. Şekil 3.13, Bro bileşenlerini göstermektedir. Buradaki olay motoru, ağ trafiğindeki paketleri ve protokolleri Bro'nun anlayacağı dile dönüştürür. Bro için belirtilen yaklaşık 320 olay türü vardır. Politika komut yorumlayıcısı ise Bro'nun betik dili ile yazılan olay işleyicilerinin çalıştırılması ile ilgilenir. Bu betikler ile ağ trafiği için oluşturulmuş olay türleri analiz edilir ve herhangi bir anormallik olması durumunda ne tür işlemlerin gerçekleştirileceği ve bunların kayıt politikaları belirlenir [82].




**Şekil 3.13** Bro bileşenleri

Bro'nun Özellikleri [82, 84];

- Linux, FreeBSD, MacOS gibi UNIX tabanlı işletim sistemlerinde çalışabilmektedir.
- Gerçek zamanlı ya da gerçek zamanlı olmayan analizler yapabilmektedir.
- Bro farklı sunucularda da çalışıp birbirleriyle iletişim sağlayabilir.

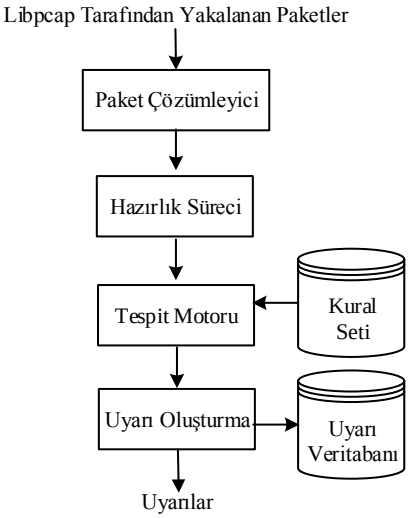
- Tüm HTTP trafiğini (sunucu/istemci istek ve cevapları, mime türleri, uri vb.), DNS istek ve cevaplarını, SSL sertifikalarını, SMTP oturumlarını, FTP trafiğini çözümleyerek kaydedebilmektedir. Ayrıca ağ akışını da kayıt altına almaktadır.
- Kayıtlar rahatça okunabilir şekilde (tab karakteriyle ayrılmış), ASCII formatında metin dosyalarına kaydedilir.
- Port bağımsız olarak uygulama katmanı protokollerinden DNS, FTP, HTTP, IRC, SMTP, SSH, SSL çözümlenebilmektedir.
- HTTP, FTP, SMTP, IRC trafiğinden geçen tüm dosyalarla ilgili bilgileri MD5/SHA1 özet değerleriyle birlikte metin formatında kaydedilebilmekte, istenildiği durumda bu dosyalar trafikten çıkarılıp belirtilen bir dizinde saklanabilmektedir.
- Dış kaynaklar kullanarak (özet değeri eşleştirmeleri, IP itibar tabloları) çeşitli zararlı yazılımları tespit edebilmektedir.
- IPv6 protokolü kapsamlı bir şekilde desteklenmektedir.
- Betik dili sayesinde anında e-posta mesajı atabilir ve ağı sonlandırabilir.
- Libpcap kütüphanesini kullanır.

**RIPPER:** Ripper saldırı tespit sistemi, 1999 yılında geliştirilen ve veri madenciliği yaklaşımını kullanan bir STS'dir. Ripper, DARPA değerlendirmesine katılmış olan sistemlerden biridir [8, 81].

**SNORT:** Snort saldırı tespit ve engelleme sistemi, 1990'lı yılların sonunda Martin Roesch tarafından geliştirilmiş, gerçek zamanlı ağ trafik analizi yapabilen ve paket kaydedebilen, açık kaynak kodlu, kural tabanlı bir sistemdir. Snort, imza ve anormallik tespiti yaklaşımlarının avantajlarını üzerinde barındırır. Snort, birçok saldırı ve sızma girişimini protokol ve içerik analizleri yaparak tespit edebilir ve tespit ettiği saldırıları çeşitli alarm mekanizmaları ile kullanıcıya bildirir. Saldırı imzalarının düzenli olarak güncellenmesi sayesinde oldukça farklı sayıda ve türde saldırıyı tespit edebilmek mümkündür. Açık kaynak kodlu ve sürekli geliştirilmeye açık olması popülerliğini artırmış bununla birlikte ticari saldırı tespit sistemleri ile kıyaslanabilir duruma gelmesini sağlamıştır.

Ticari yazılımlarda kaynak kodun açık olmaması, bu sebeple geliştirilebilir ve rahat yönetilebilir olmamaları ordu, kamu, özel sektör gibi üst düzey güvenlik isteyen kuruluşlar tarafından Snort'un daha çok tercih edilir olmasını sağlamıştır. Farklı kullanım modları sayesinde kullanıcıya kolaylık sağlaması ve yönetilebilir olması ise diğer önemli özelliklerindendir [8]. Snort, *paket çözümleyici, ön işlem, tespit motoru* ve *uyarı oluşturma*

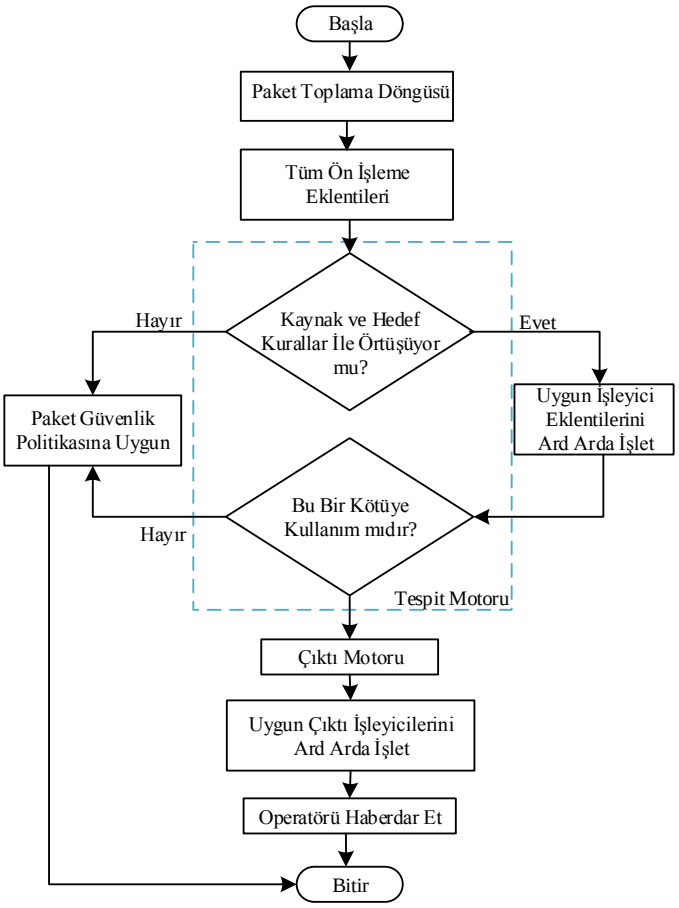
olmak üzere 4 temel modülden oluşmaktadır. Snort'un sahip olduğu bu modüller ve çalışma mantığı Şekil 3.14'te verilmiştir [84, 159].




**Şekil 3.14** Snort çalışma yapısı

*Paket çözümleyici*, Libpcap tarafından yakalanan katman-2 paketlerini ayrıştırır ve hazırlık sürecine gönderir. *Hazırlık süreci*, paketleri tespit motorunun anlayabileceği hale getirir. *Tespit motoru*, Snort'un temel bileşeni olup saldırı tespitinin yapıldığı kısımdır. *Uyarı oluşturma* kısmı ise tespit motoru tarafından yakalanan saldırılar için ne tür uyarı oluşturulacağını belirleyen kısımdır.

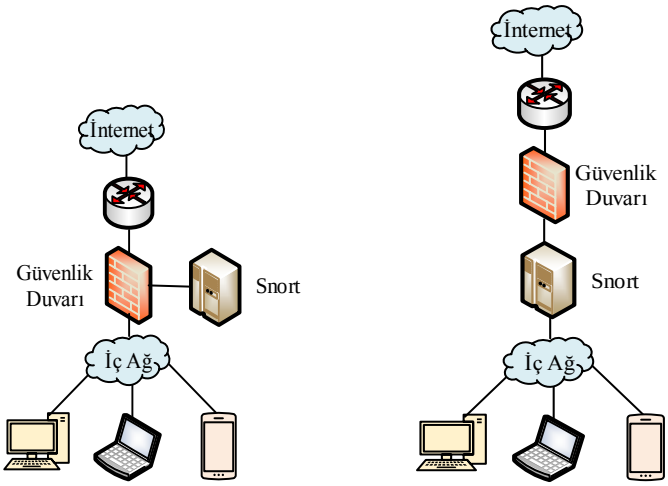
Snort, imza temelli ve anormallik tespiti yaklaşımlarının avantajlarını üzerinde barındırır. Protokol ve içerik analizleri yaparak birçok saldırı ve sızma girişimini tespit eder ve çeşitli alarm mekanizmaları ile kullanıcıyı uyarır. Saldırı imzalarının düzenli olarak güncellenmesi sayesinde oldukça farklı sayıda ve türde saldırıyı tespit edebilmek mümkündür. Açık kaynak kodlu ve sürekli geliştirilmeye açık olması popülerliğini artırmıştır. Ticari yazılımlarda kaynak kodun açık olmaması, ordu, kamu kuruluşları ve özel sektör gibi üst düzey güvenlik isteyen kuruluşlar tarafından Snort'un daha çok tercih edilmesini sağlamıştır. Farklı kullanım modları sayesinde kullanıcıya kolaylık sağlaması ve yönetilebilir olması ise diğer önemli özelliklerindendir [8, 81, 84, 159].



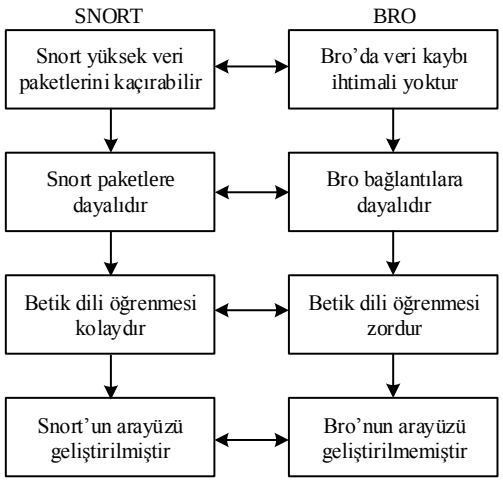

**Şekil 3.15** Snort akış şeması

Snort saldırı tespit sisteminin üç farklı çalışma modu bulunmaktadır. Bunlar, ağ trafiğini izleme modu (packet sniffer), paket kaydedici modu (packet logger) ve ağ temelli saldırı tespit sistemi modu (NIDS)'dur. Paket izleme modu, Tcpdump gibi çalışmakta ve bilinen paket izleme programlarına benzer yapıdadır. Paket loglama modu, istenilen türdeki paketlerin loglanıp analiz edilmek üzere saklanmasını sağlar. NIDS modu ise Snort'un asıl kullanım amacını barındırır. Bu mod, gelen paketlerin kullanıcı tanımlı kurallarla analiz edilmesini sağlar. Snort'un çalışma mantığını ve izah edilen farklı çalışma modlarını anlamak açısından Şekil 3.15'te verilen akış şeması yararlı olacaktır.

Snort'un temel olarak bir bilgisayar ağında konumlandırılması Şekil 3.16'da verildiği gibidir. Snort, güvenlik duvarının pasif izleme yeteneğini dinamik, öğrenen izleme kabiliyeti ile tamamlar. Buradan da anlaşıldığı gibi bilgisayar ağlarında firewall ile birlikte kullanılır ve olmazsa olmaz güvenlik araçlarından biridir. Şekil 3.17 ise, Snort ve Bro saldırı tespit sistemlerinin bir karşılaştırmasını içermektedir.




**Şekil 3.16** Snort'un konumlandırılma senaryoları




**Şekil 3.17** Snort&Bro karşılaştırılması

**Snortsam:** Snortsam, Snort saldırı tespit sistemine ses özelliği katan bir eklentidir. Snortsam iki ana kısımdan oluşmaktadır. Bu kısımlardan biri Snort için çıkış sistemidir, diğeri de güvenlik duvarı üzerinde ajan vazifesi görecek kısımdır. Snortsam kurulduktan sonra Snort kurallarına "fwsam" anahtar kelimesi eklenir. Snortsam bu anahtar kelimeyle kullanılır. Snort üzerine yazılan kurallar snortsam ajanı aracılığı ile güvenlik duvarına iletilir ve engellenmesi gereken trafik bu güvenlik duvarı tarafından durdurulur [8].

**Selks:** Debian üzerine, Suricata, Elasticsearch, Logstash, Kibana ve Scirius isimli bileşenlerini kullanarak oluşturulmuş, açık kaynak kodlu bir saldırı tespit sistemidir. Debian bileşeninin kullanılmış olması Selks'i farklı kılmıştır. Selks, Stamus Networks tarafından geliştirilmiştir ve GPLv3 lisansı ile dağıtılmaktadır. SELKS için tanımlanmış 10'dan fazla varsayılan IDS dashboard (pano) bulunmaktadır [84, 160]. *Bu IDS Panoları,* ALL,

ALERTS, DNS, FILE-Transactions, FLOW, HTTP, HTTP-Extended-Custom, PRIVACY, SSH ve TLS şeklindedir. Selks saldırı tespit sistemini oluşturan bileşenler ise;

**S-** Suricata IDPS,

**E-** Elasticsearch,

**L-** Logstash,

**K-** Kibana,

**S-** Scirius şeklindedir.

*Suricata:* OISF (Open Information Security Foundation) tarafından geliştirilmiştir ve GPL lisansı ile dağıtılan bir tür saldırı tespit sistemidir.

*Elasticsearch:* Esnek, güçlü ve ölçeklendirilebilir bir (full-text) arama ve analitik motorudur. Suricata üzerinden gelen alarmlar, Elasticsearch üzerinde depolanmaktadır. Daha sonra depolanan alarmlar analiz edilmektedir.

*Logstash:* SELKS mimarisinde Suricata tarafından üretilen ham verinin sorgulanarak analiz edilebilmesine olanak sağlamak amacı ile formatlanması (JSON) ve Elasticsearch'e yazılmasını sağlamaktadır.

*Kibana:* Suricata tarafından üretilmiş bir SELKS bileşenidir. Logstash tarafından formatlanarak Elasticsearch tarafından depolanan verinin analiz edilip görselleştirilmesine imkân sağlar.

*Scirius:* Stamus Networks tarafından geliştirilen ve Suricata için web tabanlı yönetim arabirimi imkânı sunan bir başka açık kaynak kodlu bir projedir. SELKS, eğer nodesktop şeklinde kurulacaksa minimum 4gb bellek, 2gb ram ihtiyacı olacağı düşünülmektedir. Fakat desktop seçeneği ile masaüstü sürümü kullanılacaksa minimum çift çekirdek kullanılması tavsiye edilir [84].

**Ossec:** Ossec (Open Source Host-based IDS), Trend Micro firmasınca desteklenen Linux, OpenBSD, Windows 2000/XP/2003/, FreeBSD, Vista, Solaris, OSX işletim sistemli bilgisayarlarda herhangi bir sorun oluşturmadan kurulabilen ve bilgisayar üzerinde yüklü olan uygulamaların çalışmalarını kontrol eden, açık kaynak kodlu bir STS aracıdır. Ossec, mevcut durumdaki veri paketleri, servis ve onların kaynaklarını kullanarak sistemi inceler ve kullanıcıları risklere karşı uyarır.

Ossec, tek bir oturumda çalışabilmeye olanak tanıyarak, merkezi yönetim sağlayabilen agent/master yapılı bir host-based STS'dir. Ossec, mevcut olan yapısından dolayı SIM/SIEM platformları ile de tümleşik olarak tasarlanabilir. Ossec, bu şekilde tanımlar

oluşturabilmesinin yanında başka özelliklere de sahiptir. Bu özellikler aşağıda açıklanmıştır [84, 161].

*Dosya Bütünlük Kontrolü:* File Integrity Monitoring (FIM) denen bu özellik, sistemde var olan dosyaların belirli aralıklarla kontrol edilmesi kuralını temel almaktadır. Yapılan her kontrolde, var olan dosyaların toplamından MD5 tarzı şifreleme algoritmalarıyla bir şifre oluşturulur. Dosyaların boyutlarında veya diğer özellikler bazında yapılan değişikliklerin tarih ile kayıt altına alınması yoluyla oluşturulan şifre, yapılan kontrollerde bir önceki şifre ile kıyaslanır. Böylelikle dosyalarda herhangi bir değişiklik olup olmadığı gözlemlenir. Değişikliğe uğramış dosyaların tespitini ve yapılan değişikliklerin sistem yöneticisine bildirilmesini temel hedef edinmiştir.

Sisteme sızmayı amaçlayan her çeşit saldırının ortak noktasının sistemde mevcut olan dosyalar üzerinde değişiklik yapmak ya da sisteme bir takım dosyalar ilave etmek olduğu biliniyorsa, HIDS'ler açısından dosya bütünlük kontrolünün önemli bir bileşen olduğu söylenebilir. Sistemlerde meydana gelebilecek bu değişiklikleri, değişiklik olduğu an tespit etmek büyük önem arz etmektedir.

*Rootkit Tespiti (Rootkit Detection):* Rootkit, sistemde mevcut olan süreç/dosya yapılarını gizlemek şartı ile işletim sisteminin gerçeği görmesine engel olmak amacıyla geliştirilmiş olan yazılımlardır. Örnek verilecek olursa; "ps" komutunun görevini değiştirmek için oluşturulan rootkit, sistemdeki kullanıcıların tüm süreci görüntülemesi için verilen izni kaldıracaktır. Bu özellik ile Ossec, sistemdeki mevcut rootkitlerin tespit edilmesini sağlayarak tehlikelere karşı uyarı verir [161].

*Log Monitoring:* Değişikliklerin tespit edilmesinin hemen ardından ise kontrolü yapılması gereken yapılar log dosyalarıdır. Bu amaca hizmet etmek ve bu soruna çözüm bulmak amacıyla sistem loglarını izleyen Ossec, analiz işlemini gerçekleştirir ve bir problemin tespitinde hemen alarm oluşturarak sistem yöneticisine bu konuda bilgi verir. Bu duruma örnek vermek gerekirse, sistem üzerinde bir paket kurulduğu vakit ya da web sunucusunun loglarına bir sızma girişimini belirten satırlar gelmeye başladığında, alarm üretilir ve bu durumun farkına varılması, sisteme sızma girişiminin tespitine katkı sağlamış olur.

Ossec, bu özellikleri ile rakiplerinden daha fazla ön plandadır. Açık kaynak kodlu bir yazılım olması ve genel kamu lisansı ile dağıtılmasından dolayı da STS araçları arasında en fazla tercih edilenler arasındadır.

**Suricata:** Suricata, ticari kaygı barındırmayan bir oluşum olan OISF (Open Information Security Foundation) tarafından geliştirilmiş, açık kaynak kodlu bir saldırı tespit sistemidir. İmza/kural tabanlı olarak çalışan Suricata, gelen saldırıları tespit etmesinin yanında bu saldırıları engelleme özelliğine de sahiptir. Bu özellik ile yakalanan paketlerin çalıştırılmasına, reddedilmesine ve istatistiksel bilgilerinin tutulmasına da imkân sağlar.

Suricata tarafından sistem çalışması yapılırken kullanılacak mod, başlangıçta verilecek parametreler ile düzenlenmektedir. Çalışma sürecinde ise kural ekleme-çıkarma işlemi yapılmasına olanak tanıyan Suricata, IPv6 yapısını da desteklemektedir.

Bu özelliklerden farklı olarak Suricata, bazı ek özellikler ile de yenilik getirmiştir. Bu özellikler [84, 162];

*HTTP kütüphanesi:* HTTP trafiklerinin ayrıştırılmasına olanak tanıyarak, saldırganların STS'leri atlatarak sisteme sızma girişimlerini engelleyen bir normalizasyon aracıdır.

*Çoklu iş parçacıkları:* Paket işleme işlevi farklı bölümlerde yapılmaktadır. Yani paketler farklı makinelerde çalışıyormuş gibi dağıtık olarak işlenmekte ve performansta artış sağlanarak yük dengesi yapılmaktadır. Bu işlem Multi- threading olarak bilinmektedir.

Suricata, Unix soket modunda işlem görerek ağ trafiğinin *pcap* formatında kayıt edilmesini sağlar. Kaydedilen bu ağ trafiği kayıtları daha sonra offline olarak analiz edilebilmektedir. Ayrıca Suricata, HTTP istekleri, SSH bağlantıları, TLS el sıkışmalarını kaydedebilmeye olanak tanımaktadır. Suricata, işleyişinde 4 kısma ayrılmaktadır [84]. Bu kısımlar;

*Paket Yakalama Modülü: Capture Module* olarak bilinen bu modül, belirlenmiş olan paketlerin ethernet kartından Suricata'ya iletilmesini sağlamaktadır. Bu modül, paketlerin farklı veri bağı çözümleyicileriyle uygun olarak çalıştırılmasını sağlar.

*Paket Çözümleme Modülü: Decoding Module* denilen bu modül, paketlerin ara belleğe alınarak Suricata'nın kullanabileceği veri yapısına çevrilmesinden sorumludur. Burada paketler veri linklerine göre ayrılıp, çözümleyiciler tarafından işlenmiş olurlar.

*Akış İşlemleri Modülü: Stream Module* denilen bu yapı ise temelde 3 görev üstlenmektedir [84]. Bunlar;

- Ağ bağlantılarının doğru olması için akış takibi,
- TCP bağlantılarında tekrar oluşturulan ana akış için paketlerin sıralanması,
- HTTP - DCERPC analiziyle uygulama katmanı denetimi.

*Tespit Modülü:* Detect Module, yapılandırılmış kuralların belirlenip yüklenmesine, tespitin başlamasına ve kuralların paketler ile eşleştirilmesine olanak tanımaktadır.

Suricata'da tanımlanmış 3 profil mevcuttur. "yüksek, orta ve düşük" olmak üzere oluşturulmuş bu profiller içerisinde, varsayılan profil, bellek kullanımı ile performans arasında bir denge sağlayan "orta" dır.

**Snorby:** Snorby, Snort, Suricata gibi popüler saldırı tespit sistemi uygulamaları gibi ağ güvenliğini izlemek amacıyla, Ruby dili ile yazılmış bir uygulamadır [163]. Ruby ile bir deneyim kazanmadan, çalışma yapılmadan Snorby ile uğraşmak zor olabilmektedir.

Saldırı tespit sistemimizde doğru ayar ve güncellemeleri yapmak için Snorby'i iyi bir şekilde kavramak önemlidir. Snorby, sensörlerden gelen STS/SES alarmlarını toplayan, merkezi bir konsol olarak da kabul edilebilir. Snorby'nin diğer STS uygulamalarından farkı daha fonksiyonel olmasıdır.

**PHPIDS:** PHPIDS, PHP web uygulamalarında kullanılan açık kaynak kodlu, hızlı, kullanımı kolay, iyi yapılandırılmış ve güvenilir bir saldırı tespit sistemidir. PHPIDS, web uygulamaları üzerinde Çapraz Site Betikleme (XSS), SQL Enjeksiyonu (SQL Injection), Başlık Enjeksiyonu (Header Injection), Dizin Atlama (Directory Traversal), RFI/LFI, Hizmet Engelleme (DoS) gibi saldırı türlerini analiz edebilmektedir [164]. PHPIDS, kötü bilinen uygulamaları tespit etmek için birkaç düzenli uygulama ile çalışır. Bunu yapmak için temelde kara liste yaklaşımını, bilinmeyen saldırı örüntülerini yakalamak için sezgisel yaklaşımlar ile birleştirir. Özel dönüşüm algoritmaları sayesinde PHPIDS, siteme sızan saldırıların tespit edilmesi ve bunların kayıtlarının tutulmasını sağlar.

**.NETIDS:** .NETIDS olarak isimlendirilen bu araç, web tabanlı .NET uygulamalarını korumanın diğer bir yoludur. Bu araç, web uygulamalarına karşı yapılan saldırıları tespit yeteneğine sahiptir. .NETIDS ile oluşturulan saldırı tespit sisteminin dosyaları, mümkün olan saldırılarla etkileşime geçmek için filtreleme kuralları ve fonksiyonlar barındırır. .NetIDS, daha önce var olan PHPIDS'in .Net framework için geliştirilmiş olan bir türüdür [165]. .NETIDS'in kütüphanesi, CLS uyumludur ve PHP versiyonundaki filtreleme sistemleri benzer şekilde uygulanmıştır. .NETIDS sistemleri ile kullanılan herhangi bir .Net uygulaması için ek koruma katmanları oluşturmak temel amaçtır. Bunun yanında .NETIDS, XSS'in belirlenmesi, SQL Injection, zararlı JS nesne ve metotlarına karşı koruma, gelişmiş loglama fonksiyonları, filtreleme kuralları, arayüzleri kategorize etme ve etiketleme özelliklerini içerir.

**Firestorm:** Firestorm, son derece yüksek performansa sahip bir ağ saldırı tespit sistemidir. Firestorm son derece esnektir ve bundan dolayı yönetilebilir bir yapıya sahiptir. Bu özelliğiyle diğer sistemlere göre çok daha iyi bir performans gösterir. Firestorm ilk olarak sadece bir sensör niteliğindeydi, daha sonra uzaktan kontrol, raporlama ve analiz için gerekli destekler verilmesi planlanmıştır.

Ağ üzerinde şüpheli paketleri belirlemeye yarayan Firestorm, durum bilgilerini analiz edebilir, IP çerçevelerini yeniden oluşturup, TCP bağlantılarını izleyebilir. Firestorm, uygulama katmanı üzerinde çalışabilen tüm protokolleri çözebilme kapasitesine ve anormallik tespiti modüllerini destekleyebilecek altyapıya sahiptir. Firestorm ayrıca Snort gibi ağ trafiğini kontrol eder ve bu analiz sonucu oluşan kayıtları tutar. Firestorm, DoS saldırılarına karşı kendini koruyabilme ve uyarı verme yeteneğine sahiptir [84, 166].

**Kfsensor:** KFSensor, Windows platformunda çalışan honeypot tabanlı bir STS'dir. KFSensor, honeypot gibi zayıf bir sistem gibi görünerek hacker, solucan ve trojanları tespit eder. Bu tespit işlemini gerçekleştirirken port taramaları yaparak sisteme sızmaya çalışanları belirler ve .txt uzantılı log kaydı tutar. KFSensor, Windows tabanlı olarak kurumsal bir ortamda kullanılmak amacıyla tasarlanmış, Snort uyumlu imza motoru, Windows ağ protokolleri ve emülasyonlar gibi birçok yenilikçi ve benzersiz özellikler barındıran bir STS'dir. Kendine ait olan GUI tabanlı yönetim konsolu, kapsamlı dokümantasyon ve düşük bakım maliyeti nedeniyle KFSensor, bir kuruluşun ağ güvenliğini sağlamak için uygun bir çözüm sunar.

## 3.5 Saldırı Tespit Sistemlerinde Kullanılan Veri Kümeleri

Bir STS oluşturulurken üzerinde dikkatle durulması gereken en önemli konulardan birisi de kullanılacak olan veri kümesidir. Saldırı tespit sistemleri açısından veri kümesi, özellikle geliştirilecek olan STS'nin eğitim ve test aşamalarında saldırı örüntülerini tanımlamak için gerekli olan ve içerisinde saldırı verilerini barındıran, ağ paketleri veya log kayıtlarından elde edilmiş verilerden oluşmaktadır. Günümüze kadar yapılan çalışmalara bakıldığında, geliştirilen bazı STS uygulamalarında geliştiriciler, veri kümelerini kendileri oluşturmuşlardır. Veri kümesinin oluşturulması çok zahmetli ve bir o kadar da maliyeti yüksek olan bir çalışma süreci gerektirmektedir. Saldırı tespit sistemi geliştirme sürecinde, sınırlı olan zaman ve iş gücü, farklı teknikler, algoritmalar geliştirmek yerine, veri kümesi oluşturmaya harcandığı takdirde, geliştiricilerin yükleri oldukça artmaktadır. Ayrıca geliştirilen saldırı tespit sistemi çalışmalarında oluşturulan veri kümeleri, başarım açısından

iyi olsa da bu sonuçlar gerçeklikten uzak olabilmektedir. Geliştirilen sistemlerin tasarım ve uygulama aşamalarının hızlandırılması ve konu ile ilgili objektif bir test ortamının oluşturulması açısından güncel ve standart hale gelmiş veri kümelerine ihtiyaç duyulduğu bir gerçektir [8].

Veri kümesi gereksiniminin karşılanmasına yönelik olarak yapılan çalışmalar, genel itibariyle 1998-2000 yılları arasında sınırlı kalmıştır ve oldukça değerli sonuçlara imza atılmasına olanak sağlamışlardır. Günümüzde özellikle güncel saldırı örüntülerine ve gerçeğe uygun veri kümelerinin oluşturulması gerekmektedir. Bu bölümde literatürde yer alan, daha önce yapılmış ve halen kullanılmaya devam eden IDEVAL, DARPA ve KDD'99 veri kümeleri hakkında bilgiler verilmektedir.

### 3.5.1 IDEVAL

MIT Lincoln Laboratuvarı, Bilgi Sistemleri Teknolojisi (Information Systems Technology - IST) grubunun, DARPA (Defence Advanced Research Projects Agency) ve AFRL (Hava Kuvvetleri Araştırma Projeleri Ajansı-Air Force Research Projects Agency) desteğiyle yürüttüğü çalışmaların sonucunda, saldırı tespit sistemlerinin değerlendirilmesi ve karşılaştırılması için IDEVAL (Intrusion Detection Evaluation) adıyla ilk standart veri kümesinin gövdesini oluşturmuştur. Bu çalışmada, AFRL ile birlikte saldırı tespit sistemlerinin, ilk düzenli, tekrarlanabilir ve önemli-istatistiksel ölçütleri belirlenmiştir. Bu ölçütler, kontrol edilecek her sistem için saldırı tespit ve yanlış-alarm olasılığını içermektedir. Bu ölçütler, mevcut STS'lerin tasarımına, yeni yapılacak araştırma çalışmalarına yön verme ve objektif bir kalibrasyon sağlamak açısından önemlidir [8].

STS'ler için bir değerlendirme çalışması, Lincoln Laboratuvarları'nda 1998-2000 yılları arasında gerçekleştirilmiştir. Bu değerlendirme çalışmasında, gerçek zamanlı olmayan (off-line) veri kümeleri oluşturulmuş ve bu veri kümeleri, geniş saldırı ve ağ trafiği örnekleri ile araştırmacılar tarafından kullanılabilecek hale getirilmiştir.

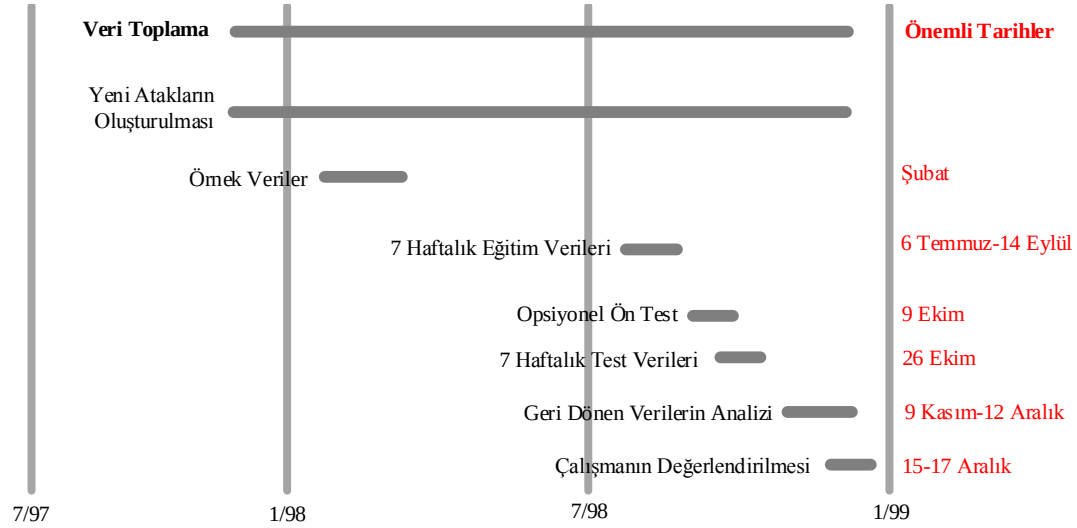
DARPA saldırı tespit değerlendirme çalışmaları sürecinde 1998 ve 1999 yıllarında DARPA ve IDEVAL veri kümeleri olmak üzere iki veri kümesi elde edilmiştir. Bunların dışında 2000 yılında özel senaryolar için üç farklı veri kümesi daha oluşturulmuş ve bu veri kümeleri bilgi güvenliği araştırmacılarının kullanımına sunulmuştur [8, 167].

IDEVAL adıyla 1998 ve 1999 yıllarında geliştirilen veri kümeleri, gerçek zamanlı ve gerçek zamanlı olmayan değerlendirme olmak üzere iki kısımdan oluşmaktadır. Lincoln Laboratuvarlarında oluşturulan veri kümeleri, STS'lerin gerçek zamanlı olmayan (off-line)

değerlendirilmesinde kullanılır. IDEVAL 1998 ve 1999 değerlendirmesinin gerçek zamanlı kısmı hakkında ayrıntılı bilgi AFRL'den elde edilebilir.

### 3.5.2 DARPA 1998

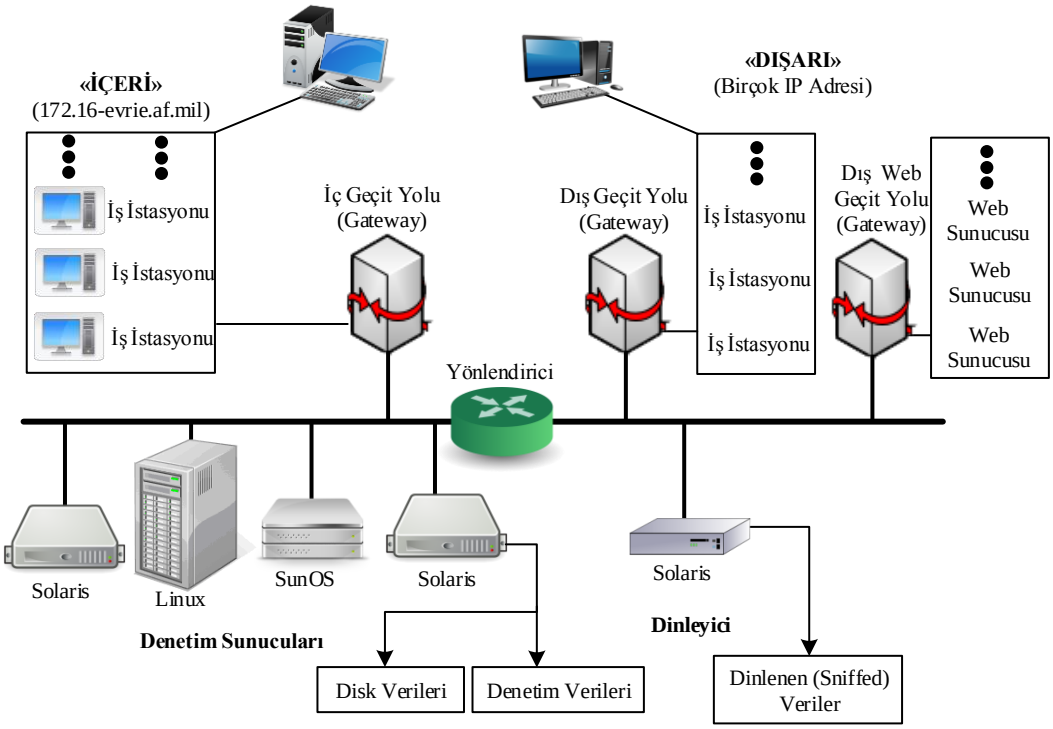
DARPA 1998 veri kümesi, 1998 IDEVAL veri kümesinin gerçek zamanlı olmayan değerlendirme kısmı için geliştirilmiştir olan bir veri kümesidir. Şekil 3.18'de bu veri kümesinin oluşturulma aşamaları görülmektedir [8, 167].




**Şekil 3.18** DARPA 1998 veri kümesinin oluşturulma aşamaları

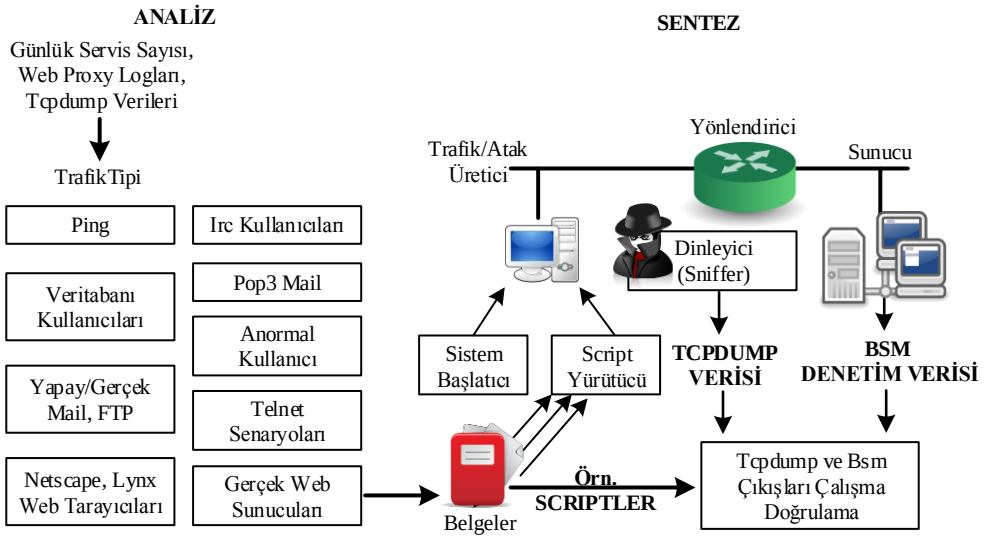
1998 yılının ilk olarak Şubat ayında 10 dakikalık örnek bir veri kümesi oluşturulmuştur. Bu veri kümesi, STS'lerin başarımlarının nasıl derecelendirildiğini göstermek amacıyla, sadece bazı örnek saldırı tiplerini içerir. Bunlar; guess, ping-sweep, port-scan, phf, rlogin, rsh ve rcp'dir. Örnek veri kümesinden sonra, aynı yılın Mayıs ayında eğitim verilerinin alt kümesi olan 4 saatlik veri kümesi oluşturulmuştur. Oluşturulan bu veri kümesiyle, ağa dair bazı ilk veriler oluşturularak, verinin doğru okunabilmesi ve değerlendirme için yeterli bilginin sağlanabilmesi amaçlanmıştır. Oluşturulan veri kümesi, gerçek eğitim veri kümesinin ilk 4 saati değildir ve ilk hafta elde edilen verilere benzer olmasına rağmen özdeşi de değildir. 4 saatlik veri kümesi, örnek 10 dakikalık veri kümesindeki saldırıların haricinde 2 farklı saldırı içerir. Bunlar *dict* ve *eject* türünde saldırılardır [8, 167].

1998 IDEVAL eğitim veri kümeleri, 7 haftalık değerlendirme sonucunda elde edilen verilerden oluşmaktadır. Bu verilerin oluşturulması için kullanılan simulasyon ortamının ayrıntıları Şekil 3.19'da verilmiştir. Ağda iç ve dış olmak üzere iki ortam oluşturulmuştur. İçerideki sunucularda denetim verileri ve ağ paketlerinin toplanması sağlanmıştır [8].



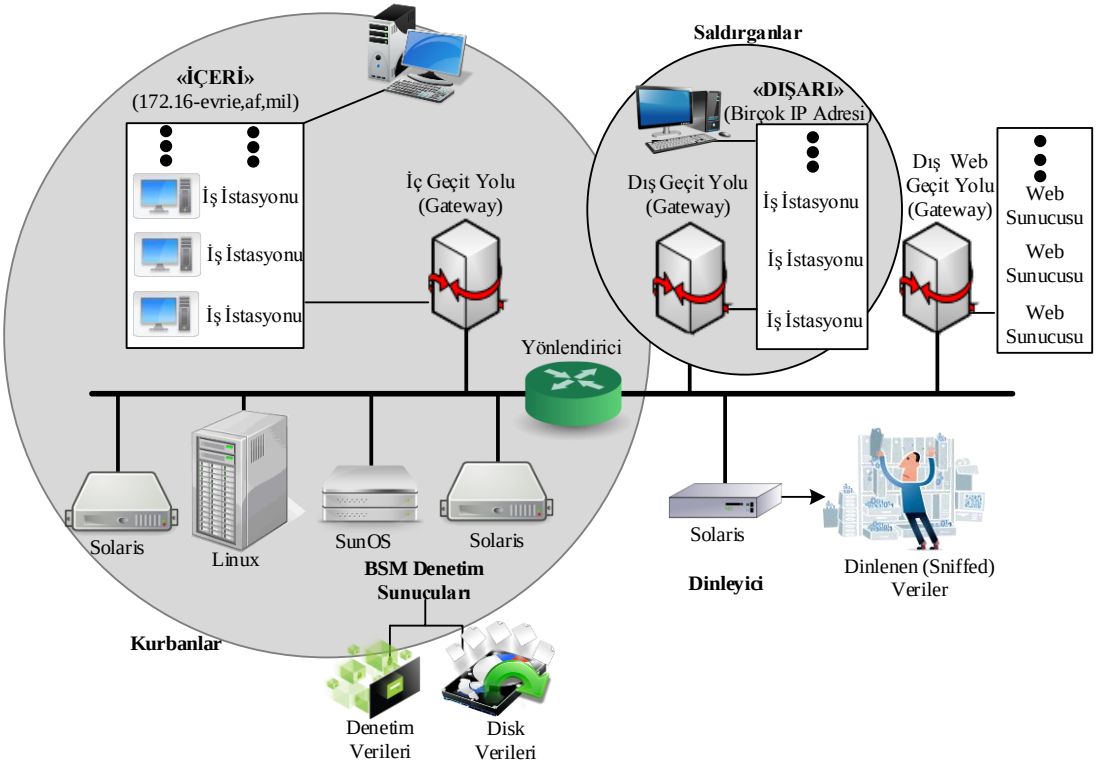

**Şekil 3.19** IDEVAL simülasyon ortamı

Simülasyonu yapılan ağ içerisinde dört ana "kurban" makine bulunmaktadır. Bunların üzerinde Sun OS, Solaris, Linux, ve Windows NT işletim sistemleri çalışmaktadır. Genel olarak trafik oluşturucular, yüzlerce sunucuyu ve çeşitli uygulamaları çalıştıran ve İnternet bağlantısına sahip kullanıcıların simülasyonunu yapmaktadırlar. Ağ trafiğinin oluşturulmasına yönelik olarak bir diyagram Şekil 3.20'de verilmiştir.




**Şekil 3.20** IDEVAL için ağ trafiğinin oluşturulması ve analiz edilmesi

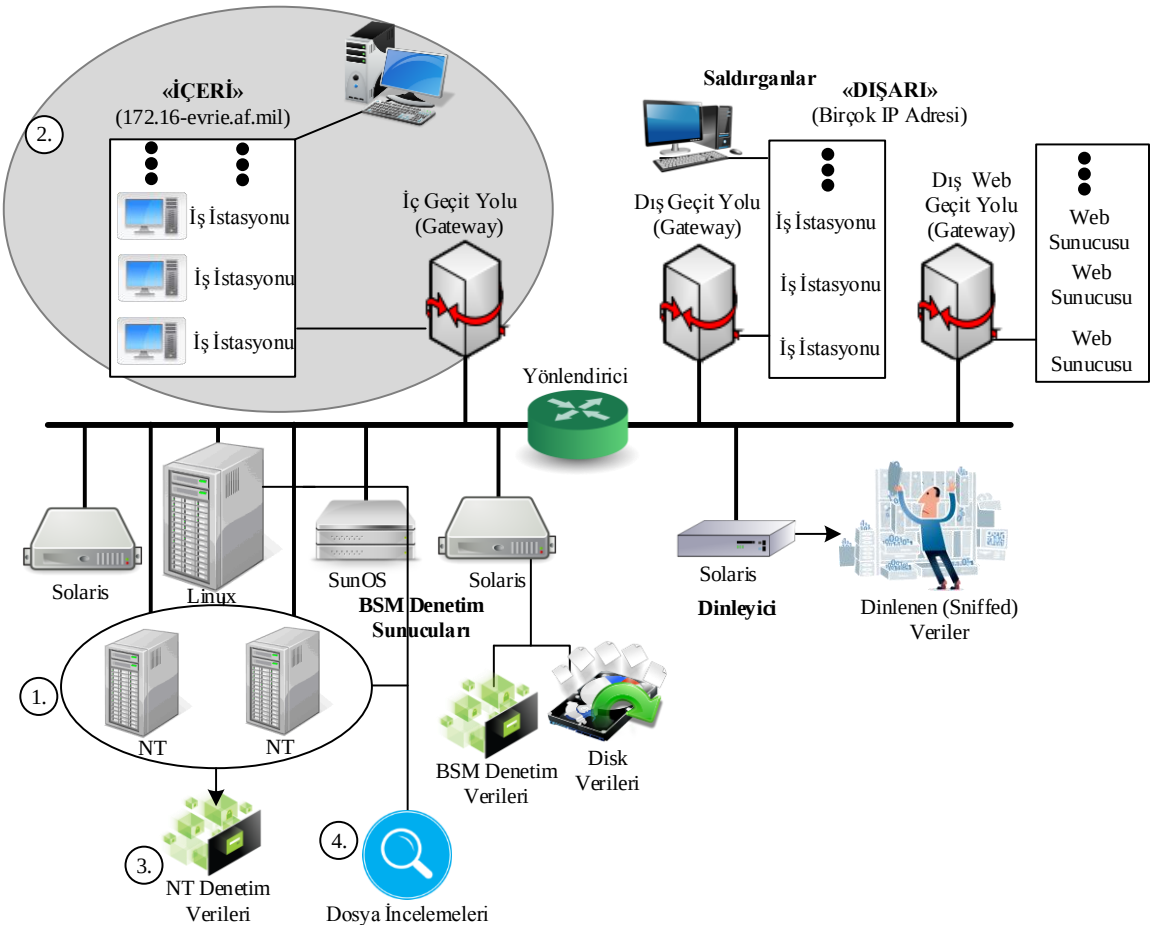
Simülasyon ortamındaki saldırganlar ve bu saldırılara maruz kalan kurbanlar Şekil 3.21'de gösterilmektedir. Burada, saldırganların sadece dışarıdan birileri olabileceği varsayılarak tasarım yapılmıştır. İç ağdaki bilgisayarların tamamı kurban olarak kabul edilmiştir.




**Şekil 3.21** IDEVAL simülasyonundaki saldırgan ve kurbanlar

### 3.5.3 DARPA 1999

DARPA 1999, 1998'de geliştirilen veri kümesinin değiştirilmesi ile elde edilmiş bir veri kümesidir. Böyle bir değişikliğe gerek duyulmasındaki amaç, yeni ve farklı bazı saldırıların da veri kümesine dahil edilmesidir. DARPA 1998 veri kümesinin geliştirilmesinde kullanılan ağ yapısına eklenen bu yeni özellikler Şekil 3.22'de gösterilmektedir [8, 167].




**Şekil 3.22** DARPA 1999 veri kümesi ağ yapısı

Şekil 3.21'de sunulan ağ yapısının bazı yeni özellikleri şunlardır [8];

- NT iş istasyonları- NT trafiği ve saldırıları
- İç tehditler (iç ağdan gelebilecek saldırılar)
- NT denetleme verilerinin analizi
- Dosya sistemi analizi
- Eğitim verilerinde saldırı olmayan günler sağlamak
- Eğitim ve test verilerinde daha az örtüşen saldırılar
- Liste dosyası başarım değerlendirme prosedürlerini tekrar gözden geçirme.

DARPA veri kümelerinin geliştirilmesi ile STS'lerin değerlendirilmesi sağlanmıştır. Simülasyon ağından toplanan ağ trafiği ve günlük dosyaları kullanılarak offline bir değerlendirme gerçekleştirilmiştir. STS'ler, gerçek zamanlı test için AFRL'ye gönderilmiştir. Bu sistemler, AFRL ağ test yatağına eklenmiş ve gerçek zamanlı olarak, normal aktiviteler arasında saldırılar belirlenmeye çalışılmıştır [8].

### 3.5.4 KDD'99

KDD'99 veri kümesi, DARPA veri kümesinin çeşitli işlemlerden geçirilmesi ile elde edilmiş 41 özellikten oluşur. KDD'99 veri kümesinin oluşturulmasının amacı, son zamanlarda farklı tekniklerle gerçekleştirilmek istenen STS'ler için eğitim ve test işlemlerinde kolaylık sağlamaktır. KDD'99 veri kümesi ile eğitim ve test sonuçlarının daha hızlı alınabilmesi, araştırmacılar için büyük bir avantaj olmuştur.

Bu veri kümesinde, 9 temel ve 32 adet türetilmiş olmak üzere toplamda 41 tane özellikten oluşan bir özellik haritası çıkarılmıştır. Bu 41 özellik 3 temel kategoriye ayrılarak ifade edilmiştir [8]. Bunlar aşağıda sıralandığı gibidir.

- İçerik özellikleri (content features)
- Sunucu tabanlı trafik özellikleri (host-based traffic features)
- Zamana bağlı trafik özellikleri (time-based traffic features)

Aşağıda verilen Tablo 3.4, Tablo 3.5 ve Tablo 3.6, yukarıda verilen bu 3 temel kategoriye ait veri özelliklerini göstermektedir [8].

**Tablo 3.4** KDD'99 içerik özellikleri

| **Özellik adı** | **Tanım** | **Tip** |
|---|---|---|
| Duration | Bağlantı uzunluğu | Sürekli |
| protocol_type | Protokol tipi | Ayrık |
| Service | Servis tipi | Ayrık |
| src_bytes | Kaynaktan hedefe veri | Sürekli |
| dst_bytes | Veri byte sayısı | Sürekli |
| Flag | Bayrak | Ayrık |
| Land | Kaynak ve hedef IP aynı ise 1 değilse 0 | Ayrık |
| Wrong_fragment | Yanlış parçalama | Sürekli |
| Urgent | Acil paket sayısı | Sürekli |

Tablo 3.4'te verilen içerik özellikleri, temel TCP/IP bağlantılarından alınan özelliklerdir. Herhangi bir ön işlem adımına gerek duyulmadan, ağ trafiği üzerinden alınabilen özelliklerdir. Bu sebeple elde edilmesi daha kolay olan verilerdir.

**Tablo 3.5** KDD'99 veri kümesi için sunucu tabanlı trafik özellikleri

| **Özellik adı** | **Tanım** | **Tip** |
|---|---|---|
| hot | "hot" göstergesi | Sürekli |
| num_failed_logins | Hatalı giriş sayısı | Sürekli |
| Logged_in | Giriş başarılı ise 1 değilse 0 | Ayrık |
| num_compromised | Gizliliğin ihlal edilme sayısı | Sürekli |
| root_shell | "Root Shell" elde edildiyse 1 değilse 0 | Ayrık |
| su_attempted | "Su Root" komutu girildiyse 1 değilse 0 | Ayrık |
| num_root | "Root" erişim sayısı | Sürekli |
| num_file_creations | Dosya oluşturma işlemleri sayısı | Sürekli |
| num_shells | Shell promptlarının sayısı | Sürekli |
| num_access_files | Kontrol dosyalarına erişim işlemleri sayısı | Sürekli |
| num_outbound_cmds | ftp oturumunda giden komut sayısı | Sürekli |
| is_hot_login | Giriş "hot" listesindeyse 1 değilse 0 | Ayrık |
| is_guest_login | Giriş "guest" ise 1 değilse 0 | Ayrık |

Sunucu tabanlı trafik özellikleri, etki alanı bilgisi ile ortaya çıkan bağlantı özellikleridir.

**Tablo 3.6** KDD'99 veri kümesi için zamana bağlı trafik özellikleri

| **Özellik Adı** | **Tanım** | **Tip** |
|---|---|---|
| count | Aynı sunucuya önceki iki bağlantıyla aynı | Sürekli |
| serror_rate | "SYN" hata bağlantılarının yüzdesi | Sürekli |
| rerror_rate | "REJ" hata bağlantılarının yüzdesi | Sürekli |
| same_srv_rate | Aynı servise bağlantıların yüzdesi | Sürekli |
| diff_srv_rate | Farklı servislere bağlantıların yüzdesi | Sürekli |
| srv_count | Aynı servise önceki iki bağlantıyla aynı | Sürekli |
| srv_serror_rate | "SYN" hata bağlantılarının yüzdesi | Sürekli |
| srv_rerror_rate | "REJ" hata bağlantılarının yüzdesi | Sürekli |
| srv_diff_host_ | Farklı servislere bağlantıların yüzdesi | Sürekli |

Zamana bağlı trafik özellikleri, "aynı sunucu" ve "aynı servis" özellikleri kullanılarak çıkarılan özelliklere verilen isimdir. "Aynı sunucu" özellikleri, son iki saniye içerisinde aynı sunucuya yapılan bağlantıların incelenmesi ile elde edilir. Benzer olarak "aynı servis" özellikleri son iki saniye içerisinde aynı servise yapılan bağlantıların incelenmesi sonucu elde edilir [8].

# 4. WEB TABANLI SALDIRILAR İÇİN GERÇEK ZAMANLI HONEYPOT TEMELLİ HİBRİT BİR SALDIRI TESPİT VE ENGELLEME YAKLAŞIMI

## 4.1 Giriş

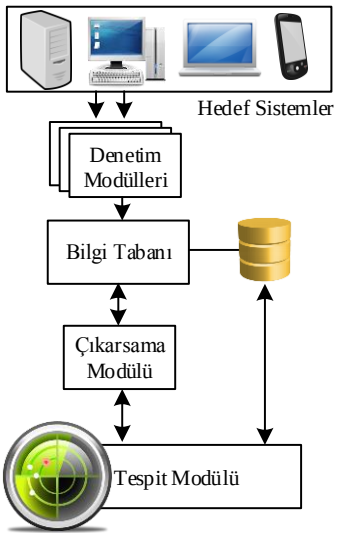
Bilişim dünyasındaki olağanüstü gelişmelere paralel olarak, kişi, kurum ve kuruluşlar açısından bilgi sistemleri güvenliğinin önemi de artmaktadır. Bilgi sistemlerinde güvenliğin sağlanması için birçok farklı çalışma yapılmaktadır. Hem donanımsal hem de yazılımsal olarak gerçekleştirilebilen bu çalışmalar ile güvenliğin sağlanması amaçlanmaktadır [6]. Bu amaçla kullanılan güvenlik yazılımları kişi kurum ve kuruluşlara ait sistemlerin korunmasında büyük önem taşımaktadır. Gelişen teknoloji ile birlikte bilişim dünyasındaki ilerleme seviyesi ülkeler arası ilişkilerde yeni bir güç unsuru haline gelmiştir. Askeri, ekonomik vb. gibi ülkeler bazında güç unsuru olan kavramlar arasına günümüzde "siber uzay", "siber güç", "siber ordu", "siber savaş" olarak adlandırılan yeni kavramlar eklenmiştir. Artık fiziksel, sıcak savaş ve saldırılardan farklı olarak siber saldırılar da gerçekleştirilmektedir. Bu sebeple ülkeler artık askeri ordularının yanı sıra siber ordularını kurmakta ve siber uzay dediğimiz yapıda saldırılar gerçekleştirmektedir.

2013 yılında Symantec firmasının "İnternet Güvenliği Tehdit Raporu" adıyla yayınladığı incelemeye göre; dünya genelinde 157 farklı ülkede toplam 69 milyon siber saldırı tespit edilmiştir. Bilişim sistemlerinin tarihsel sürecine bakıldığında, sistemler geliştikçe, gelişen teknolojinin zafiyetlerinin yol açtığı bir takım yeni açıklıklar ve bu açıkların istismarından dolayı siber saldırı sayılarında artan bir oran söz konusu olduğu görülmektedir. Symantec tarafından yayınlanan rapor bu gerçeği ispatlar niteliktedir. Dijital ortamlarda saklanan bilginin her türlü saldırı ve tehditlere karşı korunması için güvenlik duvarları, antivirüs yazılımları, saldırı tespit sistemleri, saldırı engelleme sistemleri gibi araçlar geliştirilmiştir.

Bu tez çalışmasının bu bölümünde web tabanlı uygulamalar için, hem imza temelli hem de anormallik tespiti temelli yaklaşımları birlikte kullanan hibrit bir saldırı tespit yaklaşımı geliştirilmiştir. Kullanılan sistem gerçek zamanlı olarak web uygulamalarının güvenliğinin sağlanması için hem bilinen saldırıları yakalayacak şekilde hem de yeni olası saldırı örüntülerini keşfedecek şekilde tasarlanmıştır. Uygulama hem Windows hem Linux sunucularda çalışabilecek modüllere sahiptir.

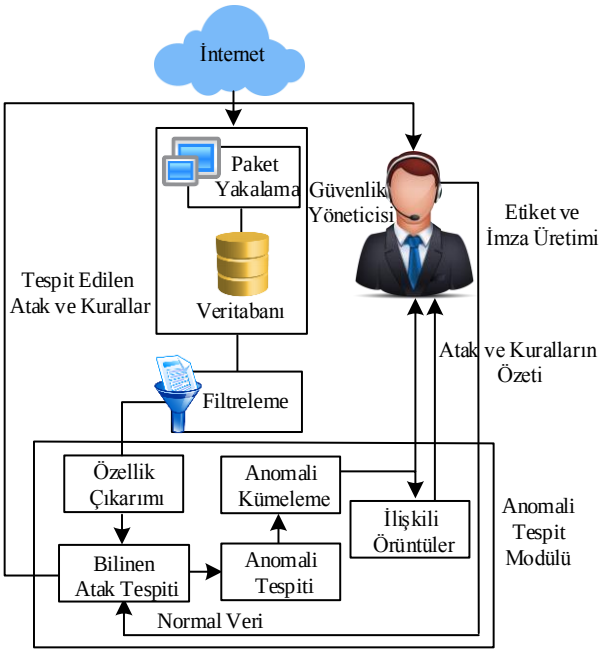
## 4.2 Saldırı Tespit Metodolojisi

Bu kısımda hem imza tabanlı hem de öğrenen akıllı sistem temelli iki saldırı tespit yaklaşımı da uygulanmıştır. Kötüye kullanım tespiti, imza temelli bir yaklaşım olup saldırgan davranışlarının birer saldırı imzası olarak modellenmesine dayanmaktadır. Kötüye kullanım tespiti, Bölüm 3.8.2'de anlatılmıştır. Genel çalışma mantığı, Şekil 4.1'deki çalışma diyagramı ile verilen bu yaklaşımda, saldırılara yönelik imza veri tabanları oluşturulduktan sonra, bu imzalarla eşleşen hareketler saldırı olarak tespit edilmektedir. Bu yaklaşımın avantajı imzası bilinen bütün kötücül aktivitelerin tespit edilebilmesidir. Dezavantajı ise, imzası bilinmeyen saldırıların tespit edilememesidir.




**Şekil 4.1** Kötüye kullanım tespiti

Anormallik tespiti yöntemi, öğrenen akıllı yöntemler içeren ve özellikle bilinmeyen saldırı örüntülerinin de yakalanmasını sağlamaya yönelik çalışmaları barındırır. Anormallik tespiti yaklaşımının detayları Bölüm 3.8.1'de verilmiştir. Şekil 4.2'de çalışma mantığı sunulan anormallik tespiti yaklaşımı, yakalanan ağ paketlerinin gerçek zamanlı analizler sonucunda kötücül aktivite olarak değerlendirilip değerlendirilemeyeceğinin tespit edilmesi süreci olarak tanımlanabilir. Bu yaklaşım, hem bilinen saldırıların hem de daha önceden bilinmeyen yeni tür saldırıların tespitinde etkin çözümler sunmaktadır. Bu yaklaşımın dezavantajı, esasen saldırı olmayan bazı aktivitelerin saldırı olarak tanımlanması oranının (false positive) yüksek olabilmesidir. Bu durumda yanlış alarmlar üretilir. Yanlış alarm üretiminin fazla olması aşırı bir yük getirebilmektedir. Yanlış alarm seviyesinin düşürülmesine yönelik olarak honeypot temelli yaklaşımlar önerilmektedir. Bölüm 4, özellikle web tabanlı saldırılara karşı, Bölüm 5 ise genel amaçlı olarak, yanlış alarm seviyesinin düşürülmesine yönelik honeypot temelli saldırı tespit yaklaşımları sunmaktadır.



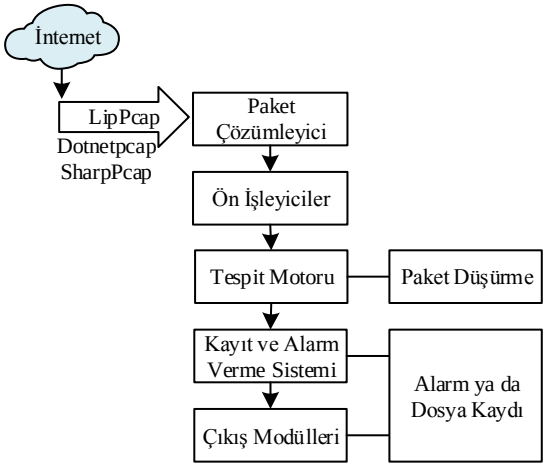

**Şekil 4.2** Anormallik tespiti

## 4.3 Güncel Web Saldırılarının Tespitine Yönelik Önerilen Hibrit Yaklaşım

Bu kısımda gerçek zamanlı veriler kullanılarak güncel web saldırılarının tespitine yönelik çeşitli uygulamalar geliştirilmiştir. Uygulamalar hem Linux temelli sunucularda hem de Windows temelli sunucularda çalışan tüm web uygulamalarında kullanılabilecek şekilde gerçekleştirilip test edilmiştir. Windows işletim sistemi üzerinde çalışan web uygulama sunucuları için gerçekleştirilen çalışmalar, C#-ASP.NET programlama dilleri kullanılarak geliştirilmiştir. Linux işletim sistemli sunucular için ise PHPIDS temelli web uygulamaları gerçekleştirilmiştir. Uygulamalar, opsiyonel olarak kullanıcı tarafından seçilebilecek bir ağ adaptörü üzerinden gelen-giden ağ trafiğini ve HTTP protokolü üzerinden yapılan tüm web isteklerini izleyebilmektedir.

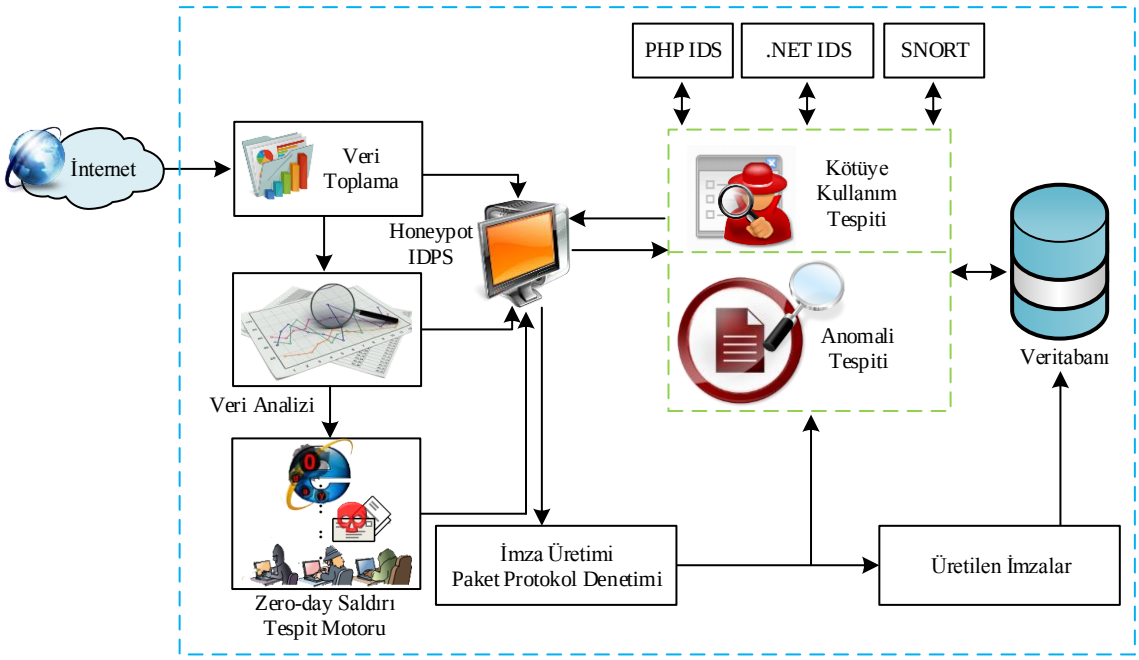
Temel olarak TCP/IP protokolünü kullanan Windows temelli uygulama, C# dili ile *SharpPcap* ve *PacketDotNet ActiveX* nesneleri kullanılarak gerçekleştirilmiştir. Şekil 4.3, önerilen sistemin fonksiyonel yapısını göstermektedir. Bu yapı, SNORT saldırı tespit sisteminin mimarisi ile örtüşmektedir. Temel olarak sistem üzerinde, *paket yakalama, paket çözümleme, ön işleyiciler, tespit modülü* ve *çıkış modülü* bulunmaktadır. Linux sunucu temelli uygulama ise, PHP dili ile gerçekleştirilmiştir. Ağ trafiğindeki IP paketleri okunup TCP paketi içerisindeki kullanıcı istekleri ve gerçek zamanlı olarak web sunucusuna gelen istekler yorumlanarak kural tabanlı bir uzman sistem yardımıyla web saldırılarının tespiti yapılmaktadır. Geliştirilen yazılımlar, kurumsal bir ağ trafiğinin izlenmesi, denetlenmesi ve ağdaki kötücül aktivitelerin önüne geçilebilmesi amacıyla kodlanan honeypot temelli bir

saldırı tespit ve engelleme sistemi ile birleştirilmiştir. Geliştirilen ve LAN bölgesinde konumlandırılmış honeypot sunuculara gelen paketlerin yakalanıp STS uygulamasındaki anormallik tespiti ve kötüye kullanım tespiti yöntemleriyle trafik analizinin yapılması sağlanmıştır.



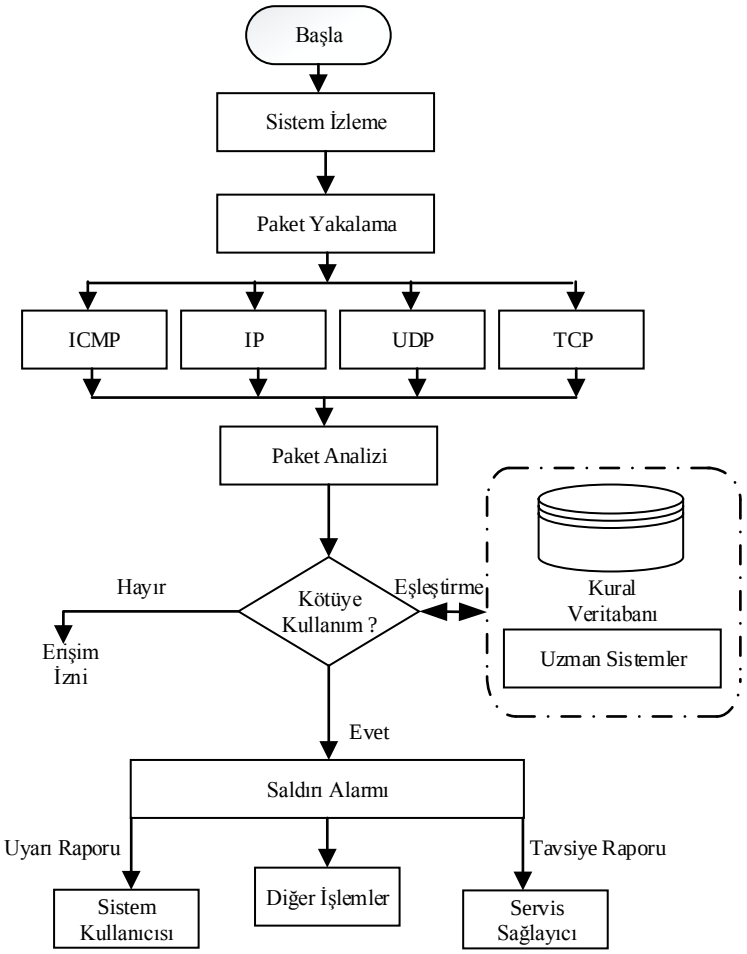

**Şekil 4.3** Önerilen sistemin fonksiyonel yapısı

Honeypotlar, ağ üzerinde güvenlik amacı ile kullanılan sistemlerden biridir. Tek başlarına özel bir amaç için kullanılmayan honeypot sistemler; STS, güvenlik duvarı vb. güvenlik araçlarıyla birlikte ağ güvenliğinin sağlanması için kullanılırlar. Saldırganlar, honeypotun gerçek bir bilgisayar olduğu algısına kapıldığı zaman hedef sistem üzerinde yapacağı araştırmalarda, mevcut açıklıkların olduğunu görür. Bu zafiyet noktaları saldırganları tuzak sisteme düşürmek amacıyla oluşturulan çeşitli konfigürasyonlardır. Honeypotlarda temel amaç, kaba tabirle saldırganı bal peteğine düşürmektir [15, 16]. Böylece ağda faaliyet gösteren gerçek sistemler de kamufle edilmiş olur. Honeypotların taklit ettikleri servislerde, gerçek bir bilgi bulundurulmaz. Bu nedenle ağda oluşturulan bu tuzak IP adreslerine saldırgan olmayan sıradan bir kullanıcının erişmesi beklenen bir durum değildir. Bu şekilde honeypotun ağ üzerinde simüle ettiği bilgisayarların IP adreslerine gelen tüm istekler saldırı olarak nitelendirilebilir. Daha sonra yeni saldırı yöntemlerini (zero-day) tespit etmek için honeypotlardan saldırı niteliğindeki paket bilgileri toplanır. Bu bilgiler log dosyaları şeklinde analiz edilmek için kaydedilir. Şekil 4.4, geliştirilen sistemin genel bir mimarisini sunmaktadır. Honeypot sistemler Bölüm 5'te detaylıca incelenmiştir.



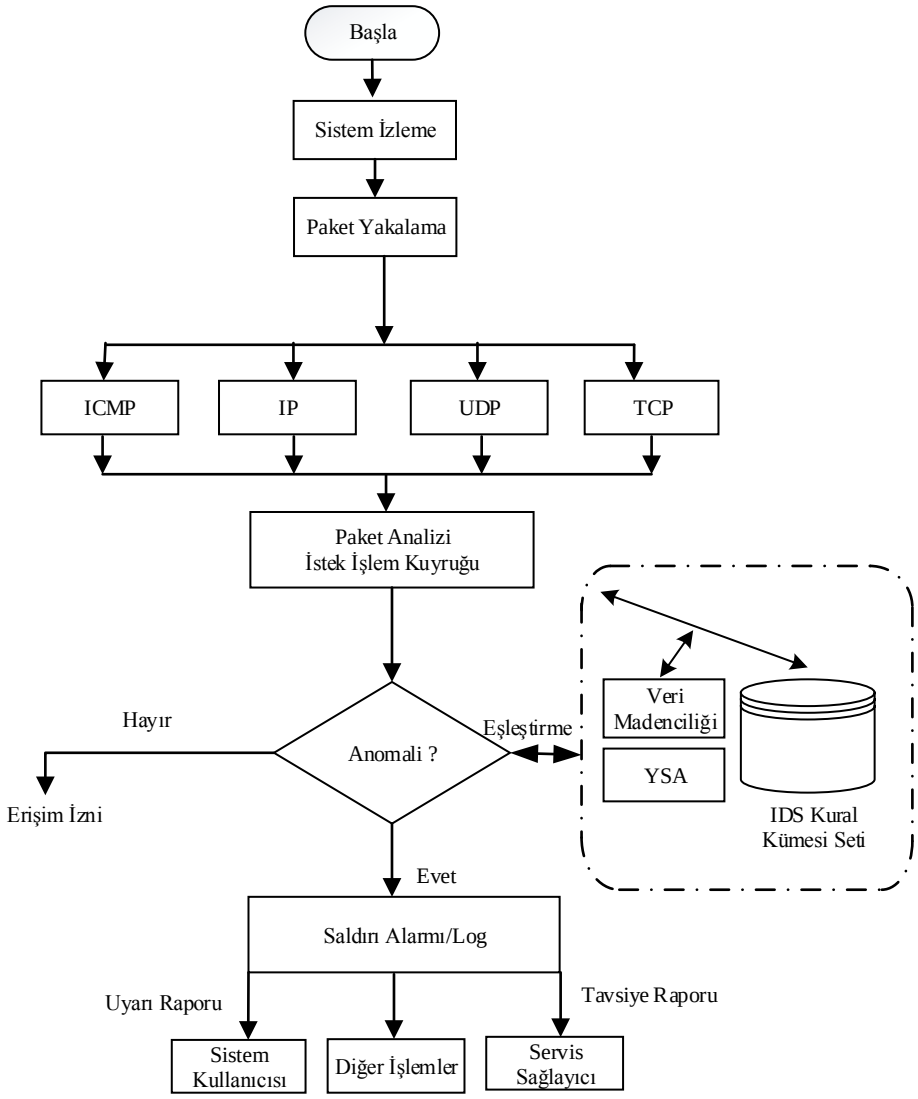

**Şekil 4.4** Geliştirilen sistemin genel yapısı

Sistem merkezinde yer alan Honeypot-IDPS yapısındaki düşük etkileşimli LAN bölgesi honeypotu, içerisinde saldırı çekme bileşeni, veri toplama ve veri analizi bileşenlerini bulundurmaktadır. Sistemde ayrıca, honeypota düşürülmüş saldırganın aktivitelerinin takibi ve analizinin yapıldığı sıfır gün saldırı tespit modülü bulunmaktadır. Bu modül sayesinde yeni saldırı örüntülerinin tespit edilebilmesi ile yanlış alarm seviyesi düşürülmektedir. Elde edilen yeni saldırı imzaları veri tabanlarında sonraki analizler için kayıt altına alınmaktadır. Geliştirilen uygulamalarda saldırı tespit ve engelleme sistemlerinin sınıflandırılmalarında kullanılan temel kriter olan *saldırı tespitinde kullanılan yöntem* açısından mevcut olan iki temel yöntem de kodlanmıştır. Bu yöntemlerden ilki Şekil 4.5'te akış şeması verilen *kötüye kullanım tespiti* yöntemidir. İkincisi ise, Şekil 4.6'da akış şeması verilen *anormallik tespiti* yöntemidir.




**Şekil 4.5** Kötüye kullanım tespiti için genel akış diyagramı

Geliştirilen sistemde, Şekil 4.5'te gösterildiği gibi, belirlenen ağ dinleme katmanına göre seçilen parametreler ve dinlenilmesi istenen paket tipleri ayarlanabilmektedir. TCP, UDP, ICMP, IP ve ARP gibi katman 2 ve katman 3 protokolleri ve istenilen portlar dinlenebilmektedir. Burada izleme işlemi için SNORT'un paket izleyici modu kullanılmıştır. Yakalanan paketler, geliştirilen STS paket analizi modülünde analiz edilmekte ve kural veri tabanındaki saldırı imzaları ile karşılaştırılmaktadır. Herhangi bir saldırı örüntüsü ile eşleşen bir paket içeriği yakalanması durumunda, *saldırı alarmı* üretilerek sistem yöneticisi uyarılmakta veya belirlenen işlemler yapılmaktadır. Bu yapıda sadece kural veri tabanlarında bulunan saldırılar yakalanabilmektedir. Olası yeni tür saldırıların tespit edilebilmesi için anormallik tespiti yaklaşımı kullanılmıştır.




**Şekil 4.6** Anormallik tespiti için genel akış diyagramı

Şekil 4.6'da verilen akış şemasından da anlaşılacağı üzere anormallik tespiti yaklaşımını kullanan STS'ler öğrenen, akıllı sistemlerdir. Bu yapıda YSA gibi çeşitli akıllı yöntemler kullanılmaktadır. Gerçekleştirdiğimiz sistemde, içerisinde gerçek veri barındırmayan, düşük etkileşimli *honeyd* uygulamasını baz alan bir honeypot, LAN bölgesinde konumlandırılarak anormallik tespiti yaklaşımı test edilerek uygulanmıştır. Honeypot sistemler, gerçek sistemler üzerinde çalışan servisleri taklit ettikleri için saldırganların dikkatini çekebilecek özellikte konfigüre edilirler. Böylece honeypot sistemlere gelen trafiğin normal bir trafik olması beklenmez. Buradaki trafik, ağın normal trafiği ile karşılaştırılarak sınıflandırma işlemine tabi tutulur. Böylece yeni tür saldırıların da tespit edilebileceği bir yapı oluşturulmuştur. SNORT vb. açık kaynak kod STS yapılarında, esasen saldırı olmadığı halde saldırı olarak nitelenen paket sayısı oldukça fazladır. Bu durum, kural kümesinin iyi yapılandırılmadığı ağlarda oldukça fazla sayıda yanlış alarm üretimi demektir ve istenmeyen

bir durumdur. Literatürde *yanlış pozitif* olarak tanımlanan bu durum, geliştirilen Honeypot-IDPS yapısı ile azaltılmıştır. Ayrıca olası yeni saldırı örüntüleri de honeypota düşen saldırganların davranışlarının analizi ile yakalanabilmektedir.

## 4.4 Uygulama Sonuçları

Gerçekleştirilen uygulamalar, gerçek zamanlı olarak güncel web saldırılarını tespit edebilen, kural tabanlı ve anormallik temelli bir hibrit saldırı tespit ve engelleme sistemi örneğidir. Honeypot sistemlerin avantajları ile birleştirilen hibrit sistem tarafından tespit edilebilen saldırılar, OWASP tarafından belirlenen ve en çok karşılaşılan web açıklıkları arasında gösterilen, XSS, CSRF, Ldap Injection, Xpath Injection, Header Injection, Directory Traversal, RFI/LFI, DoS/DDoS, SQL Injection vb. saldırılardır. Ayrıca DoS/DDoS gibi önlenmesi güç veya yeni üretilmiş saldırılarda da gerçek zamanlı anormal durum analizi ile tespit yapılabilmektedir. Bu tespit yöntemlerini içeren akış şemaları, Şekil 4.5 ve Şekil 4.6'da verilmiştir. Bu kısımda web tabanlı güncel açıklar ve bu açıklıkların test edilmesi ile elde edilen uygulama sonuçları verilmektedir.

### 4.4.1 SQL Enjeksiyonu

Günümüzde profesyonel olarak hizmet sunan web uygulamalarının neredeyse tamamı veri tabanlarını kullanmaktadır. Online çalışan bu web uygulamaları, veri tabanı ile yapısal bir sorgulama dili olan SQL aracılığıyla haberleşirler. SQL Enjeksiyon ise, web uygulamalarından alınan kullanıcı girdileri ile oluşturulan SQL sorgularının manipülasyonu olarak tanımlanabilir [85]. Kullanıcıların etkileşimde bulundukları veri tabanlı web uygulamalarında, sorgu veya parametreler kullanılarak veri tabanı tablolarındaki bilgiler belli şartlara göre filtrelenerek uygulama ara yüzüne aktarılır. Aktarılan bu sonuç değerleri, uygulamanın tasarımına göre kullanıcıya veya yöneticiye belli formatlarda sunulur. SQL enjeksiyonu yöntemi tam bu işlemler gerçekleştirilirken yapılır. Saldırgan, web tarayıcı adres çubuğuna veya uygulamada bulunan giriş kontrollerine kötücül kodlar ekleyerek, SQL enjeksiyon saldırısını gerçekleştirir. Genel kullanıma açık olmayan ancak bu şekilde elde edilen bilgiler önemli ve gizli olabilir.

Saldırgan, sistem ve veri tabanı hakkında elde ettiği bu önemli bilgilerle SQL enjeksiyon senaryosuna farklı boyutlar kazandırarak, veri tabanında bulunan diğer bilgilere ulaşabilir. Sonrasında elde ettiği bilgileri kullanarak, hedefini gerçekleştirir [85]. Web uygulamalarında kullanılabilecek bir SQL Enjeksiyon saldırı örneği aşağıda verilmiştir. Şekil 4.7'de bir web uygulamasında kullanılan örnek bir yönetici formu gösterilmektedir.


**Şekil 4.7** Örnek bir giriş formu

Yöneticinin bu form üzerinden doğru bir şekilde giriş yapması sonucunda oluşan SQL sorgusu;

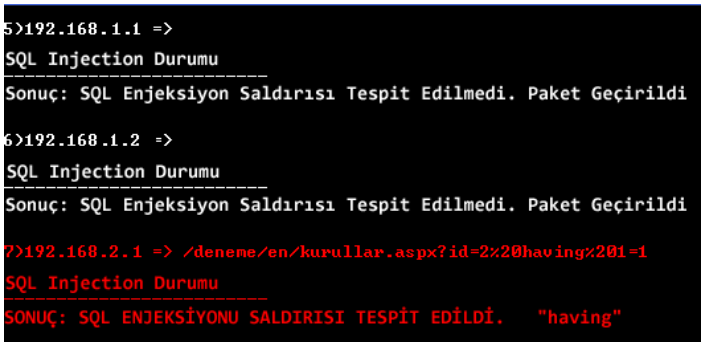
**SELECT * FROM Users WHERE Username = 'admin' AND Password = '1234'**

şeklinde olmaktadır. Saldırgan bu vb. gibi giriş formlarında kullanıcı adı ve şifre alanlarına Tablo 4.1'de verilen ve SQL enjeksiyonu için kullanılabilecek ' OR 1=1-- ifadesi ile saldırı girişiminde bulunabilmektedir. Bu saldırı girişimi sonucunda oluşan SQL sorgusu;

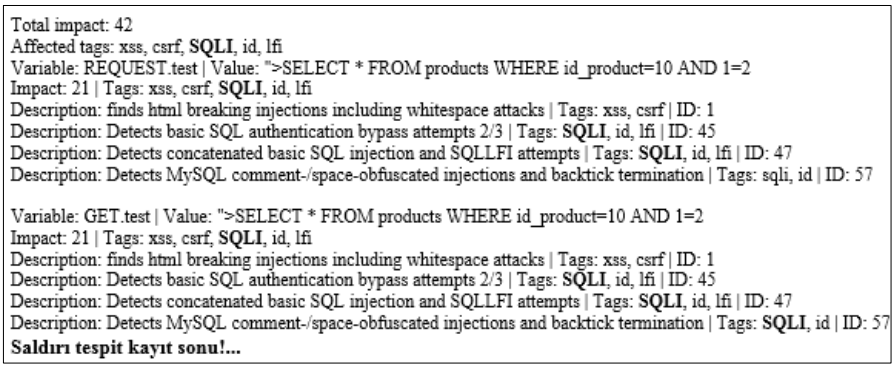
**SELECT * FROM Users WHERE Username = 'admin' AND Password = '' OR 1=1--**

şeklinde olmaktadır. Oluşan bu sorgu ifadesine dikkat edilirse, saldırgan başarılı bir biçimde sisteme erişim hakkı kazanmış olacaktır.

Tablo 4.1'de, SQL enjeksiyon saldırısında kullanılabilecek sözcükler listelenmiştir. Bu sözcüklerin her biri virgülle ayrılmıştır. Bu sözcükler, uygulama içerisinde seçilen ağ arabiriminden gelen ağ paketlerinin içeriğindeki isteklerle karşılaştırılmaktadır. Karşılaştırma sonucunda eşleşen kayıtlar, saldırı olarak algılanmaktadır. Bu saldırı kayıtları konsol ekranında kırmızı yazı rengi ile listelenmektedir. Şekil 4.8'de, 7 numaralı paket "192.168.2.3" istemci adresinden gönderilmiştir. İstek adresine dikkat edildiğinde "Having" SQL sözcüğü ile SQL enjeksiyonu saldırı girişimi yapıldığı anlaşılmaktadır. Önerilen yöntem, önceki bölümlerde Şekil 4.5 ve Şekil 4.6'da verilen akış şemalarında görüldüğü gibi bir alarm mekanizmasına sahiptir. Bu alarm mekanizması STS'lerin kullanıldığı sisteme göre farklılık gösterebilir. Şekil 4.8'de geliştirilen uygulamaların, Asp.Net temelli web uygulamalarında çalışması ve örnek bir SQL Enjeksiyon saldırısının yakalanması gösterilmektedir. Bu uygulamamızda tespit edilen saldırı için konsol ortamında kırmızı renkli yazı ve sesli uyarı kullanılmaktadır. Şekil 4.9 ise Wordpress temelli olarak gerçekleştirilen bir PHP uygulamasında, PHPIDS saldırı tespit sistemi yapısı ile birlikte çalışan uygulamamızın kullanıldığı sistemde SQL enjeksiyonu saldırısının tespitini göstermektedir.




**Şekil 4.8** SQL enjeksiyonu tespiti




**Şekil 4.9** PHPIDS temelli SQL enjeksiyonu saldırısı tespiti

**Tablo 4.1** SQL enjeksiyonu saldırılarında kullanılabilecek anahtar kelimeler

| **SQL Enjeksiyon** |
|---|
| ', \, --, ' --, --';, ' ;, = ', = ;, = --, \x23, \x27, \x3D \x3B', \x3D \x27, \x27\x4F\x52 SELECT *, \x27\x6F\x72 SELECT *, 'or select *, admin'--, ';shutdown--, <>\'%;)(&+, ' or ''=', ' or 'x'='x, \ or \x\=\x, ') or ('x'='x, 0 or 1=1, ' or 0=0 --, \ or 0=0 --, or 0=0 --, ' or 0=0 #, \ or 0=0 #, or 0=0 #, ' or 1=1--, \ or 1=1--, ' or '1'='1'--, \' or 1 --'\, or 1=1--, or%201=1, or%201=1 --, ' or 1=1 or ''=', \ or 1=1 or \\=\, ' or a=a--, \ or \a\=\a, ') or ('a'='a, \) or (\a\=\a, hi\ or \a\=\a, hi\ or 1=1 --, hi' or 1=1 --, hi' or 'a'='a, hi') or ('a'='a, hi\) or (\a\=\a, 'hi' or 'x'='x';, @variable, ,@variable, PRINT, PRINT @@variable, select, insert, procedure, limit, order by, asc, desc, delete, update, distinct, having, truncate, replace, like, handler, bfilename, ' or username like '%, ' or uname like '%, exec xp, exec sp, '; exec master..xp_cmdshell, '; exec xp_regread, t'exec master..xp_cmdshell 'nslookup www.google.com'--, --sp_password, \x27UNION SELECT, ' UNION SELECT, ' UNION ALL SELECT, ' or (EXISTS), ' (select top 1, '||UTL_HTTP.REQUEST, 1;SELECT%20*, to_timestamp_tz, tz_offset, <>"'%;)(&+, '%20or%201=1, %27%20or%201=1, %20$(sleep%2050), %20'sleep%2050', char%4039%41%2b%40SELECT, '%20OR, 'sqlattempt1, (sqlattempt2), \|, %7C, *\|, %2A%7C, *(\|(mail=*)), %2A%28%7C%28mail%3D%2A%29%29, *(\|(objectclass=*)), %2A%28%7C%28objectclass%3D%2A%29%29, %28, %29, %26, !, %21, ' or 1=1 or ''=', ' or ''=', x' or 1=1 or 'x'='y, //, //*, */*, @*, SELECT @@VERSION, SELECT * from v$version;, union, delete, alter, having |

Tablo 4.2, okunan örnek bir ağ paketi içeriğini göstermektedir. Tablodan görüldüğü üzere gerçek zamanlı olarak elde edilen ağ paketlerinin analiz edilmesi sonucu, Şekil 4.5 ve Şekil 4.6'da belirtilen akış şemalarındaki yöntemlerle gerçek zamanlı durum analizleri sonucu saldırılar tespit edilir.

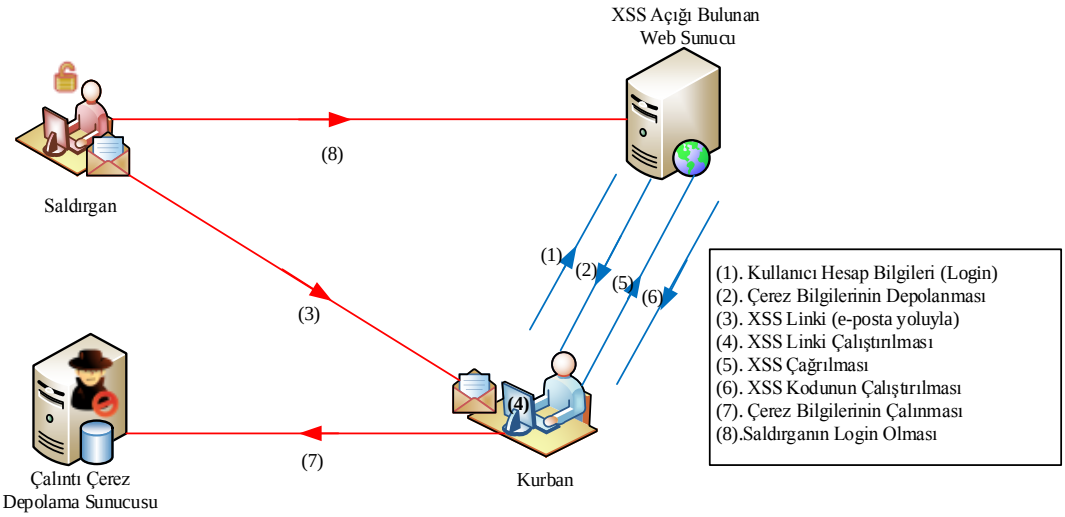
**Tablo 4.2** Okunan örnek bir paket içeriği

| **Okunan Paket** |
|---|
| **GET** /deneme/en/DuzenlemeKurulu.aspx HTTP/1.1<br>**Host:** 192.168.2.2<br>**Connection:** keep-alive<br>**Accept:**text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8<br>**User-Agent:** Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/31.0.1650.63 Safari/537.36<br>**Referer:** http://192.168.2.2/deneme/en/Kurullar.aspx<br>**Accept-Encoding:** gzip,deflate,sdch<br>**Accept-Language:** tr-TR,tr;q=0.8,en-US;q=0.6,en;q=0.4<br>HTTP/1.1 200 OK<br>**Content-Type:** text/html<br>**Content-Encoding:** gzip<br>**Last-Modified:** Mon, 06 Jan 2015 19:43:50 GMT<br>**Accept-Ranges:** bytes<br>**ETag:** "1f1ea59f17bcf1:0"<br>**Vary:** Accept-Encoding<br>**Server:** Microsoft-IIS/8.0<br>**X-Powered-By:** ASP.NET Date: Tue, 14 Jan 2014 23:03:57 GMT Content-Length: 3523 |

### 4.4.2 Çapraz Site Betikleme

Çapraz site betikleme (XSS-Cross Site Scripting) saldırıları, web siteleri arasında kod yazma şeklinde gerçekleştirilen saldırılardır. Bu saldırı, sitedeki kontrol yapısı içermeyen alanlarda betik kodların çalıştırılmasıyla gerçekleştirilir. Bu betik, HTML veya Javascript kodlarından oluşabilir. Genel olarak Javascript kodlarıyla gerçekleştirilir. Ancak web tarayıcılar tarafından desteklenen VBScript, Ajax, ActiveX benzeri diller tarafından yazılan betiklerle de bu saldırılar yapılabilmektedir. Betik kodlar, HTML etiketleri arasında çalıştırıldığından XSS yerine HTML Enjeksiyonu ifadesi de kullanılmıştır [86].

XSS saldırısı ile web siteleri oturum çerezleri ele geçirilebilir, web tarayıcısı istenen adrese yönlendirilebilir, DoS saldırıları yapılabilir. XSS saldırısı, Şekil 4.10'da örnek bir senaryo ile gösterilmektedir. Burada kurban, üzerinde XSS açığı bulunan web sitesine normal giriş yaptığında, oturum bilgileri normal olarak makinesine yüklenecektir. 3 nolu adımda görüldüğü gibi saldırgan, bir XSS betiğinin linkini e-mail yoluyla yollamaktadır. Kurban, XSS betiğini çalıştırdığı anda sunucuyla etkileşime geçilip kod çalıştırılacak ve kurbanın oturum bilgileri çalınabilecektir. Şekil 4.11 ise XSS saldırısının tespit edilmesini göstermektedir. Görüldüğü gibi XSS etkilenen tag'ı ve ilgili XSS betiği (<script>…</script>) yakalanmış ve tespit sonuç ekranında gösterilmiştir.




**Şekil 4.10** XSS saldırı senaryosu

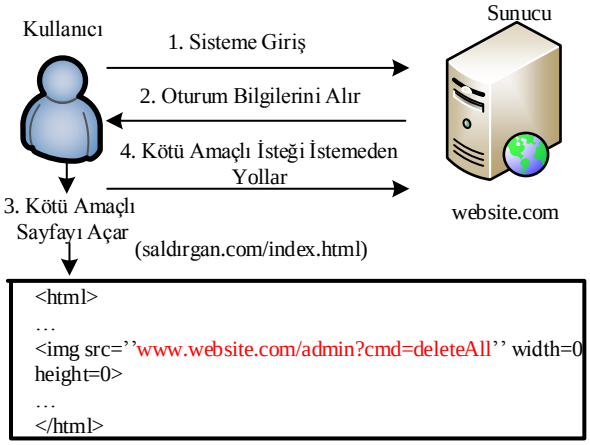
```
Total impact: 108
Affected tags: XSS, csrf, id, rfe, lfi, sqli
Variable: REQUEST.test | Value: "><script>document.write('<iframe src="http://php.net?cookie=' document.cookie.escape()
'" height=0 width=0 />');</script>
Impact: 54 | Tags: XSS, csrf, id, rfe, lfi, sqli
Description: finds html breaking injections including whitespace attacks | Tags: XSS, csrf | ID: 1
Description: finds attribute breaking injections including whitespace attacks | Tags: XSS, csrf | ID: 2
Description: Detects possible includes and typical script methods | Tags: XSS, csrf, id, rfe | ID: 16
Description: Detects JavaScript object properties and methods | Tags: XSS, csrf, id, rfe | ID: 17
Description: Detects JavaScript string properties and methods | Tags: XSS, csrf, id, rfe | ID: 19
Description: Detects very basic XSS probings | Tags: XSS, csrf, id, rfe | ID: 21
Description: Detects JavaScript location/document property access and window access obfuscation | Tags: XSS, csrf | ID: 23
Description: Detects obfuscated script tags and XML wrapped HTML | Tags: XSS | ID: 33
Description: Detects possibly malicious html elements including some attributes | Tags: XSS, csrf, id, rfe, lfi | ID: 38
Description: Detects classic SQL injection probings 1/2 | Tags: sqli, id, lfi | ID: 42
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, lfi | ID: 45
Description: Finds attribute breaking injections including obfuscated attributes | Tags: XSS, csrf | ID: 68

Variable: GET.test | Value: "><script>document.write('<iframe src="http://php.net?cookie=' document.cookie.escape() '"
height=0 width=0 />');</script>
Impact: 54 | Tags: XSS, csrf, id, rfe, lfi, sqli
Description: finds html breaking injections including whitespace attacks | Tags: XSS, csrf | ID: 1
Description: finds attribute breaking injections including whitespace attacks | Tags: XSS, csrf | ID: 2
Description: Detects possible includes and typical script methods | Tags: XSS, csrf, id, rfe | ID: 16
Description: Detects JavaScript object properties and methods | Tags: XSS, csrf, id, rfe | ID: 17
Description: Detects JavaScript string properties and methods | Tags: XSS, csrf, id, rfe | ID: 19
Description: Detects very basic XSS probings | Tags: XSS, csrf, id, rfe | ID: 21
Description: Detects JavaScript location/document property access and window access obfuscation | Tags: XSS, csrf | ID: 23
Description: Detects obfuscated script tags and XML wrapped HTML | Tags: XSS | ID: 33
Description: Detects possibly malicious html elements including some attributes | Tags: XSS, csrf, id, rfe, lfi | ID: 38
Description: Detects classic SQL injection probings 1/2 | Tags: sqli, id, lfi | ID: 42
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, lfi | ID: 45
Description: Finds attribute breaking injections including obfuscated attributes | Tags: XSS, csrf | ID: 68
Saldırı tespit kayıt sonu!...
```

**Şekil 4.11** XSS saldırı tespiti

### 4.4.3 Siteler Arası İstek Sahteciliği

Bu saldırı (CSRF- Cross Site Request Forgery), siteler arası istek dolandırıcılığı/sahteciliği anlamında olup uyuyan dev olarak adlandırılır ve XSS saldırılarına benzer bir mantıktadır. CSRF, herhangi bir İnternet kullanıcısının kullandığı uygulamada kendi isteği dışında işlemler yapmasının sağlanmasıyla gerçekleştirilen bir saldırı türüdür. İnternet banka hesaplarından para gönderimi ve e-posta gönderme gibi işlemler, CSRF ile yapılabilecek saldırılar arasında yer alır. CSRF saldırısını örnek bir senaryo üzerinden göstermek için Şekil 4.12'deki senaryo çizilmiştir. Örneğin kullanıcı, üyesi olduğu "*website.com*" adresli siteye

giriş yaptığında (1. adım), web sitesinden oturum bilgilerini alır (2. adım), giriş yapılıp oturum bilgileri alındıktan sonra tamamen farklı bir domain olan "*saldırgan.com/index.html"* üzerindeki saldırı linkine tıkladığında (3. adım) farkında bile olmadan website.com üzerindeki kayıt bilgileri değiştirilebilir. Bu saldırı türü ile web uygulamasının türüne göre farklı işlemler gerçekleştirilebilir. Örneğin bir alışveriş sitesi ise, alışveriş sepetine yeni ürünler eklenebilir. Ayrıca kullanıcı 3. adımda belirtilen siteyi normal sıradan bir site gibi dolaşmaktadır, saldırı linki içerdiğini bilmemektedir [87]. Şekil 4.13, CSRF saldırısı test ve tespit sonuçlarını göstermektedir. Buradaki domain isimleri örnek olarak verilmiştir.




**Şekil 4.12** CSRF saldırı senaryosu

Total impact: 72
Affected tags: xss, **CSRF**, id, rfe, lfi, sqli

Variable: REQUEST.test | Value: "><SCRIPT>alert("Cookie" document.cookie)</SCRIPT>
Impact: 36 | Tags: xss, **CSRF**, id, rfe, lfi, sqli
Description: finds html breaking injections including whitespace attacks | Tags: xss, **CSRF** | ID: 1
Description: Detects possible includes and typical script methods | Tags: xss, **CSRF**, id, rfe | ID: 16
Description: Detects JavaScript object properties and methods | Tags: xss, **CSRF**, id, rfe | ID: 17
Description: Detects very basic XSS probings | Tags: xss, **CSRF**, id, rfe | ID: 21
Description: Detects JavaScript location/document property access and window access obfuscation | Tags: xss, **CSRF**| ID: 23
Description: Detects obfuscated script tags and XML wrapped HTML | Tags: xss | ID: 33
Description: Detects possibly malicious html elements including some attributes | Tags: xss, **CSRF**, id, rfe, lfi | ID: 38
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, lfi | ID: 45

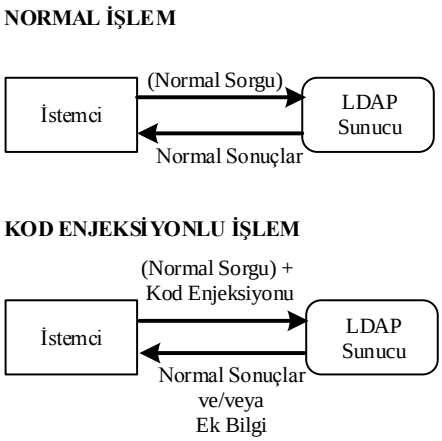
Variable: GET.test | Value: "><SCRIPT>alert("Cookie" document.cookie)</SCRIPT>
Impact: 36 | Tags: xss, **CSRF**, id, rfe, lfi, sqli
Description: finds html breaking injections including whitespace attacks | Tags: xss, **CSRF** | ID: 1
Description: Detects possible includes and typical script methods | Tags: xss, **CSRF**, id, rfe | ID: 16
Description: Detects JavaScript object properties and methods | Tags: xss, **CSRF**, id, rfe | ID: 17
Description: Detects very basic XSS probings | Tags: xss, **CSRF**, id, rfe | ID: 21
Description: Detects JavaScript location/document property access and window access obfuscation | Tags: xss, **CSRF** | ID:23
Description: Detects obfuscated script tags and XML wrapped HTML | Tags: xss | ID: 33
Description: Detects possibly malicious html elements including some attributes | Tags: xss, **CSRF**, id, rfe, lfi | ID: 38
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, lfi | ID: 45
**Saldırı tespit kayıt sonu!...**

**Şekil 4.13** CSRF saldırı tespiti

### 4.4.4 Basit Dizin Erişim Protokolü Enjeksiyonu

LDAP (Lightweight Directory Access Protocol), dizin yönetimi için kullanılan bir protokoldür. Bir çeşit dizin servisi standardıdır. LDAP enjeksiyonu, verilerin LDAP cümlecikleri şeklinde saklandığı web uygulamalarına yönelik yapılan saldırıları ifade eder.

Örnek bir LDAP atağı Şekil 4.14'te verilmiştir. Şekilde de görüldüğü gibi *istemci* ve *LDAP sunucu* arasında gerçekleşen sorgu-cevap trafiğinde sorguların içerisine eklenen çeşitli parametreler ile hedef sistem üzerindeki veriler ele geçirilebilecektir [88].




**Şekil 4.14** LDAP saldırısı

### 4.4.5 Başlık Enjeksiyonu

Başlık enjeksiyonu saldırısı, HTTP Header bilgilerinin manipüle edilmesiyle ortaya çıkan açıklardan faydalanmaktır. Özellikle PHP dili, URL aracılığı ile rastgele başlık enjeksiyonu yapılabilmesine izin veren zafiyetler içermektedir. Gelişigüzel başlıklar eklenerek, saldırganlar CSRF (siteler arası istek sahteciliği), XSS (siteler arası betik yazma), HTML-enjeksiyonu ve başka saldırıları gerçekleştirebilir.

### 4.4.6 Uzaktan/Yerelden Dosya Dahil Etme

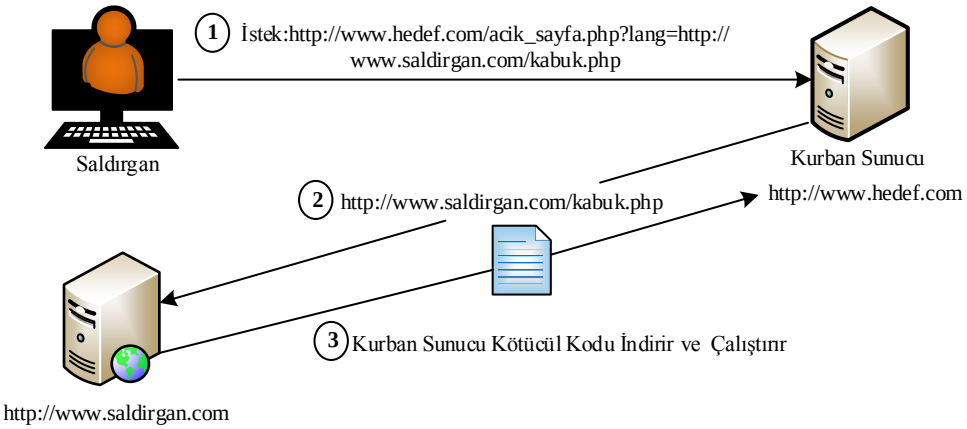
Uzaktan/Yerelden dosya dahil etme (RFI/LFI(Remote File Inclusion / Local File Inclusion)) saldırıları, dosya ekleme (file inclusion) olarak adlandırılabilecek zafiyet türüdür. Bu saldırılar, genellikle PHP dili ile yazılan web uygulamalarında karşılaşılan açıklıklardır. Burada saldırgan, yerel (local) ve uzaktan (remote) kod çalıştırabilmektedir. LFI, yerelden dosya ekleyerek kod çalıştırma ve RFI ise uzaktan dosya ekleyerek kod çalıştırma anlamına gelmektedir.

RFI/LFI saldırıları, PHP dili ile kodlanmış web uygulamalarında, tanımlanan değişkenlere değer atanmaması veya atanan değerlerin filtrelenmemesinden kaynaklanmaktadır. Bu durumun oluşumuna bir örnek Tablo 4.3'te verilmiştir.

**Tablo 4.3** PHP'de değerlerin filtrelenmesi

```
<?php
$sayfa = $_GET[sayfa];
include($sayfa);
?>
```

Tablo 4.3'te verilen program parçacığı ile yapılmak istenen *"index.php?page=index.php"* dir. Ama burada herhangi bir kontrol gerçekleştirilmeden *sayfa* adlı değişken ile GET metodu üzerinden bir yapı geliştirilmiştir. Buradaki kodlama dikkatsizliğini farkeden yazılımcı, *"index.php?page=zararli.php"* şeklinde bir Shell dosyası çalıştırarak server üzerinde gezinip dosyaları düzenlemeye kadar ilerleyebilir. Şekil 4.15'te örnek bir RFI/LFI, saldırı senaryosu görülmektedir.




**Şekil 4.15** RFI/LFI saldırı senaryosu

Şekil 4.16, RFI/LFI saldırılarının tespit edilmesini göstermektedir. Yakalanan istek paketinin testinden de anlaşılacağı üzere, üzerinde kötücül amaçlı kod barındıran saldırı sitesinden kodların indirilip çalıştırılmasına yönelik bir sayfa tespit edilmiştir.

```
Total impact: 32
Affected tags: xss, csrf, sqli, id, RFI/LFI

Variable: REQUEST.test | Value: ">http://vulnerable_host/vuln_page.php?file=http://attacker_site/malicous_page
Impact: 16 | Tags: xss, csrf, sqli, id, RFI/LFI
Description: finds html breaking injections including whitespace attacks | Tags: xss, csrf | ID: 1
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, LFI | ID: 45
Description: Detects url injections and RFE attempts | Tags: id, RFI/LFI | ID: 61

Variable: GET.test | Value: ">http://vulnerable_host/vuln_page.php?file=http://attacker_site/malicous_page
Impact: 16 | Tags: xss, csrf, sqli, id, RFI/LFI
Description: finds html breaking injections including whitespace attacks | Tags: xss, csrf | ID: 1
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, LFI | ID: 45
Description: Detects url injections and RFE attempts | Tags: id, RFI/LFI | ID: 61
Saldırı tespit kayıt sonu!...
```

**Şekil 4.16** RFI/LFI saldırısı tespiti

Bölüm 3.7.3'te incelenen hizmet aksattırma saldırılarının tespit sonucu Şekil 4.17'de gösterilmektedir. Burada temel bir çapraz site betikleme saldırısı üzerinden DoS atağı girişiminin tespit edildiği görülmektedir.

Total impact: 82
Affected tags: xss, csrf, id, rfe, sqli, lfi, **DOS**

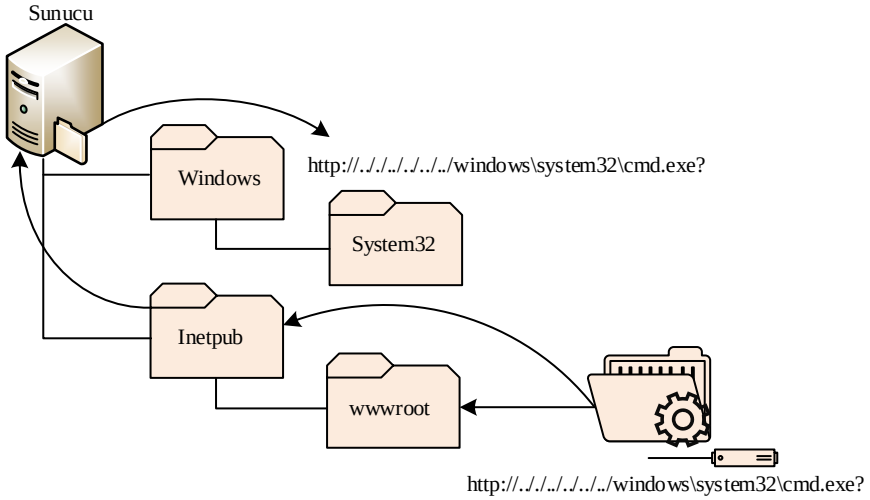
Variable: REQUEST.test | Value: "><b onmouseover=alert('Wuffff!')>click me!</b>
Impact: 41 | Tags: xss, csrf, id, rfe, sqli, lfi, **DOS**
Description: finds html breaking injections including whitespace attacks | Tags: xss, csrf | ID: 1
Description: Detects possible includes and typical script methods | Tags: xss, csrf, id, rfe | ID: 16
Description: Detects very basic XSS probings | Tags: xss, csrf, id, rfe | ID: 21
Description: Detects obfuscated script tags and XML wrapped HTML | Tags: xss | ID: 33
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, lfi | ID: 45
Description: Detects basic SQL authentication bypass attempts 3/3 | Tags: sqli, id, lfi | ID: 46
Description: Detects basic XSS DoS attempts | Tags: rfe, **DOS** | ID: 65
Description: finds malicious attribute injection attempts and MHTML attacks | Tags: xss, csrf | ID: 71

Variable: GET.test | Value: "><b onmouseover=alert('Wuffff!')>click me!</b>
Impact: 41 | Tags: xss, csrf, id, rfe, sqli, lfi, **DOS**
Description: finds html breaking injections including whitespace attacks | Tags: xss, csrf | ID: 1
Description: Detects possible includes and typical script methods | Tags: xss, csrf, id, rfe | ID: 16
Description: Detects very basic XSS probings | Tags: xss, csrf, id, rfe | ID: 21
Description: Detects obfuscated script tags and XML wrapped HTML | Tags: xss | ID: 33
Description: Detects basic SQL authentication bypass attempts 2/3 | Tags: sqli, id, lfi | ID: 45
Description: Detects basic SQL authentication bypass attempts 3/3 | Tags: sqli, id, lfi | ID: 46
Description: Detects basic XSS DoS attempts | Tags: rfe, **DOS** | ID: 65
Description: finds malicious attribute injection attempts and MHTML attacks | Tags: xss, csrf | ID: 71
**Saldırı tespit kayıt sonu!...**

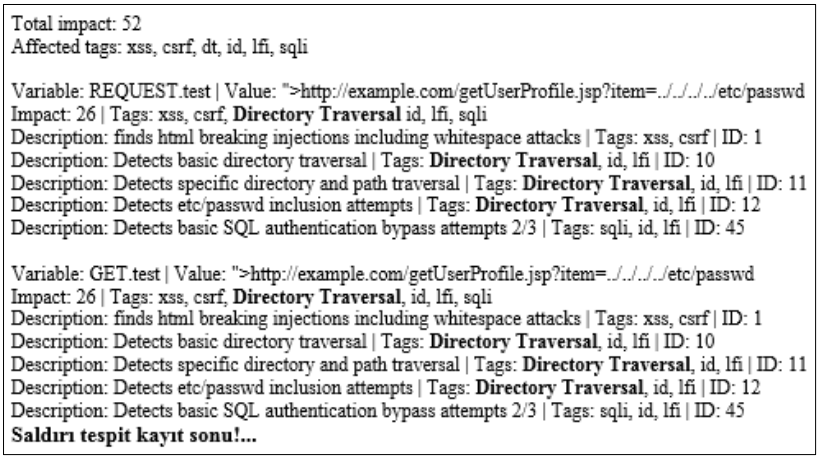
**Şekil 4.17** DoS saldırısı tespiti

### 4.4.7 Dizin Atlatma

Dizin atlatma (Directory Traversal), saldırganın erişim hakkına sahip olmadığı dizinlere erişerek faaliyet göstermesidir. Bu saldırıda saldırgan, IIS sunucuya bilinçli bir şekilde oluşturulmuş ve geçersiz *Unicode UTF-8* kodu serisi içeren URL göndererek sunucunun bir dizin dışına çıkabilmesini ve istenilen scriptlerin çalıştırılabilmesini sağlayabilir. Dizin atlama saldırısının çalışma mantığı Şekil 4.18'de verilmiştir. Şekil 4.19'da dizin atlatma saldırısı tespiti sonuçları verilmektedir. Value satırında görüldüğü gibi saldırgan, test edilen web sayfası üzerinde erişim yetkisi bulunmayan dizinlere erişmeye yönelik bir saldırı gerçekleştirmiş ve bu durum yakalanan paketin analizi ile tespit edilmiştir.




**Şekil 4.18** Dizin atlatma saldırı senaryosu




**Şekil 4.19** Dizin atlatma saldırısı tespiti

## 4.5 Sonuç ve Değerlendirme

Bu bölümde, güncel web açıklıklarına yönelik olarak geliştirilen ve önerilen gerçek zamanlı saldırı tespit ve engelleme sistemi yaklaşımının başarılı olduğu tespit edilmiştir. Geliştirilen uygulamalar, PHP ve ASP.NET temelli web uygulamalarına yapılabilecek tehditleri tespit edebilecek şekilde tasarlandığından hem Linux hem de Windows temelli sunucularda başarıyla uygulanabilmektedir. Honeypot-IDPS mantığında geliştirilen özgün bir sistemle birleştirilen uygulamalar, düşük ve yüksek etkileşimli tuzak sistemlerin olumlu özelliklerini kullanarak başarımı arttırmıştır. Gerçek zamanlı analizler sonucunda, görsel arayüzlerle sunulan bilgiler sayesinde sistem yöneticisi sistemde olup biten hakkında daha kolay bilgi sahibi olabilmektedir. Geliştirilen Honeypot-IDPS yapısı, imza tabanlı saldırı tespit sistemlerinden herhangi birine entegre edilebilecek şekilde tasarlanmıştır. Honeypot sistemler üzerine gelen trafiğin analiz edilmesiyle yeni saldırı imzaları üretilebilmektedir. Böylece sistem, sıfır gün (zero-day) diye tabir edilen, bilinmeyen yeni saldırıları da anormal durum olarak niteleyebilmektedir. Herhangi bir kötüye kullanım ya da anormallik tespiti durumunda paket iletimi kesilerek ağ, saldırılara karşı korunabilmektedir.

Geliştirilen honeypot sistem ile toplanan log kayıtlarının anormallik tespiti yöntemleri ile saldırı olmayan diğer kayıtlardan ayrıştırılması başarımı arttırmıştır. Bu log kayıtları ile STS'lerin dezavantajlarından olan yanlış alarm seviyesinin yüksek olması problemi çözülmüş olur. Çünkü honeypot sistemler genel itibariyle üzerinde kötücül aktiviteler barındırmaktadır. Böylece honeypota düşen saldırganların kimlikleri tespit edilip, gerçekleştirdikleri aktiviteler de analiz edilerek yeni saldırı örüntülerinin tespit edilmesi ve böylece yanlış alarm seviyesinin düşürülmesi mümkün olmaktadır.

Geliştirilen yazılımlar, ağ trafiğini gerçek zamanlı olarak inceleyip yakaladığı paketlerden gelen isteklerin analizine göre SQL Enjeksiyon, XSS, CSRF, Header Injection, LDAP Injection, Directory Traversal, RFI/LFI, DoS, Brute Force gibi saldırıları tespit edebilmektedir. Geliştirilen bu saldırı tespit ve engelleme sistemi yazılımları, yapısı itibariyle kural tabanlı ve anormallik tespit yaklaşımlarını birlikte sunan hibrit bir yapıda ve gerçek zamanlı olarak çalışan bir STS'dir. Gerçekleştirilen STS, kural tabanlı kötüye kullanım tespiti yaklaşımını kullandığı saldırı tespitlerinde %100 başarımla çalışmaktadır. Anormallik tespitine göre çalıştığı DoS benzeri saldırılarda ise değişen ağ trafiği şartlarına göre birçok parametreye göre kendini adapte etmek zorunda olduğundan başarımı daha düşüktür. Gerçekleştirilen saldırı tespit ve engelleme sistemi yazılımları, gerçek zamanlı veriler üzerinde çalışacak ve bilinen bütün saldırı türlerini tespit edebilecek bir seviyededir.

# 5. SALDIRI TESPİT VE ENGELLEME SİSTEMLERİ İÇİN HONEYPOT TEMELLİ YENİ BİR YAKLAŞIM

## 5.1 Giriş

Son yıllarda yaşanan olağanüstü teknolojik gelişmelerin sonucu olarak bilgisayarlar, modern çağın sosyal, kültürel, teknik anlamda her alanına girmiş ve hayatın vazgeçilmez bir unsuru olmuştur. Özellikle İnternet'in gelişimi ve yaygınlaşmasıyla birlikte global anlamda iletişimin standartları değişmiş ve farklı platformlar üzerinde aynı dili konuşan yazılımların gelişimi sonucu, sosyal ağlar, e-işlemler (e-banka, e-ticaret, e-eğitim, e-devlet vb.), temel iletişim ve etkileşim kanalı halini almıştır. Bilişim teknolojilerine ve özellikle İnternet ve İnternet temelli sistemlere olan bağımlılık hem kişisel, hem de ülkeler bazında her geçen gün artmaktadır.

2014 yılında Business Insider tarafından yapılan bir araştırmaya göre dünya genelinde dakikada 204 milyon e-posta gönderildiği belirlenmiştir. Yine aynı çalışmaya göre dakika bazında, Google arama motorunda 4 milyon arama yapıldığı, Youtube'a 120 saatlik video yüklendiği, Facebook üzerinde 3.3 milyon gönderi yapıldığı, WhatsApp üzerinde 50 milyar gönderi iletildiği tespit edilmiştir. Dünya nüfusunun yaklaşık olarak üçte ikisinin İnternet bağlantısı ve yüzde yirmisinin sosyal ağlara üyelikleri bulunmaktadır. Yine dünya genelinde cep telefonu kullanımının %85 olduğu ve cep telefonları ile alışveriş yapma yüzdesinin ise 15 olduğu bildirilmiştir [89]. Uluslararası Telekomünikasyon Birimi (ITU), 2015 yılı sonunda İnternet kullanıcısı sayısını 3.2 milyar olarak tahmin etmektedir. Bu veriler, bilişim teknolojilerinin hacminin ve bu teknolojilere olan bağımlılığın ne derecede arttığını göstermektedir.

Bilişim teknolojilerinde yaşanan bu gelişmeler, fiziksel ortamda işlenen hırsızlık, dolandırıcılık gibi eylemlerin sanal ortamda da gerçekleştirilmesini olası hale getirmiştir. Ayrıca bilişim teknolojileri, suç gruplarının veya terör örgütlerinin iletişim beceri ve imkânlarını artırmış, onlara yeni propaganda ve faaliyet alanları açmıştır [90]. Gelişen teknoloji ile birlikte ortaya çıkan bilişim suçlarının zaman içerisinde artması bilgi güvenliği kavramını ortaya çıkartmıştır. Bilgi güvenliği, bilginin gerek durağan ortamda, gerekse iletim ortamında göndericisinden alıcısına ulaşana kadar, gizlilik içerisinde, yetkisiz kişiler tarafından ulaşılamayacak bir biçimde, bozulmadan, değişikliğe uğramadan ve sürekli erişilebilir halde bulunmasını ifade eder [91].

Bilgi ve bilişim sistemlerini tehdit eden saldırı aktivitelerinin başlıca sebepleri ün, şöhret, şan gibi sebeplerden para veya ulusal çıkarlar gibi farklı olgulara dönüşmüştür [56]. Bilişim

suçlarında yaşanan bu değişim işlenen suçlarda kullanılan kötücül yazılımların içeriklerini de değiştirmiştir. Eskiden bilişim korsanları kullandıkları sistem açıklıklarını duyurarak şan ve şöhret kazanırken, artık gizlilik oldukça önemli bir nitelik olmuştur. Bilişim korsanları kullandıkları sistem açıklarını ne kadar gizli tutabilirlerse, sistemlerde alınacak güvenlik önlemlerini o kadar geciktirerek kazançlarını arttırabilmektedirler. Bu durum da bilgi sistemleri güvenliğine yeni bir anlam kazandırmıştır [92].

Bilgi güvenliğinin sağlanması amacıyla çok çeşitli donanım ve yazılım temelli araçlar kullanılabilmektedir. Kişisel ya da kurumsal ihtiyaca göre bilgi teknolojileri yöneticileri bu araçları sistem üzerinde uygun tasarım ile konumlandırıp bir bütün olarak çalışmasını sağlamalıdır. Ayrıca, bilişim sisteminin etkin ve dinamik çalışabilmesi, bu araçların sürekliliğinin ve güncelliğinin sağlanması ile mümkün olmaktadır [6]. Standartlara uygun bir bilgi güvenliği yönetim sisteminin, gizlilik, erişilebilirlik, güvenirlik, inkar edememe, kimlik tespiti, bütünlük ve log tutma özelliklerine sahip olması gerektiği Bölüm 3.2'de vurgulanmıştı.

Modern çağın vazgeçilmez kavramlarından biri haline gelen bilgi güvenliği alanında özellikle ön plana çıkan honeypot ve saldırı tespit sistemleri büyük önem kazanmaktadır. Tezin bu bölümünde, honeypot sistemler ile saldırı tespit ve engelleme sistemlerinin avantajları birleştirilerek hibrit bir yaklaşım önerilmiş ve geliştirilmiştir. Geliştirilen sistem kurumsal ağlarda güvenlik maliyetlerini düşürebilecek nitelikte tasarlanmıştır. Ayrıca, anormallik temelli saldırı tespit sistemlerinin en büyük dezavantajlarından biri olan yanlış alarm seviyesini düşürmekte ve sıfır gün (zero-day) güvenlik açıklıklarına karşı da adapte olabilen bir sistemdir. Böylece STS'ler imza tabanlarında olmayan yeni bir saldırıyı da tespit edip imza veri tabanlarını güncelleyebilmektedir.

## 5.2 Honeypot Sistemler

Dilimize "balküpü" olarak geçen honeypotlar, ağda kullanılan tuzak sistemlerdir. Honeypotlar, ağ ve bilgi güvenliğine yapılan saldırıların farkına varmak, saldırganların yöntemlerini izlemek, saldırı metotlarını belirlemek ve yeni geliştirilen saldırı çeşitlerinden önceden haberdar olmak amacı ile tasarlanmış yazılım veya sistemlerdir [13].

Honeypotlar, güçlerini saldırılabilir olma özelliklerinden almaktadır [14]. Kendi üzerlerinde güvenlik zafiyetleri oluşturularak veya güvenlik zafiyetlerinin tepkileri taklit edilerek, bir kovanın arıları bal yapmaları için üzerine çekmesi gibi, tuzak anlamında

saldırganları üzerine çeker ve saldırı yapmalarını sağlarlar. Üzerlerinde gerçek ve önemli bir bilgi bulundurmadıkları için de gerçek anlamda bir tehdit oluşturmazlar.

Diğer ağ ve bilgi güvenliği araçları olan saldırı tespit sistemleri ve güvenlik duvarlarından farklı olarak honeypotlar, özel bir problemin çözümünde kullanılmazlar. Honeypotlar, güvenlik sisteminin sadece bir parçasıdır ve hangi problemlerin çözümüne yardımcı olacakları tasarım ve kullanım şekilleri ile doğrudan bağlantılıdır [15]. Bu sebeple diğer bilgi güvenliği araçlarının aksine her problemin çözümüne genel cevap veren bir honeypot sisteminden söz edilemez [15, 16]. Literatürde honeypot sistemler ile saldırı tespit ve engelleme sistemleri gibi güvenlik uygulamalarının birlikte kullanıldığı çeşitli uygulamalar mevcuttur.

Güncel literatür araştırmalarında görüldüğü gibi, bilişim sistemleri ve ağ güvenliğinin sağlanmasında saldırı tespit sistemleri, saldırı engelleme sistemleri, güvenlik duvarları ya da honeypotlar tek başlarına yeterli olmamaktadır. Konu ile ilgili güncel çalışmalar incelendiğinde bu gibi güvenlik sistemlerinin birbirleriyle etkileşimli bir şekilde kullanıldığı hibrit yapılar önerildiği görülmektedir.

Bu tez çalışmasının bu bölümünde geliştirilen Honeypot-IDPS uygulamasında literatürdeki öneriler dikkate alınarak honeypotlar ile saldırı tespit ve engelleme sistemleri ortak bir biçimde kullanılmıştır. Böylece sistem başarımı arttırılmış, özellikle anormallik temelli saldırı tespit sistemlerinin dezavantajlarından olan yanlış alarm seviyesinin düşürülmesi ve bilinmeyen yeni saldırı örüntülerinin tespit edilebilmesi mümkün olmuştur.

Honeypotların başlıca kullanım amaçları aşağıda maddeler halinde verilmiştir;

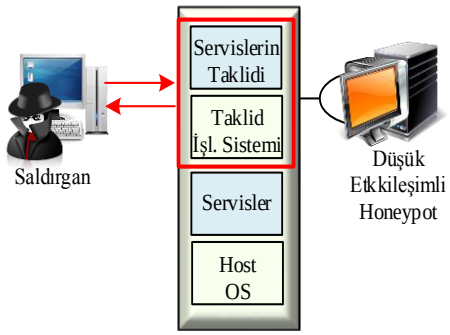
- Sistemdeki güvenlik zafiyetleri hakkında daha fazla bilgi sahibi olmak,
- Kurulan tuzak sistem sayesinde saldırganları tespit etmek,
- Ağda faaliyet gösteren kötücül eylemleri tespit etmek,
- Honeypotlar ile oluşturulacak sanal ağlar üzerinden gerçek sistemleri gizlemek,
- Yeni saldırı yöntemlerini (zero-day) tespit etmektir.

### 5.2.1 Honeypot Türleri ve Özellikleri

Honeypotlar, saldırgana sunacağı bilginin içeriği, değeri ve seviyesi, saldırganın erişebileceği kaynaklar ve honeypotun saldırganlar tarafından tespit edilmesi konularında incelenebilir. Honeypotlar, etkileşim seviyelerine göre düşük, orta ve yüksek etkileşimli olmak üzere üç grupta değerlendirilebilir [16].

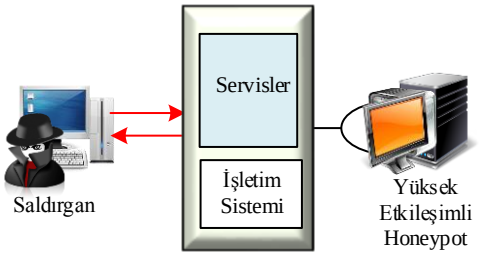
Düşük ve orta etkileşimli honeypotlar, saldırganları üzerlerine çekebilmek için özellikle güvenlik zafiyeti bulunan servisleri taklit ederler [17, 18]. Şekil 5.1'de gösterildiği gibi

üzerlerinde gerçek veya önemli bilgi bulundurmazlar. Sistemde var olmak için sanallaştırma teknolojilerinden faydalanabilirler. Saldırganın gerçek sisteme doğrudan erişimi bulunmadığı için honeypotun ele geçirilme riski çok düşüktür. Saldırgan gerçek sistemle bire bir iletişime geçmediği için saldırı hakkında daha az detaylı bilgi toplanabilir. Düşük ve orta etkileşimli honeypotlarda servislerin doğru taklit edilmesinin önemi büyüktür. Bu işlem sırasında yapılacak hatalar, saldırganın honeypotu tespit etmesine sebebiyet verebilir. Honeypotun tespit edilmesi, istenmeyen bir durumdur. Tespit edilen bir honeypot, saldırganlar için çekiciliğini kaybederek amacını gerçekleştiremeyecektir.




**Şekil 5.1** Düşük etkileşimli honeypot sistemler

Yüksek etkileşimli honeypotlarda gerçek servisler çalıştırılarak saldırganın dikkati çekilmek istenir [19]. Bununla birlikte saldırganın faaliyetlerini izleyebilmek için haricen yazılımlar kullanılmaktadır [20]. Şekil 5.2'de gösterildiği gibi yüksek etkileşimli honeypotlarda düşük ve orta etkileşimli honeypotlara nazaran saldırgan gerçek sistemle bire bir etkileşimde olduğu için honeypotu ele geçirme riski daha yüksektir. Bunun önüne geçebilmek için güvenlik duvarlarında bazı önlemler alınabilir. Ayrıca saldırgan gerçek sistem ile iletişimde olduğundan saldırı hakkında daha detaylı bilgi edinilebilir. Servisler gerçek olarak çalıştığından dolayı saldırganın honeypotu tespit etmesi daha zordur. Bu yapıda sanallaştırma teknolojileri kullanılarak ağda bir fiziksel makine üzerinde birçok honeypot konumlandırılabilir.




**Şekil 5.2** Yüksek etkileşimli honeypot sistemler

Yüksek etkileşimli honeypotlar daha maliyetlidirler ve daha sık bakım isterler. Avantajlarının yanında yeni güvenlik zafiyetlerine de neden olabilirler. Yüksek etkileşimli

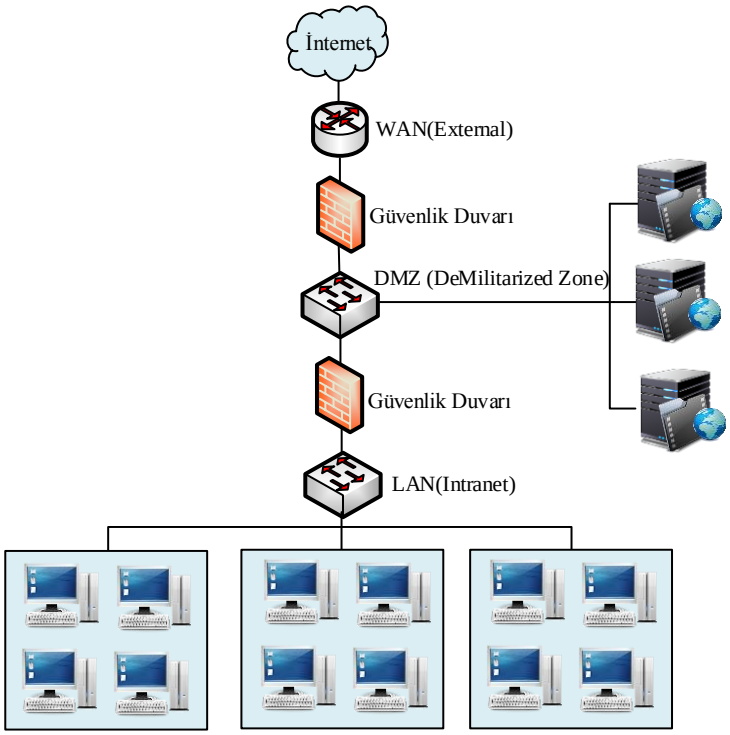
honeypotların kullanıldıkları ağların tamamen yalıtılması ve tüm güvenlik önlemlerinin alınması gerekir. Aksi takdirde saldırgan gerçek sistemle iletişimde olduğundan honeypota sızarak sistemi ele geçirebilir ve yeni güvenlik tehditleri oluşturabilir.

### 5.3 Kurumsal Ağlarda Honeypotların Konumlandırılmaları

Kurumlar, kullanım amaçları doğrultusunda kurum ağlarında güvenliğin bir parçası olarak honeypotları kullanabilir. Bunu yaparken honeypotun kullanım amacına göre ağda nerede konumlandırılacağı büyük önem arz etmektedir. Yanlış şekilde konumlandırılan bir honeypot amacına hizmet etmeyebilir. Özellikle büyük kampüs ağlarını da barındıran kurumsal ağlarda ağ faaliyetlerini izlemek ve kötücül eylemleri tespit etmek güçleşmektedir.

Honeypotlar, kurumsal ağlarda konumlarına göre iç veya dış ağdaki saldırganları üzerlerine çekerek ağda yürütülen kötücül faaliyetler hakkında bilgi toplanmasını, toplanılan bilgiler ile kurumsal ağ güvenliğini arttırmaya yönelik önlemler alınmasını, saldırganların tespit edilmesi ve engellenmesini kolaylaştırmaktadır. Honeypotların bu eylemleri gerçekleştirebilmeleri için gerekli ağ güvenlik araçlarından honeypotlara özel izinler verilmesi gerekebilir. Honeypotların ele geçirilmesini önlemek ve DoS, DDoS gibi hizmetleri sekteye uğratan saldırıların önüne geçebilmek için honeypotların kullandıkları bant genişliğinin doğru belirlenmesi ve honeypota özel güvenlik önlemlerinin alınması önemlidir. Bu güvenlik önlemleri alınmadığı takdirde, güvenliğin bir parçası durumundaki honeypotlar saldırganlar için bir saldırı aracına dönüşebilir [15, 16].

Kurumsal ağlar, bir kurumun kendi iç sunucularını, dış hizmet sunucularını ve intranet ağını İnternet ile haberleştirdiği ağlardır. Bu ağlarda farklı amaçlar için kullanılan sistemler için farklı güvenlik önlemleri almak gerekmektedir. Honeypotlar kullanım amaçlarına göre ağda farklı konumlarda bulunabilir. Bu açıdan kurumsal ağlar incelendiğinde üç ana bölüm olduğu gözlenmektedir. Bu alanlar, Şekil 5.3'te gösterildiği gibi, yönlendiricinin bulunduğu dış bölge, kurumun dış hizmetleri sunduğu sunucuların bulunduğu DMZ bölgesi ve iç ağının bulunduğu intranet bölgeleridir.




**Şekil 5.3** Kurumsal bir ağ altyapısı

Şekil 5.3'te gösterilen ağ yapısında çift güvenlik duvarı kullanılarak, DMZ bölgesi ve LAN bölgesi farklı güvenlik duvarlarıyla korunmuştur. Dış (External) bölge görüldüğü gibi yönlendirici ile güvenlik duvarı arasında kalan bölgedir. Bu bölge tamamen korunmasız ve saldırılara açıktır. Türkçe'ye sivil bölge olarak da geçen DMZ, kurumun dış hizmetlerini verdiği sunucuların bulunduğu bölgedir. Dış ağ ile doğrudan iletişim halinde olduğu için güvenlik duvarı üzerinde sadece kısmi güvenlik önlemleri alınabilir. Bu alan da kısmen saldırılara açık bir bölgedir.

Kurumun iç ağının bulunduğu LAN bölgesi ile dış ağın doğrudan iletişim halinde olması gerekmez. Bunun için arada NAT (Network Address Translation) ve PAT (Port Address Translation) gibi teknolojiler kullanılır. İç ağın güvenliğinin sağlanması için DMZ bölge ile arasına bir güvenlik duvarı daha konulabilir. Bu şekilde kısmi güvenliğe sahip DMZ'de bulunan bir sistemin ele geçirilmesi halinde iç ağ korunmaya devam edilebilir. Kurumda iç ağın da kendi içinde ağlara ayrılması ve alt ağların da birbirinden yalıtılması gerekebilir. Bu gibi durumlarda VLAN (Virtual LAN) teknolojisi kullanılarak güvenlik sağlanabilir. Kurumların saldırganlar tarafından dış ağ yoluyla saldırılara maruz kalabilecekleri gibi iç ağdan da saldırıya uğrayabilecekleri unutulmamalıdır. İç ağda bulunan bir saldırgan dış ağdaki bir saldırgandan kurumun ağ sistemi hakkında daha fazla bilgi ve yetkiye sahip olduğu için daha tehlikeli bir senaryo oluşturabileceği düşünülebilir. Bu durumda

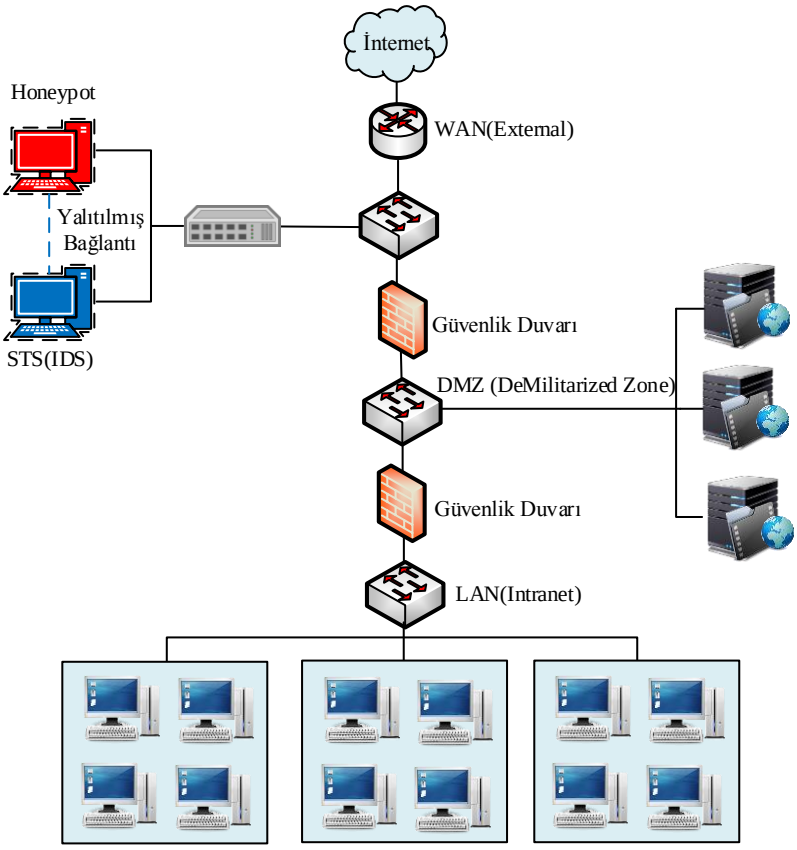
oluşturulan VLAN'ların güvenliğinin sağlanması ve VLAN'lar arasındaki iletişimin güvenlik düzeyi büyük önem teşkil etmektedir.

Honeypotların amaçlarına göre ağda nasıl konumlandırılacağı ve özellikle az sayıda honeypot ile kurumsal ağın tüm bölgelerinin güvenliğine, honeypotları dâhil etmek de önem teşkil etmektedir. Her konumlandırılan honeypot için ayrıcalıklı izinler alınması, özel güvenlik önlemleri ve bant genişliklerinin belirlenmesi gerekmektedir. Büyük kampüs ağlarını barındıran kurumsal ağlarda birden fazla honeypot konumlandırmak bakım ve kurulum maliyetlerini arttırmaktadır. Honeypotun konumunun belirlenmesinin yanında etkileşim seviyesinin de belirlenmesi gerekmektedir. Belirlenen etkileşim seviyesine göre honeypotlar için alınacak güvenlik önlemleri de değişecektir. Konumlarına göre honeypotları kurumsal ağları ayırdığımız bölgelerin özelliklerine göre gruplandırabiliriz. Bu şekilde honeypotlar İnternet bölgesi üzerinde, DMZ üzerinde ve LAN üzerinde konumlandırılabilirler [15, 16, 21]. Her yöntemin birbirlerine göre avantaj ve dezavantajları bulunmaktadır.

### 5.3.1 Honeypotların İnternet Üzerinde Konumlandırılması

Honeypot sisteminin kurumsal ağ dışındaki İnternet bölgesinde konumlandırılmasıdır. Şekil 5.4'te honeypotun İnternet üzerinde nasıl konumlandırılacağı gösterilmektedir. Bu yapıda honeypotu haricen dinleyen bir STS ile honeypotun faaliyetleri izlenmektedir. Honeypotu konfigüre etmek için STS ile arasındaki ağın tamamen yalıtılmış olması önemlidir. Bu şekilde honeypotların ağda tespit edilmeleri ve sabote edilmeleri engellenmiş olacaktır.

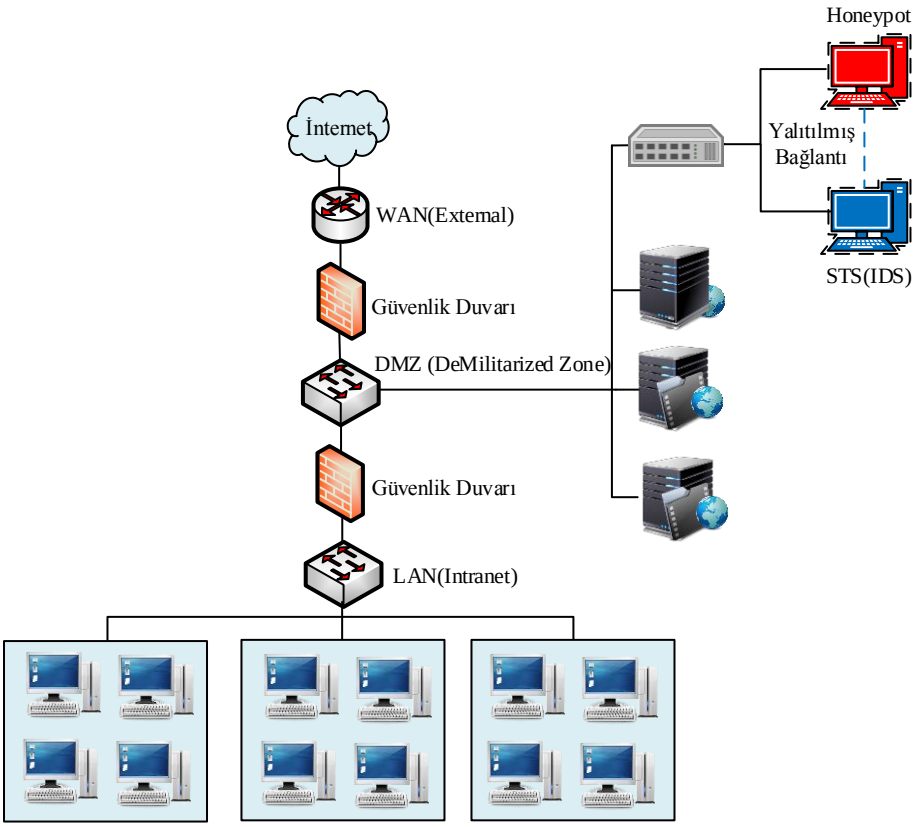
Honeypot, direk İnternet üzerinde bulunduğu için herhangi bir güvenlik duvarı tarafından korunmamaktadır. Bu durum, honeypotun faaliyetlerini engelsiz bir şekilde gerçekleştirmesini sağlayacak ama honeypotun ele geçirilmesine karşı alınacak güvenlik önlemlerini zorlaştıracaktır. Ayrıca bu konumlandırma senaryosu, DMZ ve LAN bölgelerinin güvenliğine honeypotun dahil edilmesini mümkün kılmamaktadır [21]. Bu yaklaşımda daha çok dış IP adresleri honeypota yönlendirilerek sadece dış ağdan IP aralıklarına gelebilecek saldırılar saptanabilmektedir.




**Şekil 5.4** Honeypot sistemlerin İnternet bölgesinde konumlandırılması

### 5.3.2 Honeypotların DMZ Üzerinde Konumlandırılması

Honeypotun DMZ (DeMilitarized Zone) bölgesinde konumlandırılması, hem DMZ sunucularını gizlerken, hem de DMZ bölgesinin içten ve dıştan gelecek saldırılara karşı tuzak sistemleri ağa dâhil etmiş olacaktır. Dış ağ ile DMZ bölgesi arasındaki güvenlik duvarından honeypotun kullandığı IP adreslerine özel izinler verilerek İnternet bölgesinde konumlandırılmış bir honeypotun eylemlerini de gerçekleştirebilmesi sağlanabilir. Fakat bu yapıda LAN bölgesinin ağ güvenliğine honeypotları dâhil etmek mümkün olmayacaktır [21]. Şekil 5.5'te honeypotun DMZ üzerinde konumlandırılması gösterilmektedir.

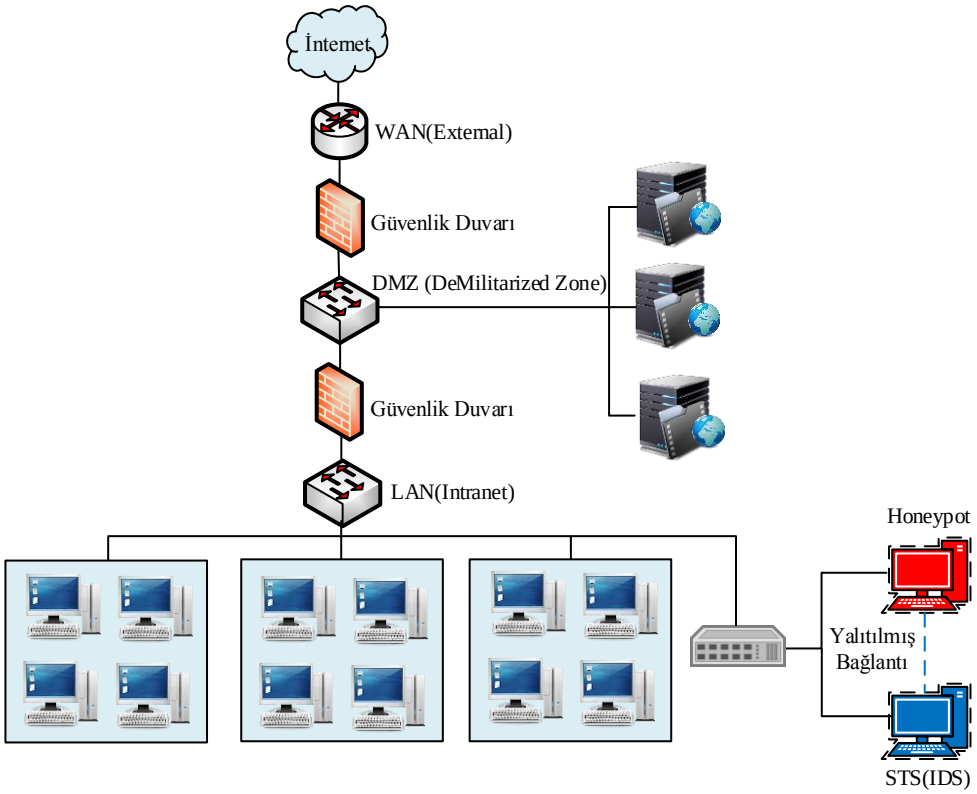



**Şekil 5.5** Honeypot sistemlerin DMZ bölgesinde konumlandırılması

### 5.3.3 Honeypotların LAN Üzerinde Konumlandırılması

Honeypotların LAN üzerinde konumlandırılması ile iç ağ güvenliğine honeypotların dahil edilmesi mümkün olmaktadır. Ağ güvenliği için kullanılan tüm güvenlik duvarlarından honeypotun kullandığı IP adreslerine özel izinler verilerek İnternet ve DMZ bölgesinin güvenliğine de honeypot sistemi dâhil edilebilir. Eğer LAN ağında VLAN konfigürasyonu yapılmışsa, honeypotun her VLAN'a dahil olabilmesi için her VLAN'da farklı bir honeypot kullanılması gerekebilir [21]. Bu durum, honeypotun özellikle büyük kampüs ağlarını barındıran kurumsal ağlardaki kurulum ve bakım maliyetini yadsınamayacak derecede arttıracaktır. Bunun için bir çözüm, honeypot sunucu üzerine ağdaki VLAN sayısı kadar ağ arabirimi ekleyerek tüm VLAN'larda faaliyet gösterilmesini sağlamaktır. Bu çözüm maliyeti azaltsa da mutlak bir çözüm sunmamaktadır. Şekil 5.6'da honeypotun LAN üzerinde konumlandırılması gösterilmektedir.

LAN bölgesinde honeypotun ele geçirilme ihtimali ağ güvenliği için çok büyük bir tehlike oluşturacağı için bu yöntemde düşük etkileşimli honeypotların kullanılması daha mantıklı olmaktadır.




**Şekil 5.6** Honeypot sistemlerin LAN bölgesinde konumlandırılması

## 5.4 Önerilen Honeypot Temelli Yaklaşımın Uygulanması

Honeypot sistemler saldırganı tuzağa çekme, saldırgan ve sistem hakkında veri toplama, toplanan veri ve bilgiler ışığında kapsamlı veri analizi yapabilen yapılar içerir. Honeypotun etkileşim seviyesi, saldırı çekme türünü belirleyici niteliktedir. LAN üzerinde konumlandırılmış honeypotlar, kurumsal ağ içindeki tüm bölgelere ulaşabilmektedir. Bu nedenle kurum ve kuruluşlar üzerinde kullanılan bu honeypotlar sayesinde, sistemler üzerine yapılan saldırıların tespiti, analizi ve engellenmesi mümkün olmaktadır.

Bu bölümde, kurumsal ağ trafiğinin izlenmesi, denetlenmesi ve kötücül aktivitelerin önüne geçilebilmesi amacıyla yazılımsal olarak honeypot temelli bir saldırı tespit ve engelleme sistemi tasarımı yapılmıştır. Gerçekleştirilen uygulamanın temel bileşenleri üç grupta incelenebilir. Bunlar, tuzak sistemleri simüle edebilecek "*honeypot sunucu uygulaması"*, honeypot sunucularından yakalanan saldırıların animasyon olarak gösterileceği "*monitör uygulaması"* ve bu monitör uygulaması ile honeypot sunucu uygulamalarının konfigürasyonlarının yapılacağı, honeypot sunuculara gelen paketlerin yakalanarak monitöre gönderileceği "*STS uygulaması*" 'dır.

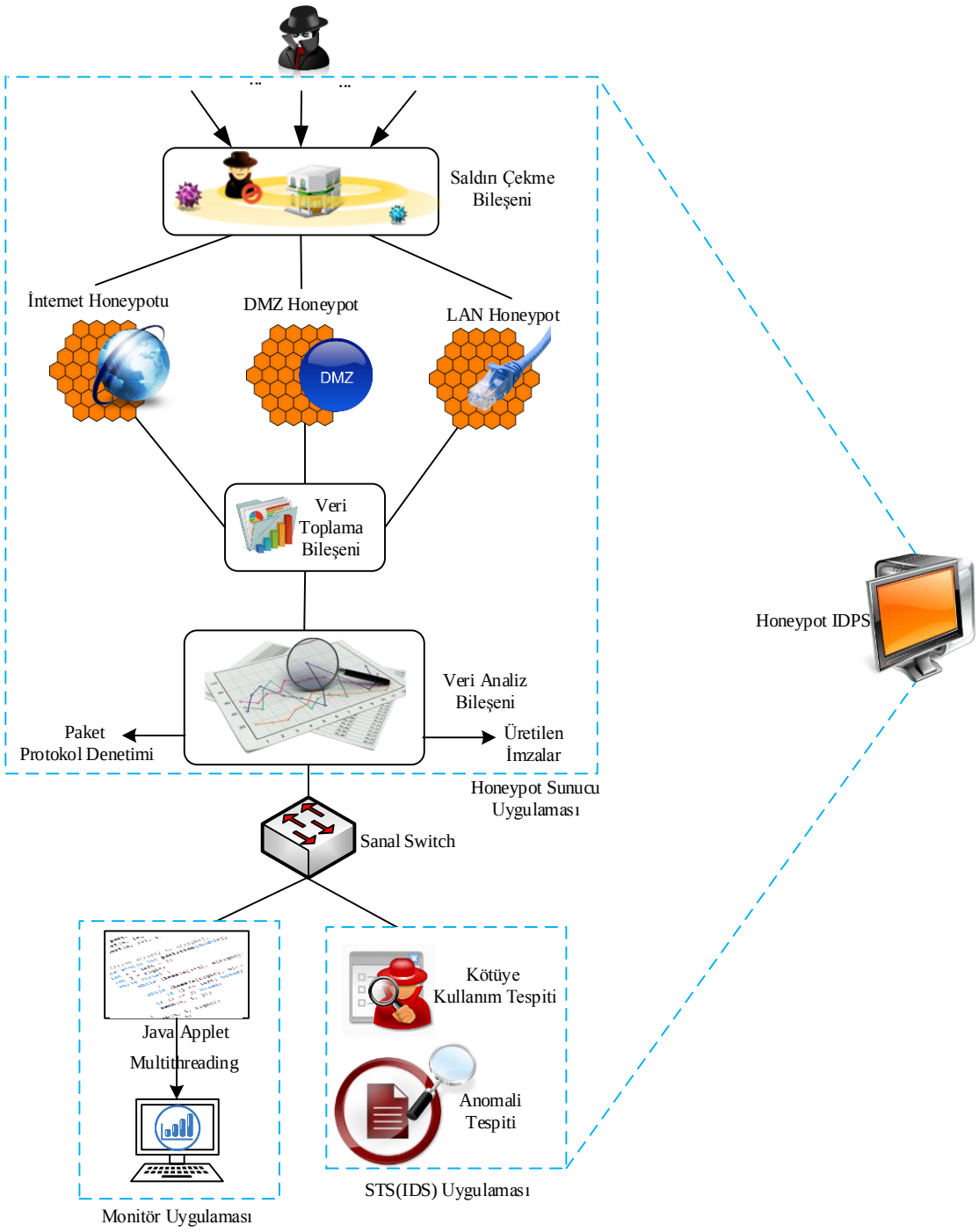
LAN bölgesinde konumlandırılmış honeypotun ele geçirilmesinin çok büyük güvenlik risklerine neden olduğu bilinmektedir. Bu nedenle, geliştirilen uygulamada honeypotun

saldırı çekme bileşeni, düşük etkileşimli yapıda tasarlanmıştır. Honeypotun veri toplama ve veri analizi bileşenlerinin yüksek etkileşimli sistemlerdeki gibi harici bir yazılım ile sağlanması planlanmıştır. Böylece daha esnek izleme ve detaylı analiz imkânı sağlayan hibrit bir honeypot sistemi tasarlanmasının daha uygun olduğu ortaya konulmuştur. Bu çalışma ile özellikle büyük kurumsal kampüs ağlarında çalışan VLAN ağları üzerinde, honeypotların konumlandırılmasında kurulum ve bakım maliyetlerini düşürmek için yeni, merkezi kontrol sağlayan bir yöntem önerilmektedir.

Mevcut saldırı tespit sistemleri ve honeypotların incelenmesi sonucu elde edilen bulgular ışığında, honeypotların düşük etkileşimli olarak LAN bölgesi üzerinde konumlandırılması uygun görülmüştür. Düşük etkileşimli honeypot tasarımı için *honeyd* uygulamasının saldırı çekme bileşeni kullanılmıştır. Ağ paketlerinin yakalanması, verilerin toplanıp detaylı analizinin daha etkin ve hatasız yapılabilmesi için merkezi bir yönetim önerilmiştir. Bunun için veri toplama bileşeni ve toplanan paketlerin daha ayrıntılı analizi için veri analiz bileşeni STS uygulaması içine eklenmiştir. Bu şekilde ağ üzerinde çalışan birden fazla honeypot merkezi bir noktadan dinlenerek, işlemlerin daha kolay ve hızlı yapılması sağlanmıştır.

Honeypot sunucusu, monitör ve STS uygulamaları sanallaştırma teknolojisi ile birbirinden mantıksal olarak ayırılmış ve bir fiziksel makine üzerinde çalışması sağlanmıştır. Böylece, saldırganın honeypotu ele geçirmesi(saldırganın tuzağa düşmesi) durumunda oluşabilecek riskler azaltılmış olacaktır.

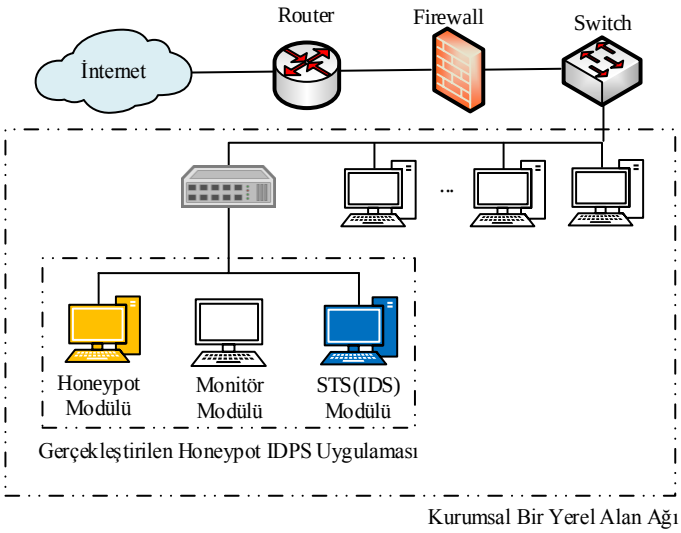
Honeypot STS uygulamaları arasında sistem ayarlarının yapılandırılması için özel sanal ağlar kurulur. Böylece honeypot uygulamalarının ağ iletişimi de fiziksel ağdan yalıtılmış olur. Önerilen Honeypot-IDPS yapısının genel bir diyagramı Şekil 5.7'de gösterilmektedir. Geliştirilen Honeypot-IDPS; Şekil 5.4, Şekil 5.5 ve Şekil 5.6'da gösterilen honeypot yerleşim senaryolarında verilen tüm konumlandırmalarda saldırı tespit ve analizi yapabilmektedir.




**Şekil 5.7** Geliştirilen Honeypot-IDPS sistemi çalışma diyagramı

Geliştirilen uygulamanın bütünlüğü için honeypot sunucu uygulaması, C# programlama dili ile yazılmıştır. Linux işletim sistemlerinde, .Net teknolojilerine çapraz-platform desteği veren ve Linux ortamında da .Net dilleriyle yazılmış olan uygulamaların derlenmesine imkân sağlayan framework alanını içeren Mono projesi kullanılmıştır. Bu şekilde Linux ortamında çalışacak olan honeypot sunucu uygulamasının saldırı çekme bileşeni için kullanacağı honeyd uygulamasının da aynı programlama ortamı kullanılarak

geliştirilebilmesi sağlanmıştır. Şekil 5.8'de, geliştirilen honeypot temelli saldırı tespit ve engelleme sistem modüllerinin bir kurumsal yerel alan ağı üzerindeki konumlandırılması gösterilmektedir.




**Şekil 5.8** Honeypot-IDPS uygulamasının ağ üzerindeki konumlandırılması

### 5.4.1 Honeypot Sunucu Modülü

Honeypot sunucu uygulaması 3 temel bileşenden oluşmaktadır. Bu bileşenler düşük etkileşimli saldırı çekme bileşeni, konfigürasyon bileşeni ve STS iletişim bileşenidir.

*Saldırı çekme bileşeni:* Saldırı çekme bileşeni, düşük etkileşimli saldırı çekme bileşenine sahip *honeyd* uygulaması ile sağlanmıştır.

*Konfigürasyon bileşeni:* Bu bileşen ile STS uygulaması ve *honeyd* uygulamasının saldırı çekme bileşeninin konfigüre edilmesi sağlanmıştır.

*STS iletişim bileşeni:* Bu bileşen ile honeypot sunucu modülü, STS modülü ile iletişim sağlayacaktır.

Düşük etkileşimli honeypotlar, ağı 2. katman seviyesinde dinleyen ağ dinleme bileşenleri sayesinde ağda sahte bilgisayar profilleri oluşturabilme yeteneklerine sahiptirler. Ayrıca 2. katman seviyesinde dinlenen bir ağda; honeypotlar, IP adreslerine gelen ARP isteklerini cevaplar. Böylece fiziksel olarak ağda bulunmamasına rağmen sahte bir bilgisayar profili oluşturabilirler. Bu şekilde sanal bir makinada çalışan bir honeypot uygulaması ile ağda birden fazla sahte IP adresine sahip tuzaklar oluşturulabilir. Aynı zamanda çeşitli scriptler kullanılarak diğer ağ katmanı paketlerini de taklit edebilirler. Bu da saldırganda honeypotun oluşturduğu tuzak IP adresine sahip sahte bilgisayarın gerçek olduğu algısını oluşturur. Honeypotun özellikle uygulama katmanı protokollerine vereceği tepkiler için kullanacağı scriptler, piyasada bilinen açıklıkların belirtilerini taklit edecek şekilde yazılır. Saldırgan,

honeypot ağının gerçek bir bilgisayar olduğu algısına kapıldığı zaman hedef sistem üzerinde yapacağı araştırmalarda, mevcut açıkların olduğunu görür. Bu açıklar, saldırganı direk tuzak sisteme düşürecek yem olarak kullanılır. Daha kaba bir tabirle saldırgan bal peteğine düşmüş olur. Bu şekilde ağda faaliyet gösteren gerçek sistemleri de kamufle etmek mümkün olabilir. Honeypotların taklit ettikleri servislerde, gerçek bir bilgi bulunmayacağı için ağda oluşturulan bu tuzak IP adreslere saldırgan olmayan kullanıcıların erişmesi beklenmez. Bu şekilde honeypotun ağ üzerinde simüle ettiği bilgisayarların IP adreslerine gelen tüm istekler saldırı olarak nitelendirilebilir. Daha sonra yeni saldırı yöntemlerini (zero-day) tespit etmek için honeypotlardan saldırı niteliğindeki paket bilgileri toplanır. Bu toplanan veriler (log files) analiz edilmek üzere kaydedilir. Bunun yanı sıra tutulan kütükler (log) incelenerek kurumsal ağdaki zafiyetler hakkında bilgi edinilebilir. Ancak, gerçek zamanlı olarak honeypotlara düşen saldırganlar, güvenlik duvarı aracılığı ile engellenebilir.

Honeypotlara gelen paketler, saldırganların davranışları hakkında kritik bilgiler verir. Saldırgan davranışları gerçek zamanlı bir arayüz ile izlenebildiğinden kötü niyetli girişimlerin görüntülenebilmesi de kolaydır. Bu çalışmada, monitör uygulaması ile honeypotlardaki trafikler kaynak ve hedef lokasyon bilgileriyle birlikte bir animasyon şeklinde görselleştirilmiştir.

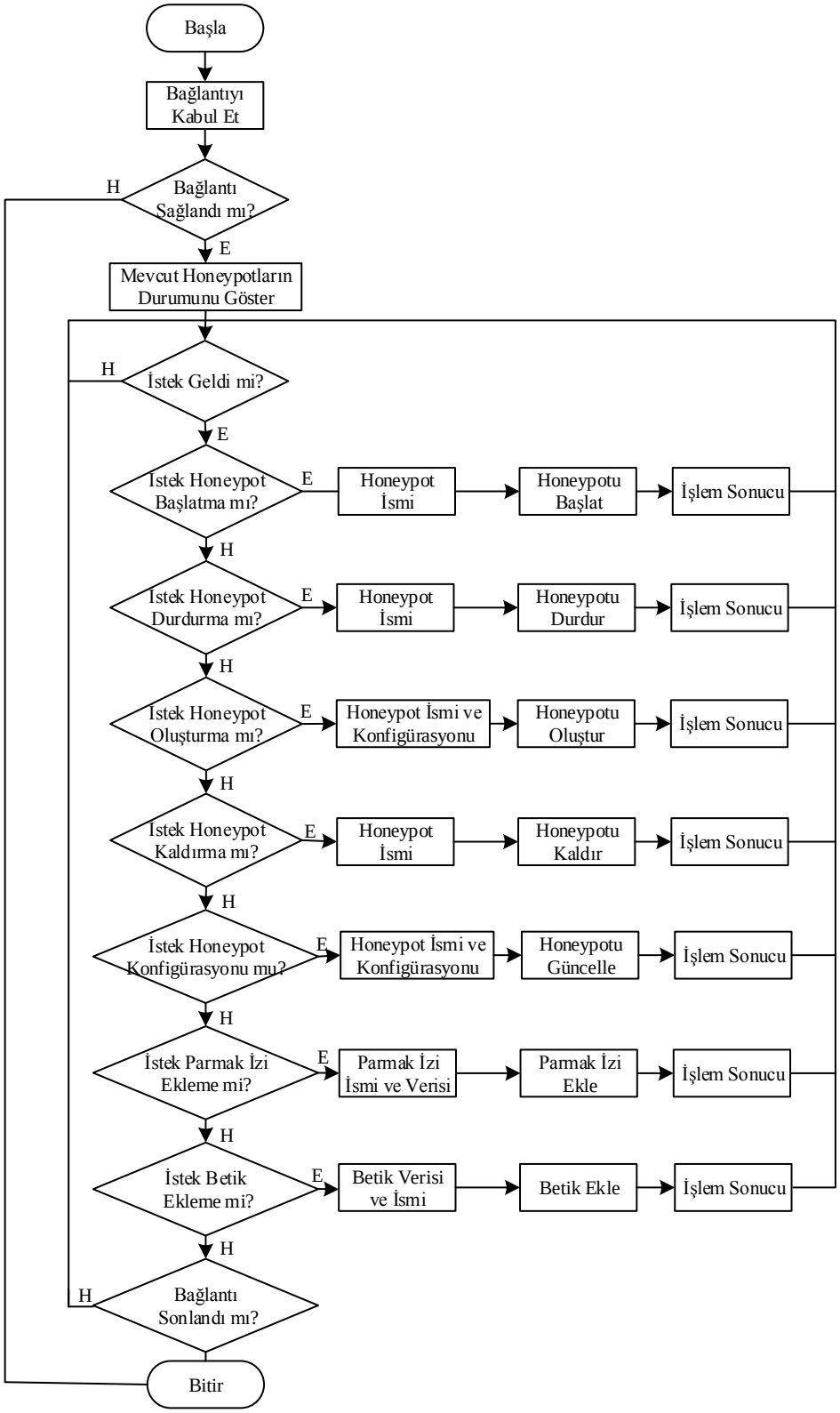
Honeypot sunucu uygulaması, üzerinde çalıştırdığı ***honeyd*** uygulamasını konfigüre etme görevine de sahiptir. Bunun için *honeyd* uygulaması ile honeypot sunucu uygulamaları arasındaki iletişimi sağlayacak olan özgün bir Python betiği yazılmıştır. Honeypot sunucu uygulaması, STS uygulamasından aldığı direktifleri Python betiği aracılığıyla *honeyd* uygulamasına iletir. Yine benzer şekilde aracı betiği kullanarak mevcut honeypotların durumları hakkındaki bilgilere erişebilir. Elde ettiği bilgileri STS'ye ileterek çalışan honeypotların durumlarının sistem yöneticisine bildirilmesini sağlar. Yanlış veya eksik konfigürasyona sahip olan honeypotları raporlayarak oluşabilecek sistem hatalarını engellemeye yönelik önlemler alır. Bir işletim sistemi üzerinde birden fazla arayüzde çalıştırılmak istenebilecek honeypotlar için aynı anda birden fazla *honeyd* uygulamasını çalıştırarak gerekli konfigürasyonu yapabilme kabiliyetine de sahiptir.

Honeypot sunucu uygulamalarının STS uygulaması ile gerçekleştireceği iletişimi sağlamak için belirli bir protokol geliştirilmiştir. STS uygulamasının honeypot sunucu ile TCP oturumu başlatması ile protokol başlar. Honeypot sunucu uygulaması STS uygulamasına üzerinde çalıştırdığı *honeyd* uygulamalarından bilgileri toplayarak, çalıştığı işletim sistemi üzerindeki tüm honeypotların durumu hakkında bilgi verir. Daha sonra

STS'den gelebilecek direktifleri beklemeye başlar. STS uygulamasından honeypot sunucu uygulamasına gönderilebilecek direktifler aşağıdaki gibidir:

- Honeypot başlatılması
- Honeypot durdurulması
- Yeni honeypot oluşturulması
- Mevcut honeypotun kaldırılması
- Mevcut honeypotun konfigürasyon yapılandırılması
- Sisteme yeni farklı izlerin (imza-parmak izi) eklenmesi
- Honeypotların servisleri taklit edebilmesi için kullanabileceği yeni betiklerin eklenmesi

Honeypot sunucu uygulaması ile STS uygulamasının iletişiminde kullanılan protokolün algoritması Şekil 5.9'da gösterilmektedir. Honeypot sunucu uygulaması konsol uygulaması olduğu için Windows temelli herhangi bir görsel arayüze sahip değildir. Bu nedenle, kullanıcıyla doğrudan bir iletişimi de yoktur. Yalnızca STS uygulaması ile iletişim halindedir.




**Şekil 5.9** Honeypot sunucusu ile STS iletişimindeki kuralların akış şeması

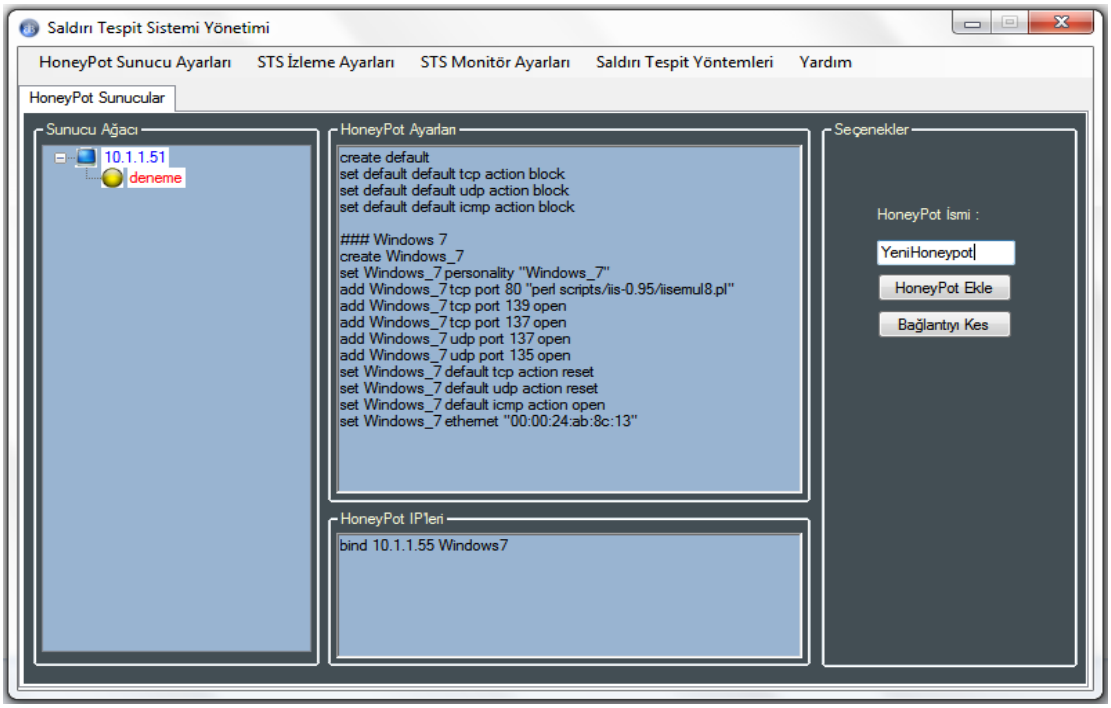
### 5.4.2 STS Modülü

STS uygulama modülü, honeypotların faaliyetlerini izleyen, arayüzleri vasıtasıyla sistem yöneticisi ile iletişime geçerek honeypot sunucu ve monitör uygulamalarının konfigürasyonlarını yapabilen yazılım aracıdır. Bunun için arayüzünde beş tane ana menü bulunmaktadır. Bunlar, honeypot sunucu ayarlarının yapılacağı "Honeypot Sunucu Ayarları" menüsü, ağ faaliyetlerini izlemek için kullanılacak "STS İzleme Ayarları" menüsü, monitör uygulamasının ayarlarının yapılabileceği "STS Monitör Ayarları" menüsü, saldırı tespit yöntemlerinin seçileceği "Saldırı Tespit Yöntemleri" menüsü ve "Yardım" menüsüdür.

Geliştirilen STS uygulaması, paket ve protokol bazlı analizler yapabilmektedir. Uygulama sayesinde istenilen portun ya da ağ katmanının dinlenmesi ve gelen/yakalanan paketlerin analiz edilmesi mümkündür. Bu seçenekli durumlar kullanıcıdan bir arayüz ile alınmaktadır. STS uygulamasına ait giriş ekranı Şekil 5.10'da gösterilmektedir. Kullanıcı ekranı olarak bilinen bu arayüz ile STS'nin yapılandırma ayarları için çeşitli menüler barındırmaktadır.

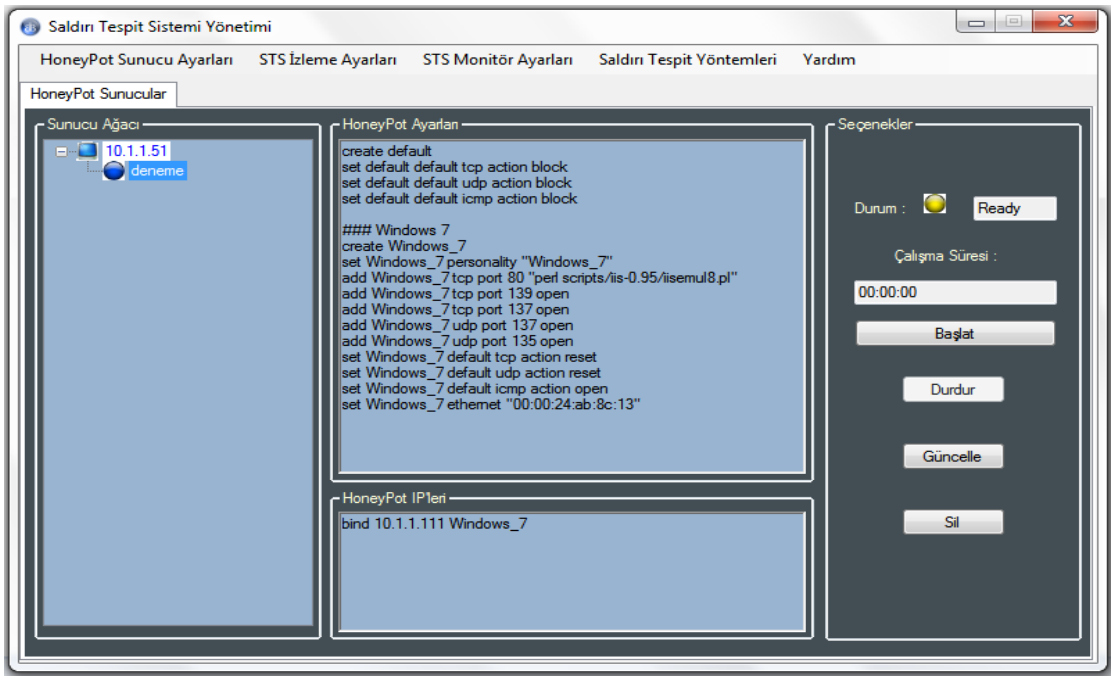

**Şekil 5.10** STS kullanıcı giriş ekranı

Şekil 5.11'de, "Honeypot Sunucu Ayarları" menüsünden yeni honeypot ekleme işlemi, Şekil 5.12'de ise honeypot işlemlerinin arayüz ile nasıl gerçekleştirildiği gösterilmektedir.




**Şekil 5.11** STS uygulamasının "Honeypot Sunucu Ayarları" menüsünde honeypot ekleme işlemi

Şekil 5.11 ile gösterildiği gibi sunucuya yeni bir honeypot eklerken aynı isimde daha önce oluşturulmuş bir honeypot var ise STS uygulaması karışıklıkları önlemek için bu işleme izin vermemektedir. Eğer honeypot ekleme işlemi sorunsuz bir şekilde gerçekleşmiş ise sunucu ağacı bölümünde yeni honeypot görünür hale gelir. Eğer işlem sırasında bir hata oluştuysa kullanıcıya bilgi verilir.

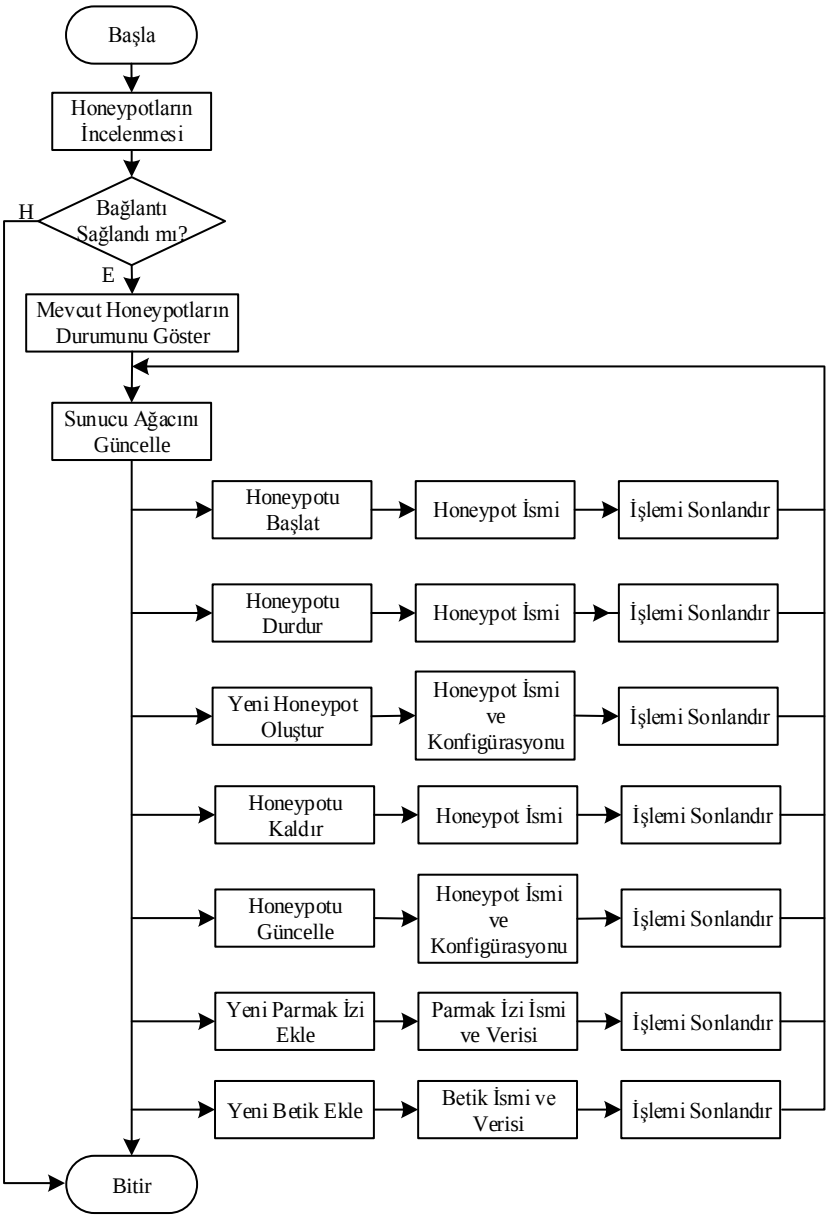



**Şekil 5.12** STS uygulamasında "Honeypot Sunucu Ayarları" menüsünde yapılabilecek honeypot işlemleri

Sistemde çalışan honeypotlar ile STS uygulamasının iletişime geçebilmesi için önce hangi IP adresleri üzerinde honeypot sunucuların çalıştırıldığını bilmesi gerekir. Bunun için honeypot sunucuların IP adreslerini girerek öncelikle STS uygulaması ile bağlantı kurulur. Daha sonra istenirse bu bağlantı sonlandırılabilir. Daha önceden kurulmuş olan bağlantılar hafızada tutulur ve bu sayede her seferinde IP adreslerinin yeniden girilmesine gerek kalmaz. Başarıyla bağlanılan honeypot sunucular, sunucu ağacı bölgesinde üzerlerinde bulunan honeypotlar ile gösterilir. Bu şekilde ağda aktif ve pasif durumdaki tüm honeypotlar hakkında bir arayüzden kolayca bilgi sahibi olunabilmektedir.

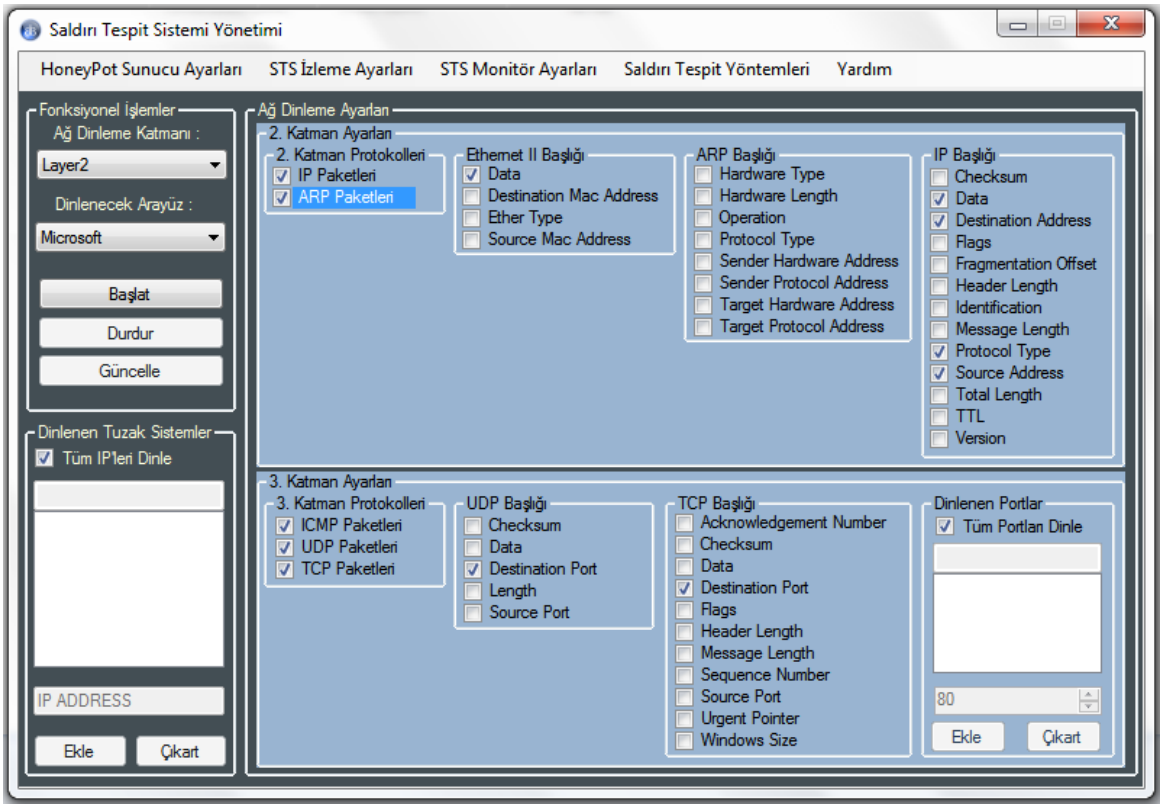
Şekil 5.12'de sunucu ağacından seçilen honeypotun konfigürasyon bilgisi ve şu anki durumu gösterilmektedir. Buna göre honeypotun konfigürasyonu değiştirilerek güncellenebilir. Bunun için honeypotun çalışma durumunda olmaması gerekmektedir. Eğer honeypot çalışır durumdaysa durdurulduktan sonra güncellenebilir. Honeypotun durumu *hazır*, *çalışıyor* veya *hazır değil* olabilir. Eğer bir honeypot çalıştırılması sırasında bir hata oluştuysa bu honeypotun durumu *hazır değil* olarak işaretlenir. Bu şekilde kullanıcı bu honeypot'u çalıştırmak için konfigürasyonunu değiştirmesi gerektiğini bilir. Sil butonu kullanılarak honeypot, sunucudan tamamen kaldırılabilir. Honeypot çalışır durumda ise ne kadar zamandır kesintisiz çalıştığı, çalışma süresine bakılarak öğrenilebilir. STS uygulamasına bağlantı sağlanan honeypotlar, otomatik olarak izleme modülüne gönderilerek hangi IP adreslerinin faaliyetlerinin izleneceği öğretilir.

STS uygulaması ile honeypot uygulamasının iletişimi sırasında kullanılan protokolün STS uygulaması tarafında çalışan algoritması Şekil 5.13'te gösterilmektedir.




**Şekil 5.13** Honeypot sunucusu kontrolü akış şeması

"STS İzleme Ayarları" menüsünden ağ dinleme katmanı, yakalanan paketlerin ayrıştırılma seçenekleri, dinlenilecek IP adresleri seçilebilir. STS uygulamasının "STS İzleme Ayarları" menüsü Şekil 5.14'te gösterilmektedir.

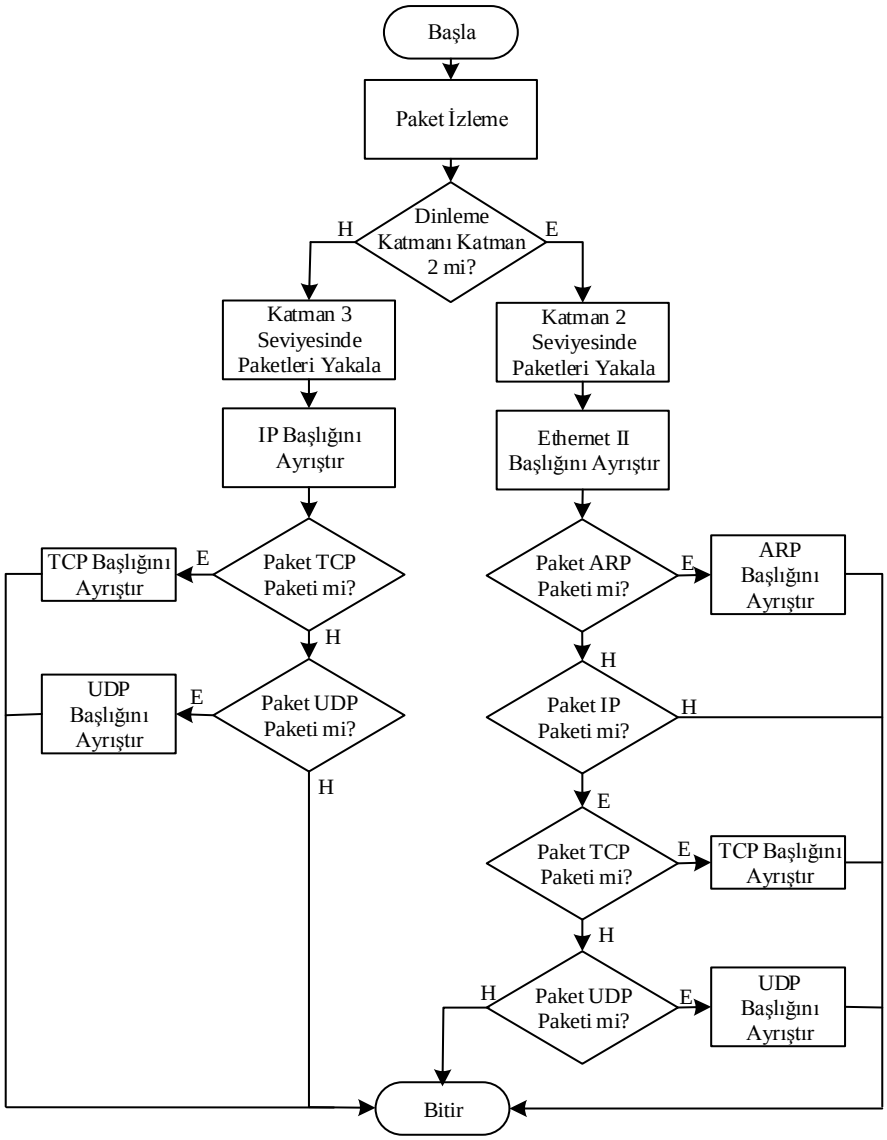



**Şekil 5.14** STS uygulamasının "STS İzleme Ayarları" menüsü

Geliştirilen uygulamada Şekil 5.14'te gösterildiği gibi, belirlenen ağ dinleme katmanına göre seçilen parametreler ve dinlenilmesi istenen paket tipleri ayarlanabilmektedir. TCP ve UDP paketleri için yalnızca istenen portlar dinlenebilir. Yazılım aracı istenilen IP adreslerinin seçilerek, dinlenilmesine olanak sağlamaktadır. "Honeypot Sunucu Ayarları" menüsünden bağlanılan aktif durumdaki honeypotların IP adresleri otomatik olarak dinlenen IP adreslerine eklenebilir.

TCP/IP, bilgisayarların karşılıklı haberleşebilmesi için kullanılan en yaygın protokol kümesidir. TCP/IP'de, gönderilen veriler katmanlara göre paketlenerek iletilir ve alıcıda bu paketler teker teker açılıp veri hedefe ulaştırılır.

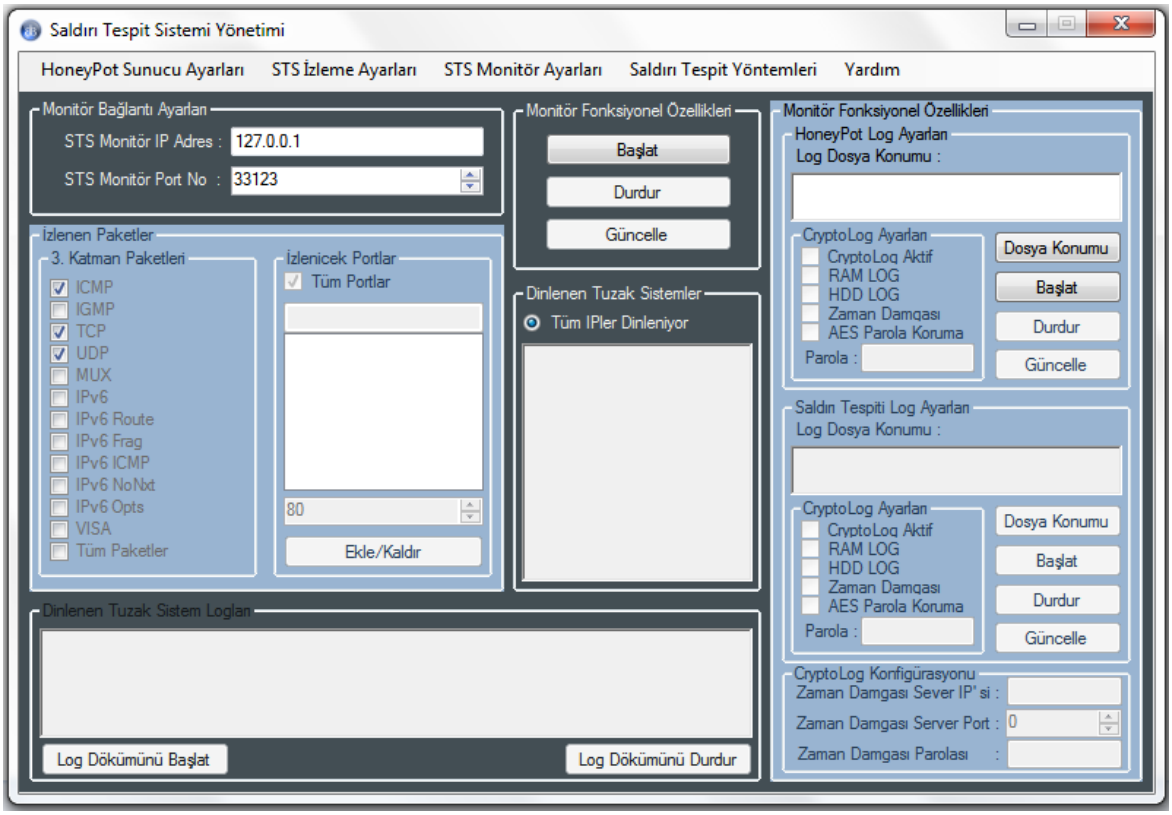
STS uygulamasında incelenip ayrıştırılan protokoller; IP, ARP, TCP, UDP protokolleridir. İkinci katmanda Ethernet II, ARP ve IP başlıkları, üçüncü katmanda UDP ve TCP başlıkları ayrıştırılmaktadır. İncelenen protokollerin başlıklarının içerikleri ait oldukları sınıflardaki değişkenler ile ifade edilerek ağda yakalanan paketler protokol başlıklarına göre ayrıştırılmıştır. Şekil 5.15'te ayrıştırma (parse) işleminde kullanılan algoritma gösterilmektedir.




**Şekil 5.15** Paket ayrıştırma işleminin algoritması

STS uygulamasında ağ dinleme katmanı, "STS İzleme Ayarları" menüsü altından seçilebilir. Bu seçim ile yakalanan paketlerin içerikleri de değişecektir. Uygulamada ağı, Katman-2 seviyesinde dinlemek için *DotNetWinPcap* Kütüphanesi kullanılmıştır. Bu şekilde seçime göre ağ dinleme katmanı değişmektedir. Şekil 5.15'te Katman-2 ve Katman-3 seviyesinde dinlenen paketlerin ne şekilde ayrıştırıldıkları gösterilmektedir.

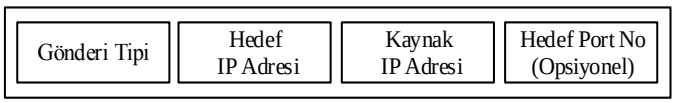
Dinlenen paketler "STS Monitör Ayarları" menüsünde ayarları yapılarak kütüklerin tutulması ve animasyon gösterim işlemleri gerçekleştirilir. "STS Monitör Ayarları" menüsünün içeriği Şekil 5.16'da gösterilmektedir.




**Şekil 5.16** STS uygulamasının "STS Monitör Ayarları" menüsü

"STS Monitör Ayarları" menüsünden monitör uygulamasının IP adresi ve port numarası girilerek *başlat* fonksiyonuyla tuzak sistemlerden yakalanan paketler, belirli bir formata getirilerek monitör uygulamasına gönderilir. Aynı zamanda bu menüden monitör uygulamasına gidecek olan paketlerin tipleri ve hedef IP adresleri de seçilebilir. Dinlenen paketlerin loglama işlemi için gereken ayarlar da bu menüde yer almaktadır. Tutulan loglar, sistem güvenliği hakkında detaylı bilgiler vererek sistemdeki zafiyetlerin bulunmasını kolaylaştırabileceği için, yetkisiz kişilerin eline geçmesi büyük bir risk oluşturmaktadır. Tutulan logların değiştirilmemiş olması da bilgi güvenliği temel ilkelerini sağlaması açısından önemlidir. Bunun için tutulan loglar, zaman damgası ile şifrelenerek güvenli bir loglama sistemi kullanılmıştır. Bu sistem, Bölüm 6'da anlatılacaktır.

STS uygulaması, yakaladığı paketleri monitör uygulamasına gönderirken performansı ve güvenliği arttırabilmek için paketleri belirli bir formata getirir. Bu format belirlenirken monitör uygulamasının animasyonu oluştururken kullanacağı minimum parametre sayısı belirlenmiştir. Buna göre dönüşüm yapılan format Şekil 5.17'de gösterilmektedir.

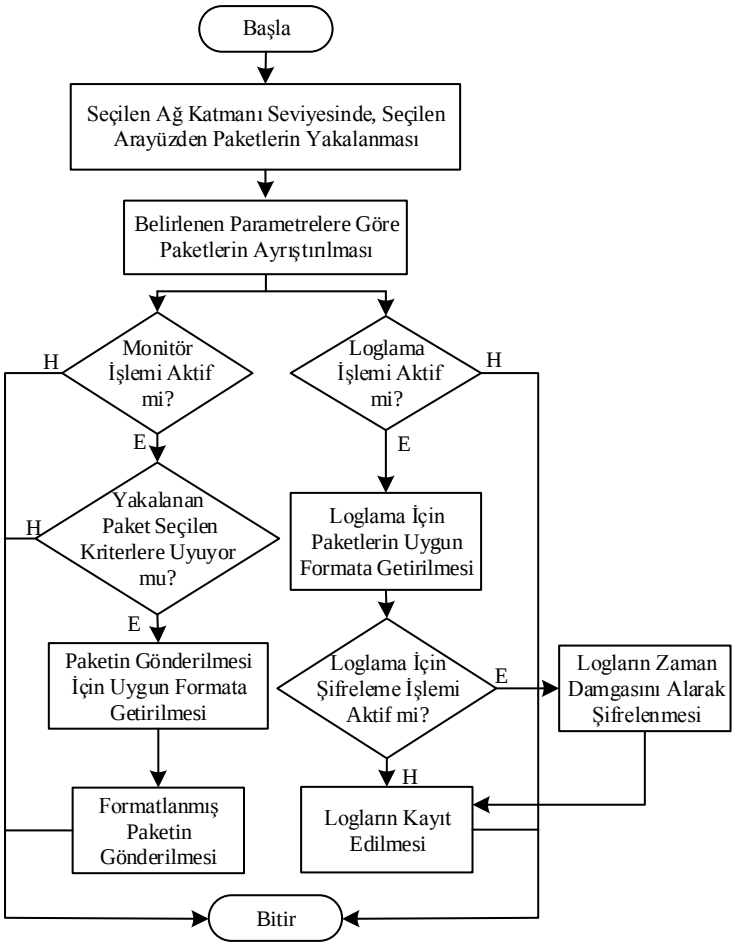



**Şekil 5.17** STS uygulamasının monitör uygulamasına gönderdiği paket formatı

Şekil 5.17'de gösterilen paket formatında; *Gönderi Tipi* alanı, paketin bir honeypot paketi mi yoksa saldırı tespit sisteminin yakaladığı bir paket mi olduğunu belirtir. *Hedef IP* ve *Kaynak IP* adresleri, paketin hangi konumdan hangi konuma gittiğini göstermek için kullanılan parametre alanlarıdır. *Hedef port no,* paketin hangi servise gönderildiğinin belirlenmesi için kullanılmıştır. Fakat port numarası bilgisi yalnızca TCP ve UDP paketlerinde bulunmaktadır. Diğer paket türlerinde bu alan kullanılmaz. Bu yüzden opsiyonel olarak belirtilmiştir.

Loglama ve paketlerin monitör uygulamasına gönderilme işlemi, performans kayıplarını önlemek için paralel olarak gerçekleştirilmektedir. STS uygulamasının bu işlem için kullandığı algoritma Şekil 5.18'de gösterilmektedir.

Loglama işlemi için özgün bir yazılım geliştirilerek log güvenliği garanti altına alınmıştır. Log dosyalarının değiştirilmesine karşın geliştirilen bu yazılım ile zaman damgaları kullanılarak RAM ve HDD üzerinde eş zamanlı kontroller ile log dosyalarının değiştirilip değiştirilmediği tespit edilmektedir. Şekil 5.16'da görülen "CrptoLog Ayarları" ve "Saldırı Tespiti Log Ayarları" kısımları bu uygulamayı temsil etmektedir. Bilgi sistemleri ve güvenliği açısından log tutma ve tutulan log dosyalarının güvenli ve doğru bir biçimde saklanması, yetkisiz kişiler tarafından değiştirilmediğinin garanti edilmesi önemli bir olgudur. Burada yapılan çalışma ile log kayıtlarının doğrulanması için kullanılan karşılaştırma işlemi, paralel programlama kullanılarak maksimum verim alınması amaçlanmıştır.




**Şekil 5.18** STS uygulamasının monitör uygulamasına paket gönderme ve loglama sistemi algoritması

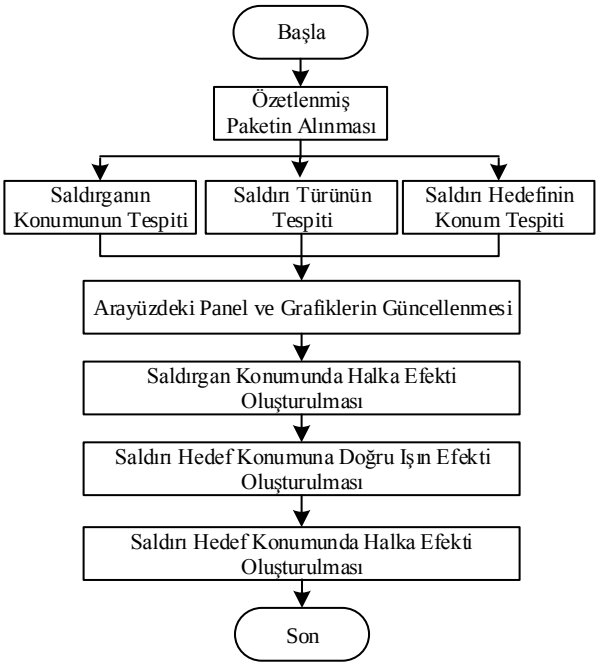
Kullanıcıların sistemle interaktif olarak iletişimde bulunduğu tek uygulama STS uygulamasıdır. Bunun için bu uygulamanın nasıl kullanılacağını anlatan "Yardım" menüsü eklenmiştir. Bu menüde yazılım ve diğer menüler hakkındaki bilgiler yer almaktadır.

### 5.4.3 Monitör Modülü

STS modülü ile yakalanan paketler monitör uygulaması ile görselleştirilerek, sistemde gerçekleşen kötü niyetli faaliyetler gerçek zamanlı olarak animasyonlar şeklinde gösterilmektedir. Monitör uygulaması, animasyonların gerçekleştirileceği tek bir arayüze sahiptir. Kullanıcı ile interaktif olarak iletişimde bulunmamaktadır. Monitör modülü, STS uygulamasının gönderdiği paketleri dinleyerek sahip olduğu konfigürasyon sayesinde hangi saldırının hangi konumda gerçekleştiğini belirleyerek animasyonları başlatmaktadır. STS uygulaması ile monitör uygulaması arasında gerçekleştirilen iletişimde kullanılan paket formatı Şekil 5.17'de verilmiştir. Bu paket formatında monitör uygulaması, *Kaynak IP* ve *Hedef IP* alanlarını kullanarak paketin hangi alt ağdan (subnet) hangi alt ağa gönderildiğini tespit edebilmektedir. Konfigürasyonunda bulunan alt ağların konum bilgisine ulaşarak bu konumlar arasındaki trafiği animasyon olarak göstermektedir. Paket formatında; *Hedef Port No* alanı mevcut ise bu portun hangi servise hizmet verdiğini bularak, başlatılan animasyondaki ışının

rengini ilgili servise ait bir saldırı olduğunu göstermek için yazılımsal olarak belirlenen bir renkte gösterir. Bu şekilde animasyonu izleyen herhangi bir kişi saldırının nereden nereye ve hangi servise yapıldığı hakkında bilgi sahibi olabilmektedir.

Çalışma prensibi gereği, STS modülü ile monitör modülü sürekli iletişim halinde olmak zorundadır. İletişim ve kurulum maliyetinin önüne geçmek için STS ve monitör modülü aynı fiziksel makine üzerinde çalıştırılabilir. Bu durumda monitör uygulamasını kullanmak isteyen kişinin sabit bir konumda olması gerekmektedir. Bu olumsuzluğun önüne geçebilmek için monitör uygulaması, web tabanlı teknolojilerle uyumlu bir teknoloji olan Java Applet ile geliştirilmiştir. Monitör uygulaması, STS uygulamasından gelen paketleri gerçek zamanlı olarak görselleştirmesi gerektiğinden aralarında gerçekleştirilen iletişimdeki gecikme minimum seviyede olmalıdır. Bu sebeple bu iletişim fazında TCP protokolünden daha hızlı bir protokol olan UDP protokolü kullanılmıştır. Monitör uygulaması, çalıştırıldığı andan itibaren STS uygulamasından gelecek olan paketleri beklemeye başlar. Monitör uygulamasının STS uygulamasından aldığı paketleri animasyona dönüştürmek için kullandığı algoritma şeması Şekil 5.19'da gösterilmektedir.




**Şekil 5.19** Monitör uygulamasının animasyon oluşturma algoritma diyagramı

Gerçekleştirilen uygulamalar, kritik düzeyde öneme sahip bilgi sistemlerini barındıran ağları koruduğu için, STS uygulamasından monitör uygulamasına gelen paketlerin eş zamanlı olarak gösterilmesi önem arz etmektedir. Herhangi bir kötücül aktivite durumunda yakalanan saldırı örüntülerinin gerçek zamanlı olarak tespiti yapılıp gerekli önlemler anlık olarak alınmaktadır. Buradaki eş zamanlı iletişim, ağ trafiğinin yoğunluğuna göre bir darboğaz oluşturabilmektedir.

Bu durumda uygulamada oluşabilecek darboğazları önleyebilmek için *multithreading* yapısı kullanılmıştır. Yakalanan her paket yeni oluşturulan bir *thread* ile animasyon haline getirilmektedir. Bu şekilde oluşabilecek çakışmaların ve performans kayıplarının da önüne geçilmesi planlanmıştır. Animasyonların oluşturulmasında kullanılan multithread yapısına rağmen yapılan performans ve yük testlerinde uygulamanın tepki süresinin azaldığı gözlemlenmiştir. Yük testleri ile uygulamanın maksimum performans limitleri belirlenerek, tepki süresinin azalmasına karşı gerekli önlemler alınmıştır. Uygulamanın anlık performansının izlenebilmesi için arayüze sistem yükü göstergesi eklenmiştir. Monitör uygulamasının Fırat Üniversitesi'ndeki fakülteler arası alt ağlarına göre modellenmesi ve anlık bir çalışma görüntüsü Şekil 5.20'de gösterilmektedir.

Şekil 5.20'de görüldüğü gibi, geliştirilen arayüz ekranının 1 numaralı göstergesi ile uygulamanın, kapasitesinin yüzde kaçını kullandığı izlenebilmektedir. 2 numaralı panelde, saldırıya maruz kalmış hedef noktalar, saldırı sayısına göre büyükten küçüğe sıralanmaktadır. 3 numaralı panelde ise saldırı gerçekleştirmiş noktalar, gerçekleştirilen saldırı sayısına göre büyükten küçüğe sıralanmaktadır. 4 numaralı panelde animasyonlarda kullanılan ışınların renklerinin hangi servise ait olduğu gösterilmekte ve saldırı sayısına göre büyükten küçüğe sıralanmaktadır. 5 numaralı panelde, gerçek zamanlı olarak yakalanan saldırıların detaylı bilgileri listelenmektedir. Bu liste gerçek zamanlı olarak trafiğe göre akmaktadır. 6 numaralı panelde, gerçekleştirilen saldırıların servislerin çeşitlerine göre renklendirilerek grafiğinin oluşturulması sağlanmıştır. 7 numaralı panelde kurumun krokisi yer almaktadır. Yerleştirilen küreler ile alt ağların konumları gösterilmektedir. Kürelerin renkleri alt ağın hangi ağ bölgesinde olduğunu göstermektedir. Buna göre mavi küre LAN bölgesini, yeşil küre DMZ bölgesini, turuncu küre ise dış bölgeyi (İnternet bölgesi) temsil etmektedir. Böylece bir arayüz ile tüm ağ bölgelerinde gerçekleşebilecek kötü niyetli faaliyetlerin de gösterilmesi sağlanmıştır.

**Şekil 5.20** Monitör uygulaması arayüzü

Saldırıların görselleştirilmesinde kullanılan animasyon, kaynak kürenin etrafında saldırıda kullanılan servisin renginde büyüyen bir halka oluşması ile başlar. Yönü hedef küreye dönük olarak ilerleyen bir ışının hedef küre üzerinde küçülen bir halka oluşturması ile sonlanır. Eğer gelen paket saldırı olarak tespit edilmiş ise hedef küre üzerindeki halka çeyrek daire şeklinde ve kırmızı renkte gösterilir. Eğer gelen paket, honeypot sunucu tarafından yakalanmış ise hedef küre üzerinde beyaz renkte tam bir halka oluşur. Bu şekilde yakalanan paketlerin animasyon üzerinde yakalama yöntemleri de görselleştirilmiş olur.

## 5.5 Sonuç ve Değerlendirme

Bu çalışmada, ağ güvenliğinde kullanılabilecek gerçek zamanlı saldırı tespit ve engelleme sistemleri için honeypot temelli bir yaklaşım önerilmektedir. Önerilen özgün yaklaşım için etkin bir yazılım geliştirilmiştir. Geliştirilen sistem, düşük ve yüksek etkileşimli honeypotların üstün özelliklerini alarak tek bir yapıda birleştiren, başarımı daha yüksek, hibrit bir honeypot sistemi olma özelliğine sahiptir. Geliştirilen sistem, gerçek zamanlı olarak bir kampüs ağı üzerinde test edilmiş ve başarılı sonuçlar gözlenmiştir. Bu kapsamlı çalışmada, honeypotların kurumsal ağlarda kullanımı incelenerek kurulum, bakım ve yönetim maliyetini düşürmek için sanallaştırma teknolojileri kullanılmıştır. Honeypot sunucularda oluşan ağ trafikleri, animasyon şeklinde görsel olarak izlenebilmektedir. Böylece, sistem durumu hakkında daha kolay bilgi sahibi olma imkânı sağlanmıştır.

Kurumsal ağlar ve honeypot tasarımları incelendiğinde, VLAN konfigürasyonunun kullanıldığı ağ sistemlerinde her VLAN için en azından farklı bir ağ arayüzüne sahip bir makine kullanılması gerekliliği görülmüştür. Her VLAN için gerçek bir makine kullanılması, özellikle kampüs ağlarını da içeren geniş kurumsal ağlarda honeypotların tüm ağda uygulanması, kurulum ve bakım maliyetlerini arttırmaktadır. Bu nedenle, büyük ölçekli kurumsal ağlarda kurulum, bakım ve yönetim maliyetlerini azaltmak için bir ağ arayüzü ile honeypotların tüm ağda faaliyet göstermesini sağlayan merkezi bir honeypot sunucu uygulaması geliştirilmiştir. VLAN içeren kurumsal ağlarda, katman-2 ve katman-3'ü dinleyebilen özgün bir *yazılımsal anahtar* tasarımı yapılarak VLAN ağlarının da dinlenebilmesi sağlanmıştır. Yazılımsal switch tasarımı Bölüm 7'de verilmiştir.

Gerçekleştirilen STS uygulaması, honeypot sunucuları dinlemenin yanı sıra, ağa bağlanarak saldırı engelleme sistemi olarak da kullanılabilmektedir. Bunun için imza tabanlı saldırı tespit yöntemleri sisteme entegre edilebilmektedir. Geliştirilen STS uygulamasında açık kaynak kodlu SNORT saldırı tespit sisteminin paket yakalama (Sniffing) modülü baz alınarak kötüye kullanım tespiti yapabilen uygulama gerçekleştirilmiştir. STS uygulaması yakaladığı paketlerin

imza tabanındaki saldırılarla uyuşması durumunda paketin iletimini keserek, ağı saldırılardan koruyabilmektedir.

İmza tabanlı saldırı tespit yöntemlerinin dezavantajlarından olan yeni saldırı örüntülerinin (sıfır gün -zero day) tespit edilememesi, honeypot sunuculardan toplanan log kayıtlarının analiziyle minimum seviyeye getirilmiştir. Honeypot sunuculardan toplanan log kayıtları, çoğunlukla saldırı içeriğine sahip olduğu için log kayıtlarından imzalar üretilerek yeni saldırı örüntülerinin de imza tabanına eklenmesi sağlanabilir. Honeypot sunuculardan toplanan logların anormallik tespiti yöntemleri ile saldırı olmayan log kayıtlarından ayıklanması, başarımı arttırmıştır. Honeypot sunuculardan toplanan saldırı log kayıtları ile anormallik tespiti yöntemlerinin dezavantajlarından olan yüksek *yanlış alarm seviyesi* oldukça azaltılmış ve başarımı daha yüksek hibrit bir saldırı tespit yöntemi elde edilmiştir.

STS uygulamasının saldırı engelleme sistemi olarak kullanılması durumunda tüm ağ trafiğinin üzerinden geçmesi söz konusu olmaktadır. Bu durum STS uygulamasının üzerinde çalıştığı donanımın özelliklerini zorlayabilir ve bir darboğaz oluşturabilir. Bu darboğaz ise tüm ağın performansını olumsuz olarak etkileyecektir. Bu sorunun bertaraf edilmesi için STS üzerinden geçen ağ trafiğinin donanım kısıtları tarafından darboğaza uğramayacağı bir gömülü sistem kullanılabilir. STS uygulaması bu gömülü sistem donanımı üzerinde çalışarak ağ performansında yaşanabilecek kayıplar da engellenmiş olacaktır.

Bu bölümde geliştirilen uygulamalar, kurumsal bir ağda bilgi güvenliği maliyetlerini azaltacak özellikte özgün bir honeypot temelli saldırı tespit ve engelleme sistemi olarak kurulup kullanılabilecek düzeydedir.

# 6. BİLİŞİM SİSTEMLERİNDEKİ LOG DOSYALARININ GÜVENLİĞİ İÇİN YENİ BİR YAKLAŞIM

## 6.1 Giriş

Log dosyaları, bir sistem üzerinde gerçekleşen aktivitelerin düz metin formatında saklandığı, bilgi güvenliği sistemlerinin vazgeçilmezi olan dosyalardır. Örneğin web sunucuları, istemcilerden gelen istekleri bir log dosyasında depolarlar. Kullanıcı isteklerinin saklandığı bu dosyalarda istemcilere ait IP adresleri, erişimin yapıldığı tarih-zaman bilgileri, erişilen web adresi bilgisi, kullanılan port numarası, durum kodları gibi önemli bilgiler yer almaktadır. Bu bilgiler, sistem yöneticileri için çok önemlidir. İlk zamanlarda log kayıtları, bir yığın düz yazı olarak değerlendirilmekte ve çoğunlukla bilgi barındırdığı sistemle ilgili bir problem olduğu zamanlarda kullanılmaktaydı. Fakat günümüzde bu dosyaların içerdiği bilgiler, daha detaylı incelemelerle daha etkin kullanılabilmektedir. Örneğin bir elektronik doküman yönetim sistemine, bir öğrenci bilgi sistemine, bir elektronik alışveriş sitesine ya da benzeri bir otomasyona yapılan saldırıların tespiti bu dosyaların içerdiği bilgiler sayesinde yapılabilmektedir. Log dosyalarındaki bilgiler, çeşitli analizlerle, web uygulamalarının tasarım ve kod optimizasyonları, hata oranının azaltılması, üyelerin ilgi analizleri, üyelere önerilerin sunulması, akıllı reklam yönetimi gibi birçok alanda kullanılmaktadır [93]. Ayrıca log dosyaları, günümüzde saldırı tespit ve engelleme sistemlerinde, adli bilişim analiz süreçlerinde sıklıkla başvurulan yapılardır.

İnternet bankacılığı hizmetini müşterilerine sunan bir banka, müşteri davranışları hakkında elde edilebilecek önemli özel bilgilerle ticari kazanımlarına katkı sağlayabilir. Erişim log kayıtları analiz edilerek, bu tarz hizmetlere ait birçok sayısal veri elde edilebilir. Elde edilen bu sayısal veriler değerlendirilerek sistem kullanıcılarıyla ilgili farklı bilgilere yönelme, kullanıcı davranışlarıyla ilgili tahminlerde bulunma ve benzeri araştırmalar yapma gibi konularda kolaylık sağlayarak, sunulan hizmetin kalitesi arttırılabilir.

Farklı sektörlerin İnternet kullanımını yaygınlaştırdığı bir gerçektir. Örnek olarak elektronik haberleşme, bilgi tarama, finans, elektronik ticaret vb. verilebilir. İnternet'e bağlı kişisel bilgisayarların kullanımının artması, bu sektörlerin hızla gelişmesini ve bunlara bağlı olarak alt sektörlerin oluşmasını sağlamıştır. Yatırım yapılan farklı sektörler, alt sektörlerin gelişimini ve istihdam oranını arttırmıştır. Bunun yanı sıra web tabanlı sistemlerin ve teknolojilerin karmaşıklığı da artmaktadır. Artan verilerin düzenli olarak saklanması ve analiz edilmesi, sistem yöneticileri ve firma sahipleri açısından artık bir gereksinim ve zorunluluk haline gelmiştir. Bu zorunluluk ve gereksinim bazı standartlar ve kanunlar tarafından da desteklenmiştir [93]. Log toplama konusuna kanun ve standartların verdiği öneme PCI veri

güvenliği standardı örnek verilebilir. PCI DSS (Payment Card Industry, Data Security Standarts), altı ana madde altında on iki gereksinimden söz etmektedir. Bunlardan biri de log toplama ile alakalıdır. Bunun yanı sıra T.C. 5651 sayılı kanun, FISMA (Federal Information Security Management Act), HIPAA (Health Insurance Portability and Accountability Act), SOX (Sarbanes-Oxley), ISO 17799 gibi ismi sıkça duyulan kanun, düzenleme ve standartlar da log tutma konusunu zorunlu tutmuştur [93]. Ülkemizde güncel yazılım ihalelerinde de artık bu standart ve kanunlara uyularak, üretilen yazılımların kendi log yapısına sahip olmaları zorunlu tutulmaktadır.

Log dosyalarının güvenliği de ayrıca üzerinde durulması gereken bir olgudur. Tutulan log dosyalarının üzerinde yetkisiz kişiler tarafından herhangi bir değişiklik yapılması veya yetkili kişilerin görevlerini kötüye kullanmaları neticesinde olabilecek olumsuz sonuçların da önüne geçilmesi, buna benzer durumların tespit edilmesi önemlidir. Bu bölümde, log dosyalarının yetkisiz erişimler tarafından değiştirilip değiştirilmediği tespit edilerek log dosyalarının güvenliğinin sağlanmasına yönelik olarak bir yazılım geliştirilmiştir.

## 6.2 Bilgi Güvenliği ve Log Toplama Sistemleri

Bilişim dünyasında, güvenlik olmazsa olmaz kriterlerden biri haline gelmiştir. Bu hususta bilgi güvenliği unsurlarının sağlanması önem teşkil etmektedir. Bütünlük, gizlilik ve erişilebilirlik bilgi güvenliğinin temel unsurlarıdır. Bilgi güvenliği unsurlarını bilişim sistemlerinde gözetebilmek için loglama sistemleri kullanılır. Bu şekilde kayıt edilen logların incelenmesiyle bilgi güvenliği unsurlarının sağlandığı teyit edilebilir. Bunun için devlet tarafından alınması gereken önlemler yasal zorunluluklar haline getirilmiştir. 5651 sayılı kanun ile loglama sistemlerinin kullanımı zorunlu hale gelmiştir. Bu şekilde işlenebilecek bilişim suçlarının adli bilişim sürecinde kullanılabilecek kanıtları oluşturulur. Bunun için kayıt altında tutulan logların güvenliğinin sağlanması önemli bir husustur. Logların güvenliği de bilgi güvenliği unsurları ile sağlanmalıdır. Bunun için logların bütünlüğünün, gizliliğinin ve erişilebilirliğinin teyit edilebilir olması gereklidir.

Mevcut ihtiyaçları karşılamak için geliştirilmiş olan loglama sistemleri iki grupta incelenebilir. Bunlar sadece entegre olarak çalışabildikleri donanımlardan log toplayabilen yazılımlar ve esnek yapısıyla donanım bağımsız olarak log toplayabilen yazılımlardır. Entegre olarak çalışabildikleri donanımın tüm özelliklerini kullanabildikleri için bu loglama sistemleri daha kapsamlı analizler yapabilir. Donanım bağımsız olarak çalışan loglama sistemleri ise esnek olabilmek için bu konudan kabul edilebilir bir derecede ödün vermişlerdir.

Mevcut loglama sistemlerinin daha çok üzerinde durduğu bilgi güvenliği unsuru bütünlüktür. Toplanılan logların değiştirilmediğinin kanıtlanması ve inkâr edilemez olması gerekmektedir. Bunun için hash algoritmaları ve zaman damgası kullanılmaktadır. Her saniye toplanan loglar hashleri alınarak zaman damgası ile damgalanır. Zaman damgasında kullanılacak sertifika 3. taraflar aracılığı ile sağlanarak güvenliğin tarafsız olması amaçlanır.

Toplanan logların 3. şahısların eline geçmemesi, logların gizliliği konusunda önem teşkil etmektedir. Logların gizliliğinin sağlanmaması, saldırganın loglara erişebilmesini ve kötü niyetli olarak eriştiği sisteme dair daha fazla bilgi toplamasını, dolayısıyla sistemin açıklarının tespit edilmesine yol açabilmektedir.

Toplanan logların hashleri zaman damgası ile damgalanarak logların bütünlüğünün sağlanabileceği düşünülse de logların kullanılan kayıt biriminden silinmesi durumunda saldırganın kendini gizlemesi yine de mümkündür. Her saniye başına yapılan, hash ve zaman damgası işlemlerine rağmen, logların değiştirilmesi bir saniyenin altında gerçekleştirilebilen bir işlemdir. Tüm bunlar göz önüne alındığında logların bütünlüğünün objektif olarak sağlandığı düşünülemez.

Gizliliği sağlanan logların incelenebilmesi ve doğrulanabilmesi için yetki sistemi kullanılmalıdır. Yalnızca sistem tarafından yetkiye sahip yönetici logları inceleyebilmeli ve doğrulayabilmelidir. Bu şekilde toplanan loglar, bilgi güvenliğinin üç temel unsurunu da sağlayarak adli bilişim sürecinde objektif birer kanıt olarak kullanılabilir olma niteliği kazanır.

5651 sayılı kanuna göre siber ortamda yer sağlayıcısı (Sunucu bulundurma, kiralama, web sitesi barındırma hizmeti veren tüm kişi ya da kurumlar) olarak hizmet veren firmalar trafik bilgilerini uygun formatlarda altı ay saklamakla yükümlü tutulmuşlardır. Erişim sağlayıcı (İnternet'e erişme hizmeti veren tüm kurumlar, operatörler, servis sağlayıcılar vb) olarak hizmet veren firmalar ise trafik bilgisini 1 yıl boyunca saklamakla yükümlüdürler [93].

### 6.3 Geliştirilen Log Güvenlik Sistemi

Sistem temel olarak iki modülden oluşmaktadır. Bunlar *Time Stamp Tool* ve *Listening Tool* modülleridir. Sunuculardaki trafik akışı, log dosyaları ile kayıt altına alınırken ilgili portlar kullanılmaktadır. Bu portlardaki bilgiler tarih bilgisiyle birlikte tutularak log kayıtları oluşturulur.

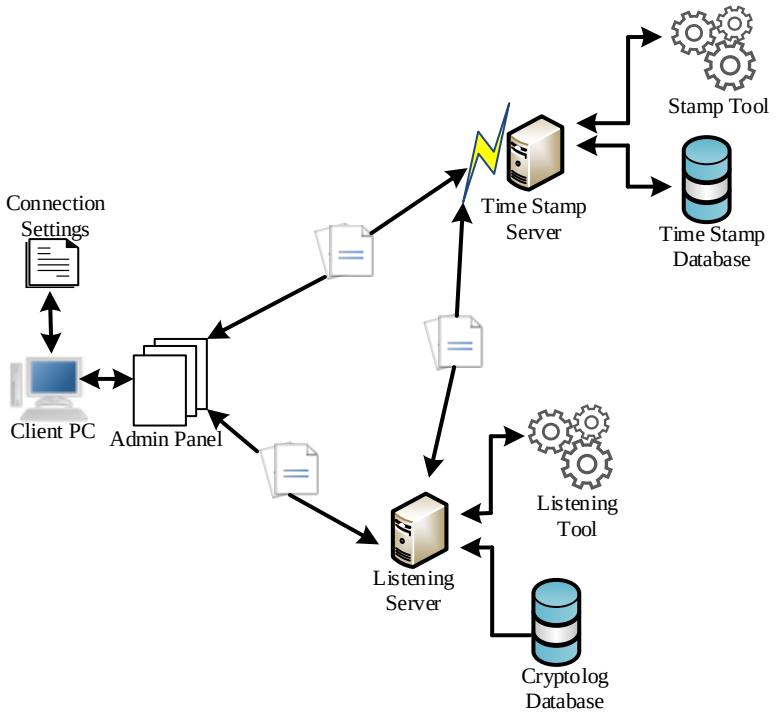
Dinleme aracı (Listening Tool), portları okuyarak log tutmaya yarar. Bu işlemi *Listening Server* üzerinden gerçekleştirir. *Cryptolog Database* ise *Listening Tool* modülünde oluşturulan logları gün sonunda rapor halinde kaydeder. Ayrıca Admin Paneli giriş bilgilerini ve log kayıtlarını da tutar. Gün sonunda oluşan log verilerini kontrol ettikten sonra damgalamak için

*Time Stamp Server'*a gönderir. *Stamp Tool*, Listening Server'dan gelen log verilerini damgalayarak *Time Stamp Database'*e kaydeder.

Admin Panel, Connection Settings dosyasından Listening Server ve Time Stamp Server bağlantı bilgilerini alır. Eğer bağlantı bilgileri yok ise, Admin Paneli'nden bağlantı kurulmasını bekler. Bağlantı kurulduktan sonra Listening Server üzerinde bulunan Cryptolog Database'deki giriş bilgileri ile sisteme giriş yapılır. Bağlantı verilerini de Connection Settings dosyasına şifrelenmiş olarak kaydeder ve daha sonraki bağlantılarda bağlantı kurulumuna ihtiyaç duymaz. Daha önce Listening Tool tarafından oluşturulan Listening Server log kayıtlarını ve Admin Panel'inin log verilerini okuyabilir.

Admin Panel kullanıcısı, Cryptolog Database'de oluşturulan log verilerinin Time Stamp Database'de damgalandıktan sonra oluşan verilerin değişikliğe uğrayıp uğramadığını kontrol edebilme haklarına sahiptir. Admin Panel, kontrol amaçlı kodu gönderir, Time Stamp Tool da Time Stamp Database'deki bilgilerle karşılaştırıp Admin Panel'e kontrol sonucunu yollar. Kodun doğruluğuna göre true/false değer dönderir.

Sistemde bulunan modüller ve bu modüllerin birbiri ile iletişimi Şekil 6.1'de verilmiştir.




**Şekil 6.1** Cryptolog çalışma diyagramı

### 6.3.1 Dinleme Aracı

Bu çalışmada web portu kullanarak elde edilen güncel veriler bir kriptolojik yöntem kullanılarak şifrelenecek ve HDD üzerine kayıt edilecektir. Bu şekilde sisteme sızan kişinin log dosyasından kendiyle alakalı kısımları bulması engellenmiş olacak ve verilerin gizliliği

sağlanacaktır. Log dosyasında değişiklik yapılması veya silinmesi durumunda verilerin bütünlüğünü sağlamak için farklı bir kriptolojik yöntem kullanılarak veriler RAM üzerinde de tutulacaktır. RAM'de tutulan şifrelenmiş veriler RAM'in işletim sisteminde güvenli kipte çalışmasından dolayı güvende olacaklardır.

Günün sonunda (belirlenen süre sonunda) RAM'de ve HDD'de tutulmuş olan, şifrelenmiş log kayıtları deşifrelenerek bloklar şeklinde karşılaştırılacaktır. Bu şekilde orijinal log dosyasında değişiklik yapılıp yapılmadığı ispatlanarak verilerin doğruluğu sağlanacaktır. Eğer RAM'den elde edilen veriler ile HDD'deki log dosyasından elde edilen veriler arasında uyuşmazlık var ise bu durum raporlanarak bildirilecektir. Verilerin uyuşması sonucunda ise deşifrelenmiş log dosyası zaman damgası yöntemiyle damgalanarak log dosyalarının tutulduğu sunucuya gönderilecek ve raporlanarak bir sorun oluşmadığı bildirilecektir. Bu şekilde log dosyasının bir günlük yaşam döngüsü sona erecektir. Şekil 6.2 dinleme aracının çalışma mantığını sunmaktadır.

Dinleme aracı (Listening Tool), portları okuyarak log tutmaya yarayan gelişmiş bir araçtır. Bunu Listening Server üzerinden gerçekleştirir. Admin Panel, Connection Settings dosyasından Listening Server bağlantı bilgilerini aldıktan sonra Listening Server, port trafik akışını dinleyerek anlık olarak bir veri oluşturur. Admin Paneli, log kayıtlarını tutar. Gün sonunda oluşan log verilerini kontrol ettikten sonra oluşan veri Admin Panel'e iletilir. Listening Tool tarafından oluşturulan Listening Server log kayıtlarını ve Admin Panel'in log verilerini okuyabilir. Bu anlık log alışverişi sayesinde Admin panelden istenilen log kayıt görüntülerine anında ulaşılabilir.

### 6.3.2 Damgalama Aracı

Damgalama aracı (Stamp Tool) sistemi, bekleme halindedir ve komut gelmesi durumunda komutları işleyen bir yapısı vardır. Sisteme iki türlü komut gelmektedir. Bu sebeple iki farklı algoritma mantığı ile çalışabilmektedir. Bu komutlardan birincisi, Listening Tool'dan gelen damgalama komutudur. İkincisi ise Admin Paneli'nden gelen damgalama kontrol komutudur.

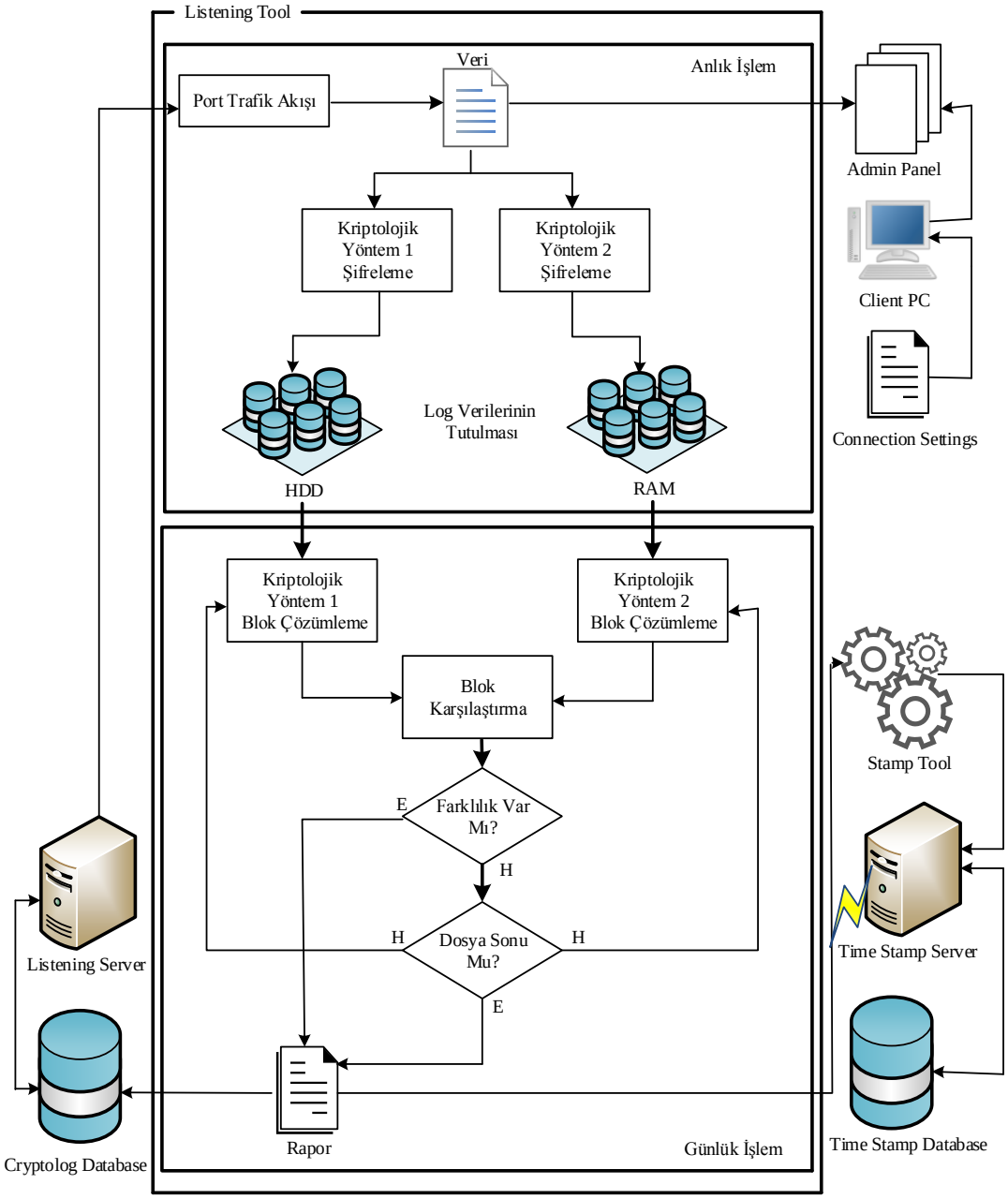
Listening Tool' dan veri geldiğinde damgalama algoritması devreye girer. Listening Tool'da oluşan veriler gün sonunda veri tabanına raporlanarak kaydedilir. Kaydedilen her bir raporun kendine özgü belirleyici bir ID'si oluşturulur. Raporu yazdıktan sonra raporun ID'sini ve Listening Tool'da bulunan damgalama parolasını Stamp Tool'a gönderir.

Stamp Tool, Listening Tool'dan gelen ID'yi kullanarak Cryptolog Database'den raporun oluşturulma tarihini ve rapor içeriğini alır. Daha sonra Stamp Tool içerisinde barındırdığı RSA parametresini kullanarak veriyi damgalar. Son olarak, raporun benzersiz olan ID'sini ve

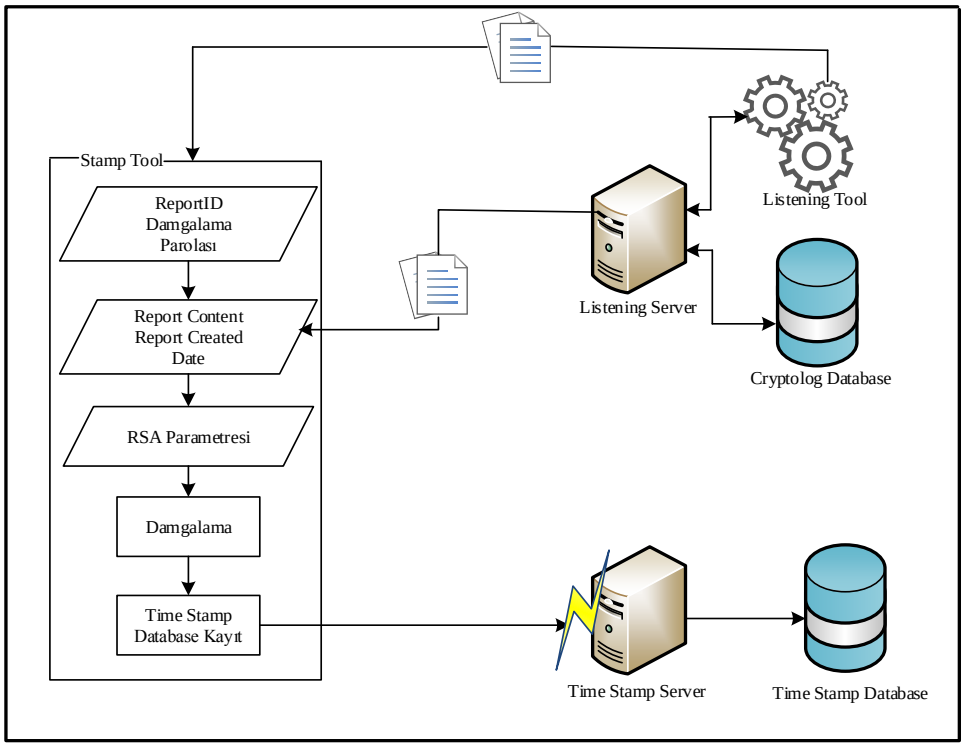
damgalama sonucu oluşan hash kodunu Time Stamp Database'e kaydeder. Bu damgalama işleminin akış şeması Şekil 6.3'te verilmiştir.

Yönetici panelinde damgalama kontrolü yapılmak istenildiğinde seçilen raporun benzersiz ID'si ve damgalama parolası Stamp Tool'a gönderilir.

Stamp Tool, bu ID'yi kullanarak Cryptolog Database'den rapor içeriğini ve raporun oluşturulma tarihini okur. Stamp Tool'da bulunan RSA parametresi ile damgalayarak yeni bir hash üretir. Daha sonra Time Stamp Database'de daha önceden damgalanmış olan hash dosyasını okur ve iki dosyayı karşılaştırır. Oluşan sonucu (True/False) Admin Panel'ine gönderir.




**Şekil 6.2** Dinleme aracı çalışma diyagramı




**Şekil 6.3** Damgalama aracı algoritması

### 6.3.3 Yönetici Paneli ve Özellikler

Kullanıcı programı (Admin Panel) ilk çalıştırdığında Connection Settings dosyasından verileri alır ve Time Stamp Server ve Listening Server'a bağlantı kurmayı dener. Bağlantı kurulmadığı takdirde, kullanıcıya bağlantı ayarlarını yapması için bağlantı yapılandırma penceresi açılır. Bağlantı gerçekleştirildikten sonra bağlantı verilerini şifreleyip Connection Settings dosyasına kaydedip giriş paneline yönelir. Daha sonraki bağlantı için verileri bu dosyadan okur ve kullanıcıya her seferinde bağlantı kurma işlemini tekrarlatmaz. Admin Panel, Şekil 6.4'te gösterilmektedir.

Sisteme giriş için kullanıcı adı ve şifresi doğru girildiği takdirde sistemde kayıtlı olan birincil seçenekli mail adresine tek kullanımlık bir şifre gönderilir. Daha sonra gelen pencerede tek kullanımlık şifre girilir, şifre doğru girildiği takdirde mail panele yönlendirilir. Bu kısımda yer alan bazı özellikler aşağıda açıklanmıştır.

*Mail Panel*, daha önce oluşturulmuş raporların listesi ve diğer işlemleri yapabilmek için yönlendirme nesnelerini içeren bir yapıdadır.

*Kişisel Veriler,* sistemdeki kullanıcının (BOSS/Admin) giriş bilgileri ve iletişim bilgilerini (Mail adresleri/Telefon numaraları/Öncelikli iletişim adresleri) güncellemek için kullanılan pencerelerdir.

*Kullanıcı Ekleme,* sisteme giriş yapabilecek kullanıcıların eklenmesini sağlayan pencerelerin bulunduğu kısımdır. Bu kısımda işlem yapma yetkisi sadece yönetici (boss) tipi kullanıcılara aittir.

*Panel Log,* Admin Panel'de yapılan her hareket bir log verisi olarak veri tabanına kaydedilmektedir. Bu kayıtları görüntülemeyi sağlayan penceredir. Bu kısımda işlem yapma yetkisi sadece yönetici tipi kullanıcılara aittir.

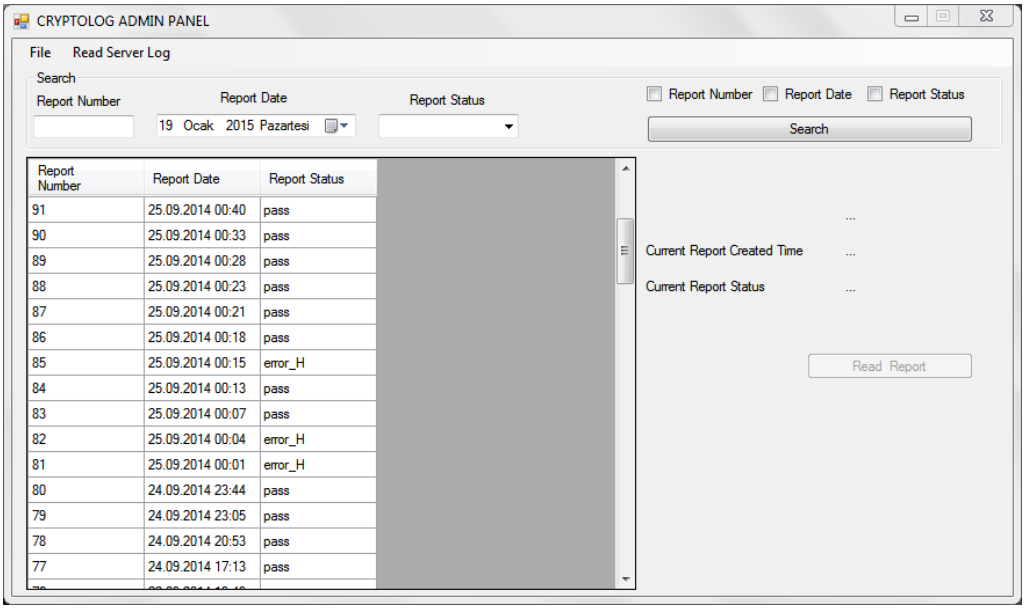
*Rapor Görüntüleme,* Listening Tool'un oluşturduğu raporu görüntülemeyi sağlar. Bu kısımda işlem yapma yetkisi kullanıcının yetkilendirilmesi ile belirlenir.

*Rapor Kaydet,* görüntülenen raporun Excel dosyasına aktarılmasını sağlar. Bu kısımda işlem yapma yetkisi kullanıcının yetkilendirilmesi ile belirlenir.

*Rapor Görüntüleme Geçmişi,* görüntülenen raporun daha önce hangi kullanıcı tarafından hangi işlemin yapıldığını gösteren penceredir. Bu kısımda işlem yapma yetkisi kullanıcının yetkilendirilmesi ile belirlenir.

*Time Stamp Checking,* Time Stamp Server'da bulunan Time Stamp Database'de kayıtlı olan görüntülenen raporun hash kodunun veri kaybına uğrayıp uğramadığının veya veri bütünlüğünün korunup korunmadığını kontrol etmek için damgalama parolası ile kullanılır. Bu adımda damgalama parolası ve rapora ait olan eşsiz ID bilgisini Stamp Tool'a gönderir ve cevap alır.

*Connection LIB,* panelle ilgili tüm verilerin bulunduğu Listening Server'da bulunan Cryptolog Database ile bağlantıyı ve veri alışverişini gerçekleştiren kütüphane dosyasıdır.




**Şekil 6.4** Cryptolog yönetici paneli

### 6.3.4 Log Formatı

Uygulamadaki veri akışı, sunucunun web portundan elde edilen verilerin *"ip adresi-kullanıcı adı-işlem zamanı-erişim isteği-sonuç durumu kodu-aktarılan bayt miktarı-URL yönlendirme-kullanıcı aracısı"* formatına getirilmesiyle başlar. Bu bilgiler log dosyalarında sıklıkla kullanılan bilgileri barındırır. Daha sonra elde edilen veri AES-128 ve 3DES şifreleme algoritmaları kullanılarak şifrelenir. Elde edilen şifreli log kayıtları, HDD üzerinde bir log dosyasına ve RAM üzerinde oluşturulan bir alana yazılır. Günün sonunda HDD'deki ve RAM'deki şifrelenmiş log kayıtları deşifrelenerek karşılaştırılır. Herhangi bir uyuşmazlığın olmaması durumunda log kayıtları TSA istemcisine gönderilerek zaman damgası ile damgalanarak saklanır ve bir sorun oluşmadığı raporlanarak bildirilir. Eğer herhangi bir uyuşmazlık oluşursa bu hata, log dosyalarının değiştirilmiş olduğu bilgisi ve orijinal halleriyle birlikte raporlanarak ilgili yere iletilecektir. Şekil 6.5, kullanılan log kayıtları içerisinde hangi bilgilerin bulunduğunu göstermektedir.

| IP Versiyonu | Kaynak IP | Hedef IP | Protokol Tipi | Kaynak Port | Hedef Port | Veri Boyutu | İşlem Zamanı |
|---|---|---|---|---|---|---|---|

**Şekil 6.5** Log formatı

*IP Versiyonu*, sistemde kullanılan IP'nin uyarlanma formatını gösterir (IPv4, IPv6).

*Kaynak IP*, IP adresi İnternet'i ya da TCP/IP protokolünü kullanan diğer paket anahtarlamalı ağlara bağlı cihazların, ağ üzerinden birbirleri ile veri alışverişi yapmak için kullandıkları adrestir. Kaynak IP, isteğin geldiği bilgisayarın IP numarasıdır.

*Hedef IP,* istek yapan bilgisayarın ulaşmak istediği IP numarasıdır.

*Protokol Tipi,* iletişimde kullanılan protokol türüdür (TCP, UDP).

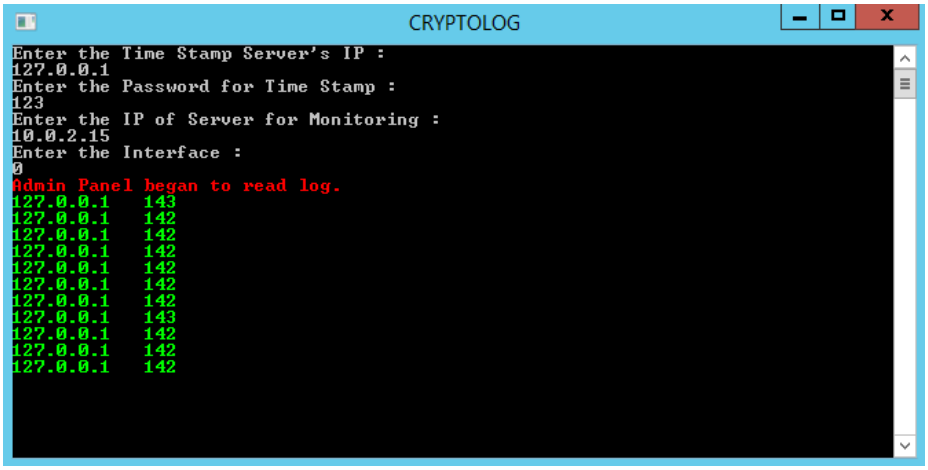
*Kaynak Port,* verinin alındığı portu belirtir

*Hedef Port,* verinin gönderildiği portu belirtir.

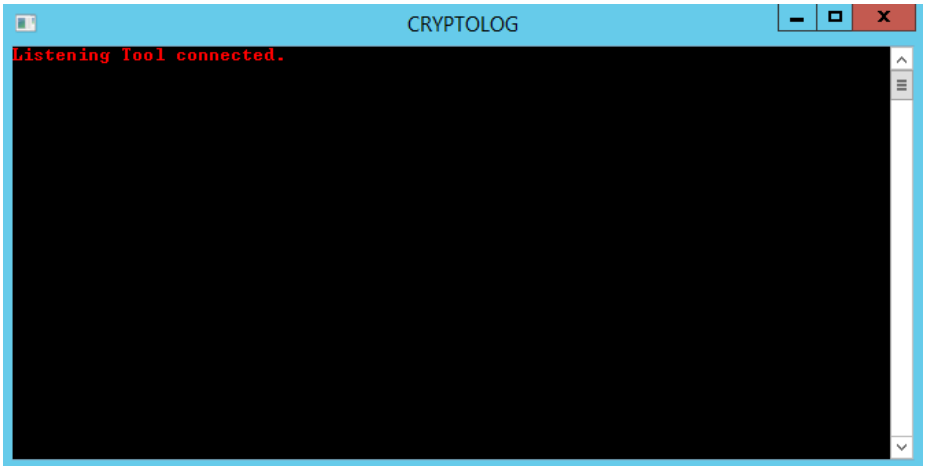
*Veri Boyutu,* istemcinin, sunucuda işlemi gerçekleştirmesi sırasında yapılan veri alışverişinin boyutunu tutar.

*İşlem Zamanı,* işlemin yapılma zamanıdır.

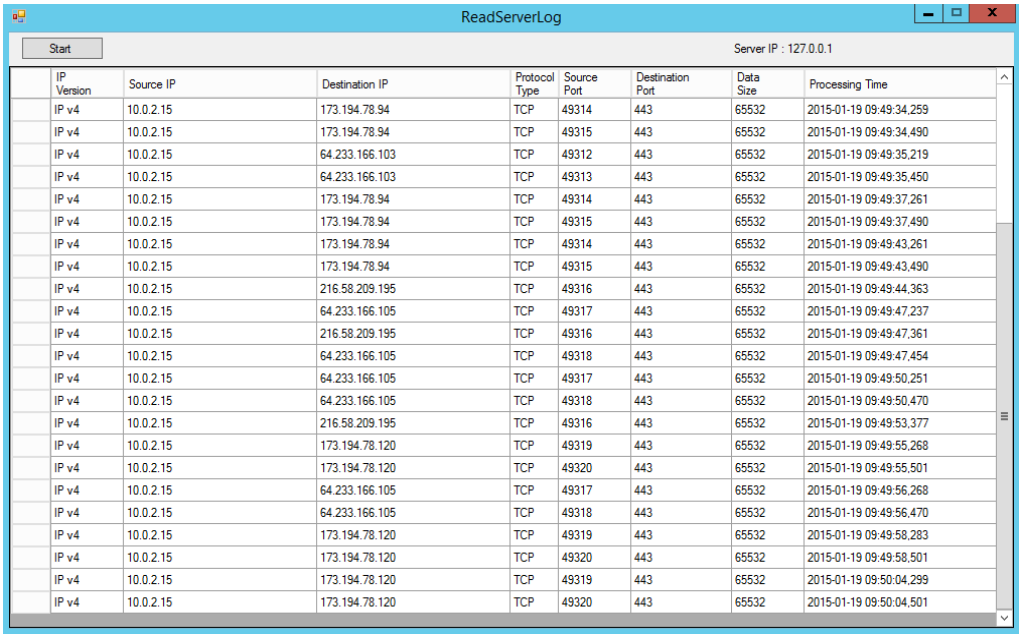
Şekil 6.6, önerilen sistemin başlatılmasını göstermektedir. Zaman damgası sunucusu IP numarasının, zaman damgası için şifrenin ve izlenecek sunucu IP numarasının girilmesi ile sisteme bağlanılır. Şekil 6.7, Listenin Tool aracına bağlantı yapılmasını göstermektedir. Şekil 6.8 ise yakalanan ağ trafiğinin ilgili log formatındaki kayıt edilişini göstermektedir.




**Şekil 6.6** Önerilen log toplama sisteminin başlatılması




**Şekil 6.7** Dinleme aracının başlatılması ve ağ trafiğinin yakalanması




| IP Version | Source IP | Destination IP | Protocol Type | Source Port | Destination Port | Data Size | Processing Time |
|---|---|---|---|---|---|---|---|
| IP v4 | 10.0.2.15 | 173.194.78.94 | TCP | 49314 | 443 | 65532 | 2015-01-19 09:49:34,259 |
| IP v4 | 10.0.2.15 | 173.194.78.94 | TCP | 49315 | 443 | 65532 | 2015-01-19 09:49:34,490 |
| IP v4 | 10.0.2.15 | 64.233.166.103 | TCP | 49312 | 443 | 65532 | 2015-01-19 09:49:35,219 |
| IP v4 | 10.0.2.15 | 64.233.166.103 | TCP | 49313 | 443 | 65532 | 2015-01-19 09:49:35,450 |
| IP v4 | 10.0.2.15 | 173.194.78.94 | TCP | 49314 | 443 | 65532 | 2015-01-19 09:49:37,261 |
| IP v4 | 10.0.2.15 | 173.194.78.94 | TCP | 49315 | 443 | 65532 | 2015-01-19 09:49:37,490 |
| IP v4 | 10.0.2.15 | 173.194.78.94 | TCP | 49314 | 443 | 65532 | 2015-01-19 09:49:43,261 |
| IP v4 | 10.0.2.15 | 173.194.78.94 | TCP | 49315 | 443 | 65532 | 2015-01-19 09:49:43,490 |
| IP v4 | 10.0.2.15 | 216.58.209.195 | TCP | 49316 | 443 | 65532 | 2015-01-19 09:49:44,363 |
| IP v4 | 10.0.2.15 | 64.233.166.105 | TCP | 49317 | 443 | 65532 | 2015-01-19 09:49:47,237 |
| IP v4 | 10.0.2.15 | 216.58.209.195 | TCP | 49316 | 443 | 65532 | 2015-01-19 09:49:47,361 |
| IP v4 | 10.0.2.15 | 64.233.166.105 | TCP | 49318 | 443 | 65532 | 2015-01-19 09:49:47,454 |
| IP v4 | 10.0.2.15 | 64.233.166.105 | TCP | 49317 | 443 | 65532 | 2015-01-19 09:49:50,251 |
| IP v4 | 10.0.2.15 | 64.233.166.105 | TCP | 49318 | 443 | 65532 | 2015-01-19 09:49:50,470 |
| IP v4 | 10.0.2.15 | 216.58.209.195 | TCP | 49316 | 443 | 65532 | 2015-01-19 09:49:53,377 |
| IP v4 | 10.0.2.15 | 173.194.78.120 | TCP | 49319 | 443 | 65532 | 2015-01-19 09:49:55,268 |
| IP v4 | 10.0.2.15 | 173.194.78.120 | TCP | 49320 | 443 | 65532 | 2015-01-19 09:49:55,501 |
| IP v4 | 10.0.2.15 | 64.233.166.105 | TCP | 49317 | 443 | 65532 | 2015-01-19 09:49:56,268 |
| IP v4 | 10.0.2.15 | 64.233.166.105 | TCP | 49318 | 443 | 65532 | 2015-01-19 09:49:56,470 |
| IP v4 | 10.0.2.15 | 173.194.78.120 | TCP | 49319 | 443 | 65532 | 2015-01-19 09:49:58,283 |
| IP v4 | 10.0.2.15 | 173.194.78.120 | TCP | 49320 | 443 | 65532 | 2015-01-19 09:49:58,501 |
| IP v4 | 10.0.2.15 | 173.194.78.120 | TCP | 49319 | 443 | 65532 | 2015-01-19 09:50:04,299 |
| IP v4 | 10.0.2.15 | 173.194.78.120 | TCP | 49320 | 443 | 65532 | 2015-01-19 09:50:04,501 |

**Şekil 6.8** Yakalanan ağ trafiğinin listelenmesi

## 6.4 Sonuç ve Değerlendirme

Bu çalışmada bilişim sistemleri ve ağ güvenliği açısından çok önemli bir yer teşkil eden log tutma ve tutulan log dosyalarının güvenli ve doğru bir biçimde saklanması, yetkisiz kişiler tarafından değiştirilmediğinin garanti edilmesi üzerinde durulmuştur. Yapılan çalışmada, log dosyalarının kaydedilmesinde bilgi güvenlik unsurlarını sağlayarak, adli bilişim sürecinde loglara objektif kanıt niteliği kazandırılmıştır.

Çalışmada log kayıtlarının doğrulanması için kullanılan karşılaştırma işlemi belirlenen zaman periyotları ile yapılmaktadır. Fakat belirlenen zaman periyodu boyunca sistemin toplayacağı log dosyasının boyutu tahmin edilemeyebilir. Paralel olarak log kayıtlarının RAM'de tutulması bellek taşması gibi sorunlara neden olabilir. Bu sorunların önüne geçebilmek için zaman periyotları yerine bellek sınırı kullanılabilir. Bu şekilde uygulama yalnızca kendine tahsis edilen bellek üzerinde çalışarak sistem yükünü azaltacak ve ayrılan bellek kapasitesini doldurduğunda log dosyalarının doğrulanması için karşılaştırma işlemini gerçekleştirecektir. Karşılaştırma işleminden sonra tekrar tahsis edilmiş RAM bloğu üzerinde çalışmaya devam edecektir.

Log kayıtlarının doğrulanması için kullanılan karşılaştırma işleminde paralel programlama kullanılarak, maksimum verim alınması amaçlanmıştır. Fakat ikili arama algoritması ile log kayıtlarının hash değerlerinin karşılaştırılarak var olan farkların

bulunmasıyla yapılacak karşılaştırma sayısı en iyi durum için 1'e en kötü durum için ise n/2'ye düşecektir. Bu şekilde karşılaştırma işleminin maliyeti yarıya indirilebilir.

Son olarak, çalışmada ağ katmanından IP protokolü ve taşıma katmanından UDP ve TCP protokollerinden log toplanmıştır. Uygulamaya diğer protokoller de tanıtılarak daha kapsamlı bir şekilde log toplanması sağlanabilir.

Yapılan çalışma, olabilecek adli olaylarda kurumsal ve kişisel anlamda logların değişmediğinin ve delil olarak kullanılacak dijital bilgilerin güvenlik ve bütünlüğünün korunduğunu garanti eden bir yapı sunmaktadır.

# 7. YAZILIMSAL ANAHTAR TASARIMI VE MERKEZİ KONTROL YAKLAŞIMI

## 7.1 Giriş

Son zamanlarda bilgi güvenliği çalışmalarında honeypot sistemlerin aktif olarak saldırı tespit ve engelleme sistemleri ile birlikte kullanıldığı önceki bölümlerde belirtilmişti. Honeypot sistemler, Şekil 7.1'de gösterildiği gibi *saldırı çekme*, *veri toplama* ve *veri analizi* bileşenlerinden oluşmaktadırlar [16]. Honeypotun etkileşim seviyesi saldırı çekme bileşeninin türünü belirler. Honeypot sistemler, ağ üzerinde genellikle LAN bölgesinde konumlandırılmaktadır. Böylece honeypot, ağ üzerindeki tüm bölgelerde faaliyet gösterebilmektedir. Bu bölgede honeypotun ele geçirilmesi çok büyük güvenlik risklerine neden olacağı için saldırı çekme bileşeni düşük etkileşimli olan bir honeypot sistem kullanılması bu sorunun önüne geçebilecektir. Honeypotun veri toplama ve veri analizi bileşenleri yüksek etkileşimli sistemlerdeki gibi harici bir yazılım ile sağlanarak daha esnek izleme ve detaylı analiz imkânı sağlanmış olur.

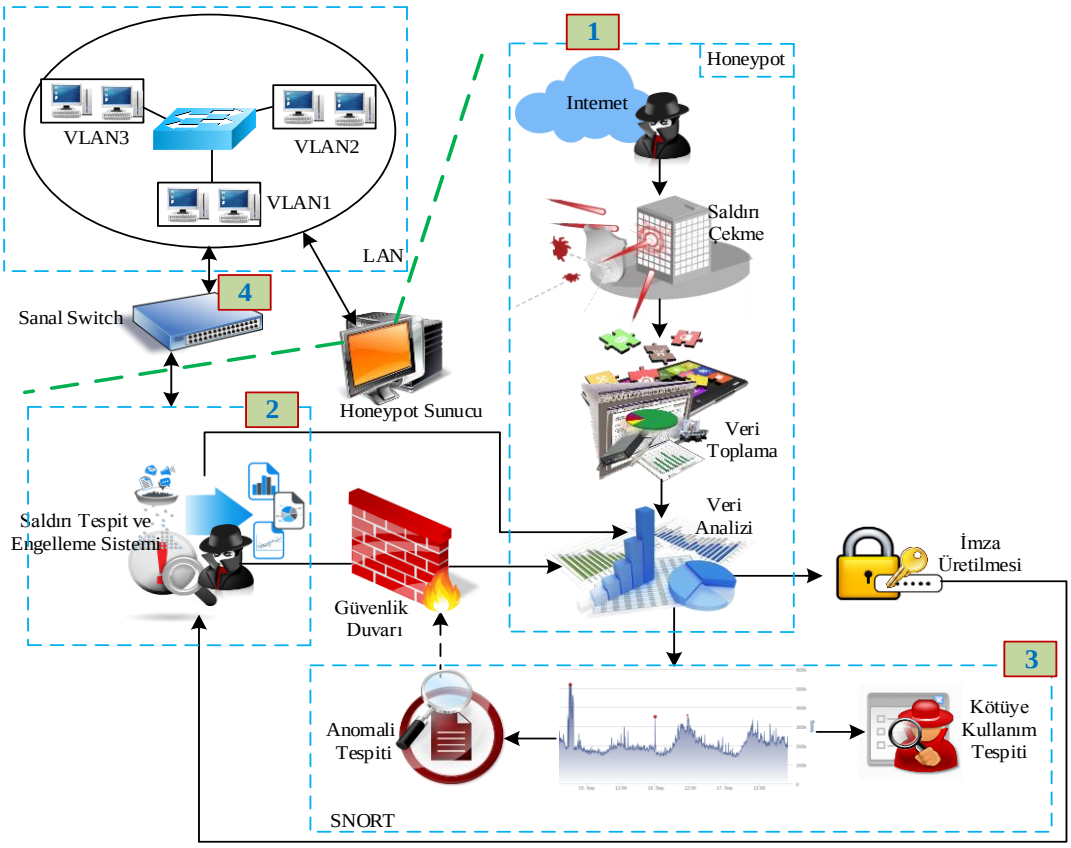
Geliştirilen ve önerilen sistemin mimarisi, Şekil 7.1'de görüldüğü gibi genel olarak 4 ana modülden oluşmaktadır. Bunlar honeypot sistem modülü (1), saldırı tespit ve engelleme modülü (2), anormallik tespiti ve kötüye kullanım tespit modülü (3) ve yazılımsal anahtar (soft-switch) (4) modülünden oluşmaktadır.

*Honeypot sistem modülü:* Yukarıda özellikleri anlatılan bu modül, Şekil 7.1'de görüldüğü gibi *saldırı çekme*, *veri toplama* ve *veri analizi* bileşenlerinden oluşmaktadır.

*Saldırı tespit ve engelleme modülü:* Bu modül, anahtar ve yönlendirici arasında konumlandırılarak ağın fiziksel olarak bölünmesi sağlanmıştır. Honeypot modülü ile VLAN iletişimi arasında kontrol görevini sağlamaktadır.

*Anormallik tespiti ve kötüye kullanım tespit modülü:* Bu modül, kötüye kullanım tespiti ve anormallik tespiti yöntemlerine göre çalışabilen STS yapılarının sisteme entegre edilebilmesi için bir arayüz niteliğinde hizmet sunan yazılımdır. Burada özel olarak SNORT adlı STS yapısının imza veri tabanı ve anormallik tespit yapısı kullanılmıştır.

*Yazılımsal anahtar modülü:* Fiziksel anahtarın *trunk* portu ile yönlendirici arasında çalışan bu modül, donanımsal anahtar gibi çalışabilen yazılımdır. Bu yazılım, ağdaki donanımsal anahtara bağlı tüm VLANların kontrol edilmesini sağlamaktadır. Böylece her bir VLAN içerisinin merkezi bir noktadan izlenmesi mümkün olmaktadır.




**Şekil 7.1** Honeypot sistemlerinin temel bileşenleri ve ağ güvenliğindeki kullanımı

Bu bölümde VLAN ağlarını da içeren büyük kurumsal ağlarda honeypotların konumlandırılırken kurulum ve bakım maliyetlerini düşürmek için yeni bir yöntem geliştirilmiştir. Bu yönteme göre VLAN ağlarının bulunduğu sistemlerde honeypotların tek bir sunucu üzerinden konumlandırılmasını sağlayabilmek için yazılımsal bir anahtar tasarlanmıştır. VLAN konfigürasyonu 2. ve 3. OSI katmanlarında çalışabilen anahtarlar üzerinde yapılabildiğinden, geliştirilen yazılımsal anahtar her iki katmanda da uygulanabilir olması açısından farklı senaryolar dahilinde incelenmiştir. Gerçekleştirilen tasarım ile 2. ve 3. katman anahtarlar üzerindeki VLAN ağlarının tüm alt ağlarında honeypot sistemleri tek bir honeypot sunucu ile temsil edilerek merkezi kontrol sağlanabilmiş ve böylelikle güvenlik seviyesi arttırılmıştır. Böylece honeypot sistemlerin avantajları, saldırı tespit sistemlerinin yetenekleri ile birleştirilmiş ve sıfır gün (zero-day) olarak adlandırılan saldırıların da yakalanabileceği bir ortam oluşturulmuştur.

Önceki bölümlerde, ağda tanımlı VLAN ağları bulunması durumunda, honeypot sistemlerin her VLAN içerisinde yapılandırılması gerektiği belirtilmişti. Ayrıca saldırı tespit

ve engelleme sistemlerinin tek başlarına yeterli olmadığı, tamamlayıcı sistemler olarak honeypotların önerildiğinden bahsedilmişti. Bu çalışma kapsamında geliştirilen yazılımsal anahtar sayesinde honeypot sistemler ile saldırı tespit ve engelleme sistemleri birbirini tamamlayıcı bir biçimde, ağ iletişiminde herhangi bir darboğaz oluşmadan kullanılabilmektedir. Geliştirilen yazılım ile merkezi bir kontrol sağlanarak kurulum ve bakım maliyetleri düşürülmüş ve VLAN içeren ağların kontrolü ve güvenliği gerçekleştirilebilmiştir. Ayrıca honeypot sistemler ile saldırı tespit sistemlerinin birlikte kullanılması ile bilinmeyen yeni saldırıların tespit edilebilmesi ve yanlış alarm seviyesinin düşürülmesi sağlanmıştır.

## 7.2 Önerilen Metot ve Yazılımsal Anahtar Uygulaması

Bu bölümde, gerçekleştirilen yazılımsal anahtar uygulaması üç kısımda tanıtılacaktır. VLAN konfigürasyonu OSI 2 ve 3 numaralı katmanlarda gerçekleştirilebildiği için hem 2.katman hem de 3.katman anahtar cihazları için hangi senaryoların geçerli olduğu açıklanmıştır. Farklı senaryolar kapsamında, merkezi olarak konumlandırılan bir STS uygulaması üzerinde gerçekleştirilen yazılımsal anahtar uygulamasının ağ iletişimini nasıl gerçekleştirdiği açıklanmıştır. Bu iletişimdeki akış şemaları ve ağ diyagramları detaylıca verilmiştir.

### 7.2.1 Katman-2 Anahtarlar Üzerinde Honeypotların Konumlandırılması

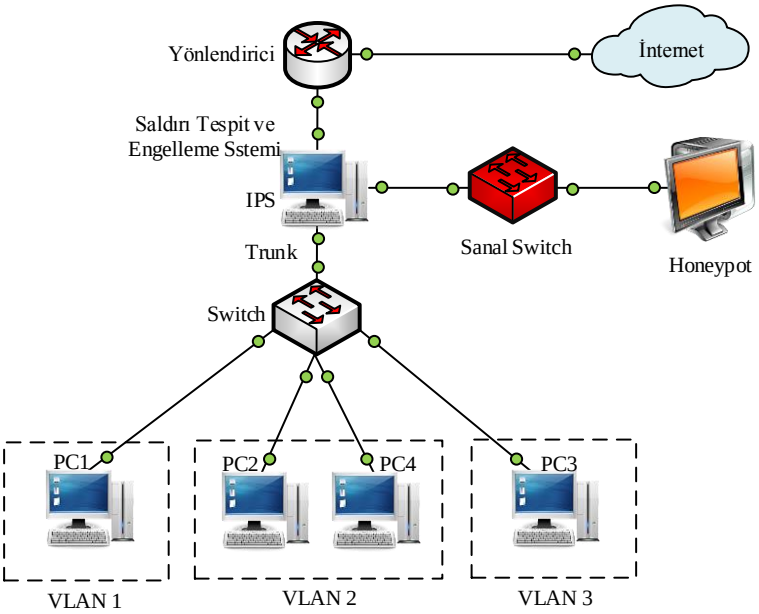
İkinci katman anahtarlar, portlarına gelen paketleri daha önceden oluşturulmuş MAC (Media Access Control) adres bilgilerinin bulunduğu tablolara bakarak diğer portlarına yönlendirir. Bazı durumlarda bir grup portun kendi içinde diğer portlardan yalıtılmasıyla fiziksel olarak ayrık bir ağ oluşturulmak istenebilir. Bu gibi durumlar için VLAN teknolojisi kullanılır. VLAN konfigürasyonunun bulunduğu bir anahtar, bir portundan gelen bir paketin hedef MAC adresinin başka bir VLAN'a ait olan farklı bir portunda olduğunu bilmesine rağmen bu paketi hedef porta göndermeyecektir. Bu şekilde VLAN'lar arası ağın tamamen yalıtımı sağlanmış olacaktır. Fakat bu şekilde oluşturulmuş VLAN'ların kendi aralarında iletişim kurmaları mümkün olmamaktadır. Eğer VLAN'ların da kendi aralarında haberleştirilmesi gerekiyorsa bunun için 3. katmanda çalışan yönlendiricilere ihtiyaç duyulmaktadır.

Yönlendiriciler, fiziksel olarak ayrılmış iki ağı birbiriyle haberleştirmeye yarayan ağ cihazlarıdır. Anahtarlar, üzerinde yapılandırılmış olan VLAN konfigürasyonuyla birbirinden

yalıtılmış ağları da birleştirebilirler. Bunun için yönlendiricilerin alt arayüzleri (sub interfaceleri) kullanılabilir. VLAN konfigürasyonunun yapıldığı anahtarlar üzerinde yönlendiriciye bağlanacak olan port *trunk* olarak ayarlanarak anahtara gelen paketlerin DOTQ1 protokolüyle kapsüllenerek yönlendiriciye gönderilmesi sağlanır. DOTQ1 protokolünün içinde VLAN bilgisi yer almaktadır. Yönlendirici bu kapsülü çıkartarak paketin yönlendirilmesi gereken VLAN bilgisiyle paketi tekrar kapsüller ve anahtara gönderir. Anahtar, DOTQ1 başlığındaki VLAN bilgisini okur ve DOTQ1 kapsülünü çıkartarak hedef MAC adresinin bulunduğu porta paketi gönderir. Bu şekilde VLAN'lar arası iletişim gerçekleşmiş olur.

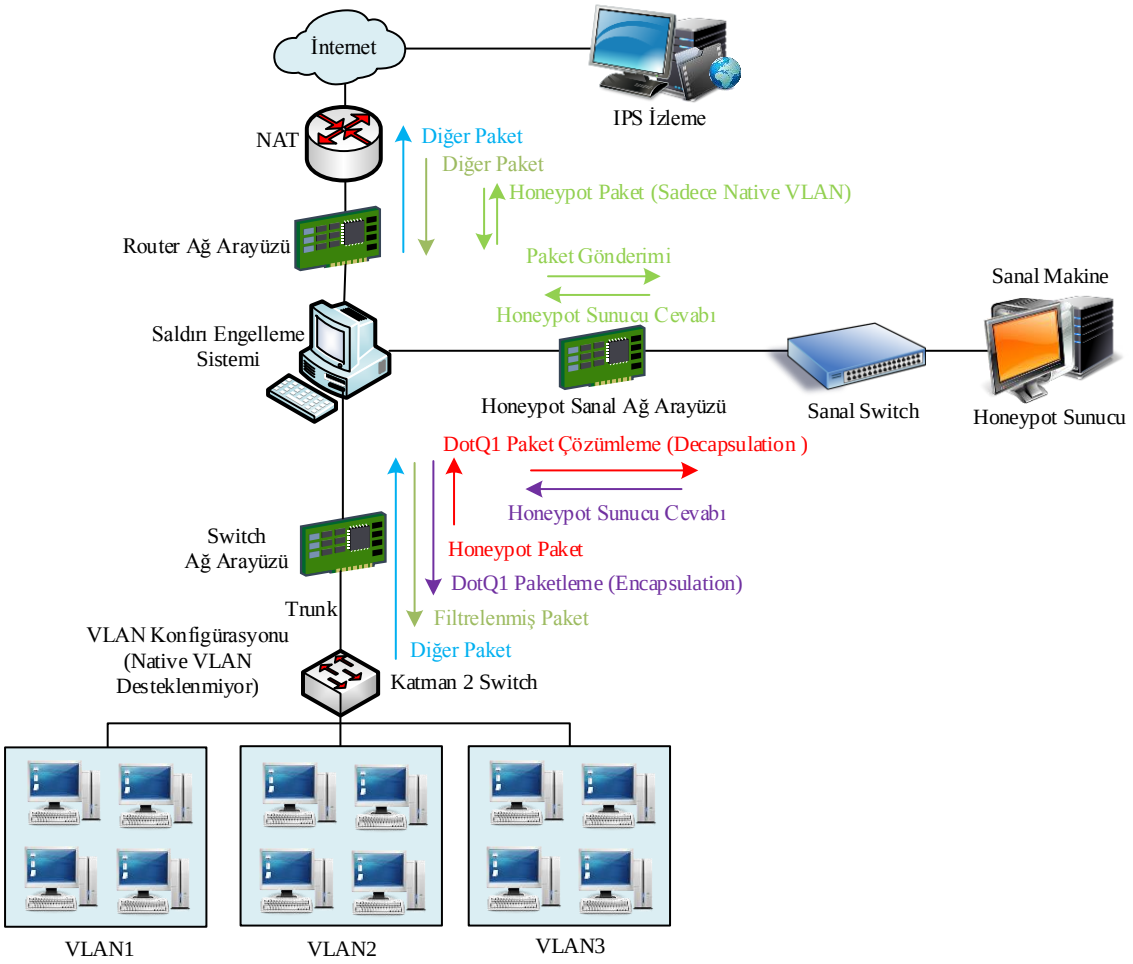
Honeypotların her VLAN'da faaliyet gösterebilmesi için her VLAN'da honeypotu temsil edecek bir sunucunun bulunması gereklidir. Bir VLAN içerisinde çalışan bir honeypot sunucunun kendi VLAN ağının alt ağı (subnet) içindeki IP adreslerini temsil ederek faaliyet göstermesi ve diğer VLAN'lar içindeki kullanıcıların bu honeypota erişmesi mümkündür. Fakat bulunduğu VLAN'ın alt ağı dışındaki bir IP adresini temsil etmesi mümkün değildir. Bunun için her VLAN üzerinde her VLAN'ın alt ağını temsil eden bir honeypot sunucu bulunması gerekmektedir.

Bu çalışmada anahtarın *trunk* olarak konfigüre edilmiş portu ile yönlendirici arasında çalışan bir yazılımsal anahtar ile honeypotun tüm VLAN'lara dahil edilmesi sağlanmıştır. Şekil 7.2'de GNS3 ağ emülatör uygulaması ile oluşturulup test edilmiş bir senaryo verilmiştir.




**Şekil 7.2** Katman-2 anahtar üzerinde STS'nin konumlandırılması

Şekil 7.2'de IPS bilgisayarının iki tane fiziksel ağ arayüzü bulunmaktadır. Sanal anahtara bağlanan arayüzü IPS üzerinde oluşturulmuş sanal bir arayüzdür ve honeypot sanal sunucusuyla iletişim kurmak için bu arayüzü kullanır. IPS, anahtar ve yönlendirici arasında konumlandırılarak ağın fiziksel olarak bölünmesi sağlanmıştır. Sanal ağ arayüzüyle bağlandığı sanal honeypot sunucusuna ilgili yönlendirmeleri 2. katman ve 3. katman protokollerini kullanarak yapmaktadır. Bu şekilde anahtar üzerinde bulunan tüm VLAN'ların alt ağlarına ait IP adreslerinin merkezi bir honeypot sunucu tarafından temsil edilmesi sağlanmıştır. Şekil 7.3'te, Şekil 7.2'deki senaryonun ağ akış diyagramı verilmiştir.




**Şekil 7.3** Katman-2 anahtar üzerinde STS'nin ağ akış diyagramı

Şekil 7.3'te, 2. katman anahtarlar üzerinde önerilen IPS sistem üzerinde çalışan yazılımsal anahtarın iletişimi nasıl sağladığı gösterilmektedir. Buna göre yazılımsal anahtar uygulamasının üç farklı arayüz için yönlendirme işlemi yapması gerekmektedir. Öncelikle anahtardan gelen paketlerin honeypot sunucusunda temsil edilen IP adreslerine denk gelip gelmediği kontrol edilir. Eğer gelen paket honeypot sunucuya ait ise paketi honeypota göndermek üzere bağlı olduğu sanal anahtara gönderir. Bu şekilde 3. katmanda yönlendirme işlemi yapmış olacaktır. Bunun yanında gelen paketler anahtar tarafından DOTQ1 protokolü ile kapsüllenerek gönderilmiştir. Bu kapsül içinden gelen paketin kaynak MAC adresi ve VLAN bilgisi bir tabloda tutularak kapsül atılmaktadır. Honeypot sunucuya gelen paketler DOTQ1 protokolü çıkartılmış olan paketlerdir. Bu şekilde dışarıdan gelecek olan *native* VLAN özelliğindeki paketler, honeypot sunucuya gönderilerek dış IP bölgelerinin temsil edilmesi de sağlanacaktır. Ama bu özelliğin kullanılabilmesi için anahtar üzerinde *native* olarak ayarlanmış bir VLAN bulunmaması gerekmektedir. Ağ yöneticilerinin *native*

VLAN'a dahil edilerek haricen ağı izlemeleri sağlanabilir. Eğer anahtardan gelen paketlerin IP adresleri honeypot sunucuya ait değil ise yine DOTQ1 başlığı içindeki VLAN bilgisi ve kaynak IP adresi tabloya eklenerek yönlendiriciye bağlı olan arayüze gönderilir.

Yönlendiriciden gelen paketlerin öncelikle honeypot sunucusunda temsil edilen IP adreslerine denk gelip gelmediği kontrol edilir. Eğer gelen paket honeypot sunucuya ait ise paketi honeypota göndermek üzere bağlı olduğu sanal anahtara gönderir. Dış IP adreslerinin honeypotlar tarafından temsil edilebilmesi için yönlendiriciler üzerinden portların honeypotlara yönlendirilmesi sağlanmalıdır ve yönlendirilmede honeypotun kullanacağı IP adresi *native* VLAN alt ağından seçilmelidir. Bu şekilde dışarıdan gelen paketler DOTQ1 protokolü ile kapsüllenmemiş olacaktır ve doğrudan honeypota gönderilebilecektir. Eğer paket honeypota ait değil ise *saldırı tespit ve engelleme sistemi* tarafından süzülerek anahtara gönderilecektir.

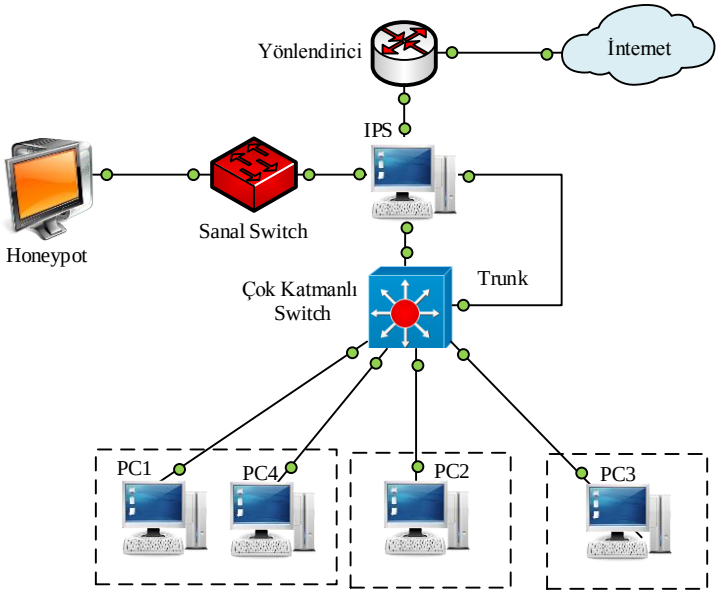
Honeypottan gelen paketlerin hedef MAC adresi yönlendiriciye ait ise paket doğrudan yönlendiriciye yönlendirilecektir. Bu şekilde 2. katmanda yönlendirme işlemi yapılmış olacaktır. Eğer hedef MAC adresi yönlendiriciye ait değil ise daha önceden oluşturulmuş MAC adresi ve VLAN bilgilerinin bulunduğu tabloya bakılarak paketin gitmesi gereken VLAN adresi bulunacak, kaynak MAC adresi yönlendiricinin MAC adresi ile değiştirilerek anahtara gönderilecektir. Bu şekilde anahtar, yönlendiricinin ilgili VLAN'a paket gönderdiğini varsayarak paketi istendiği gibi yönlendirecektir.

Bunların dışında 2. katman protokolleri kullanılarak *broadcast* adreslere paket gönderilmesi durumları için de istisnalar oluşmaktadır. Bu durumlar için de yeni senaryolar oluşturularak yazılımsal anahtar tasarımına eklenmiştir. Bu istisnaların kontrolü yazılımsal anahtar uygulamasının akış diyagramında incelenmiştir.

### 7.2.2 Katman-3 Anahtarlar Üzerinde Honeypotların Konumlandırılması

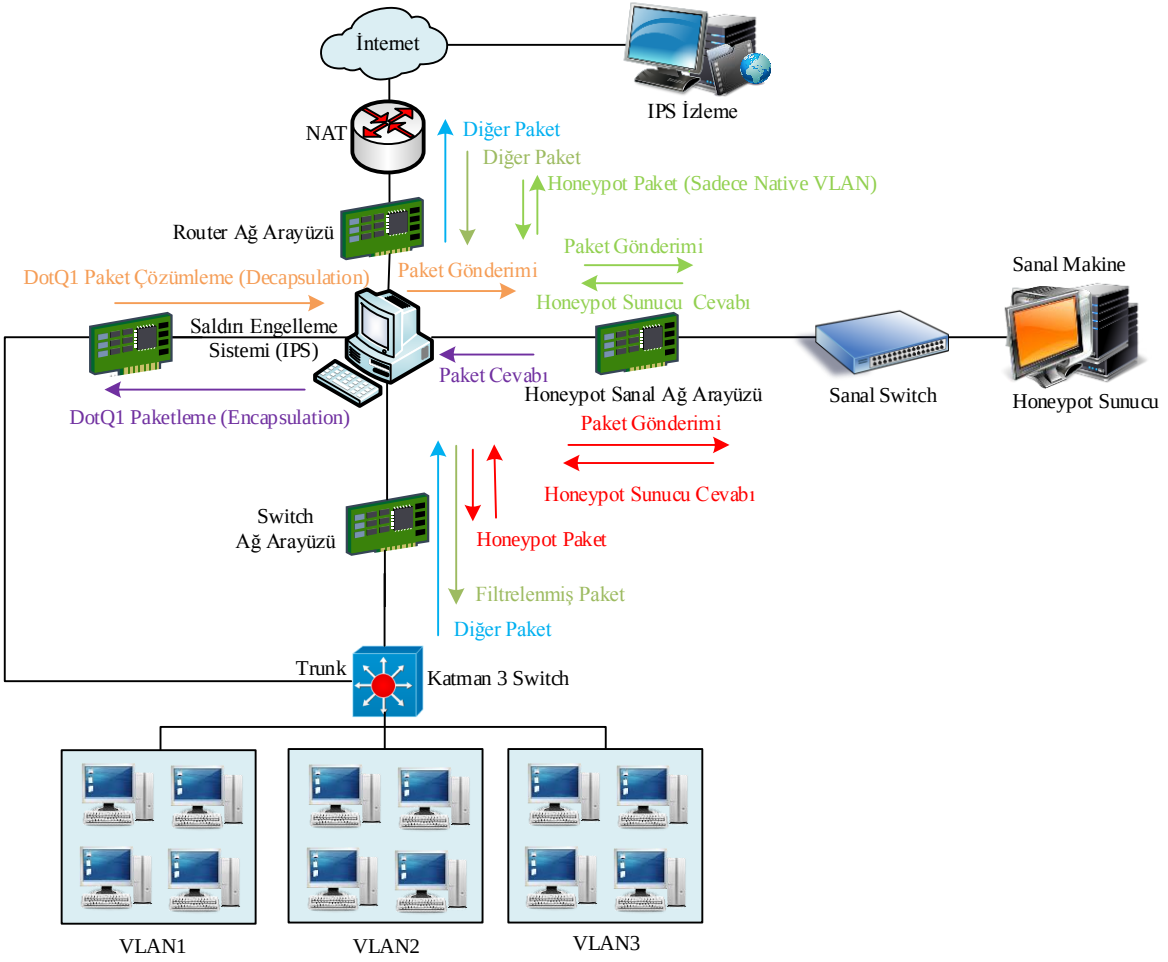
3. katman anahtarları, 2. Katman anahtarlarından farklı olarak portları arasında 3. Katman yönlendirme işlemi de yapabilir. Bu sayede oluşturulan VLAN'ları haberleştirmek için harici bir yönlendirici kullanımına gerek kalmamaktadır. Anahtarların portlarının *trunk* olarak konfigüre edilmesinin farklı bir amacı da farklı anahtarlar üzerindeki VLAN'ları birbirine bağlamaktır. Eğer bir anahtar yönlendirebildiği kendi VLAN'ları içinde hedef MAC adresine ait bir kayıt bulamazsa paketi DOTQ1 ile kapsülleyerek *trunk* olarak konfigüre edilmiş arayüzünden diğer anahtarlara gönderir. Bu şekilde birden fazla anahtar üzerindeki VLAN birbirine bağlanmış olur.

Honeypotların 3. katman anahtarları üzerine uygulanması da anahtarın farklı bir anahtar olduğunu düşündüğü yazılımsal anahtar ile iletişim kurması mantığına dayanır. Şekil 7.4'te GNS3 ağ emülatör uygulaması ile oluşturulmuş ve test edilmiş bir senaryo gösterilmektedir.




**Şekil 7.4** Katman-3 anahtar üzerinde STS'nin konumlandırılması

Şekil 7.4'te IPS bilgisayarının üç tane fiziksel ağ arayüzü bulunmaktadır. Sanal anahtara bağlanan arayüzü, IPS üzerinde oluşturulmuş sanal bir arayüzdür ve honeypot sanal sunucusuyla iletişim kurmak için bu arayüzü kullanır. IPS, anahtar ve yönlendirici arasında konumlandırılarak ağı fiziksel olarak bölmesi sağlanmıştır. Sanal ağ arayüzüyle bağlandığı sanal honeypot sunucusuna ilgili yönlendirmeleri 2. katman ve 3. katman protokollerini kullanarak yapmaktadır. Anahtarın *trunk* olarak konfigüre edilmiş portu ile anahtara ikinci defa bağlanarak anahtar üzerinde bulunan tüm VLAN'ların alt ağlarına ait IP adreslerinin merkezi bir honeypot sunucu tarafından temsil edilmesi sağlanmıştır. Şekil 7.5'te, Şekil 7.4'deki senaryonun ağ akış diyagramı verilmiştir.




**Şekil 7.5** Katman-3 anahtar üzerinde STS'nin ağ akış diyagramı

Şekil 7.5'te, 3. katman anahtarlar ile önerilen IPS sistem üzerinde çalışan yazılımsal anahtarın nasıl iletişim sağladığı gösterilmektedir. Buna göre yazılımsal anahtar uygulamasının dört farklı arayüz için yönlendirme işlemi yapması gerekmektedir.

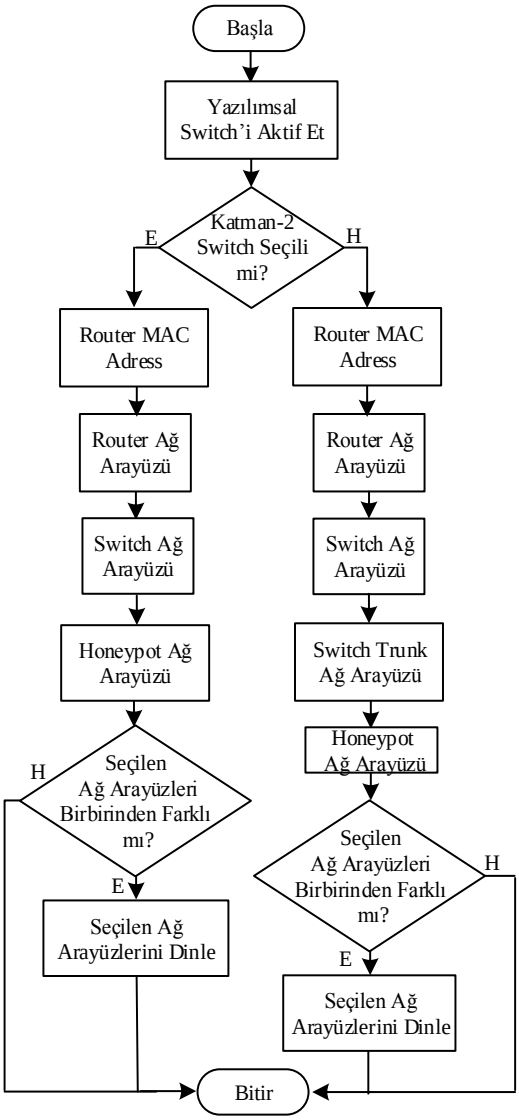
Anahtar ve yönlendiriciye bağlı olan arayüzlerde daha önce 2. katman anahtarları için tasarlanan algoritmanın aynısı kullanılmıştır. *Trunk* olarak ayarlanmış olan port, dış ağ ile iletişim için kullanılan porttan hariç olduğu için bu tasarımda *native* VLAN'lar da desteklenmektedir. Native VLAN'lar kırmızı oklar ile gösterilen yolu izlerken diğer VLAN'lar *trunk* olarak ayarlanmış porttan IPS'e ulaşacaktır. IPS üzerinde bulunan yazılımsal anahtar bir önceki uygulamadan farklı olarak anahtarın *trunk* olarak ayarlanmış portuna bağlı arayüzden gelen paketleri de kapsül çıkartma ve kapsülleme işlemlerinden geçirecektir.

### 7.2.3 Yazılımsal Anahtarın IPS Uygulama Arayüzüne Eklenmesi

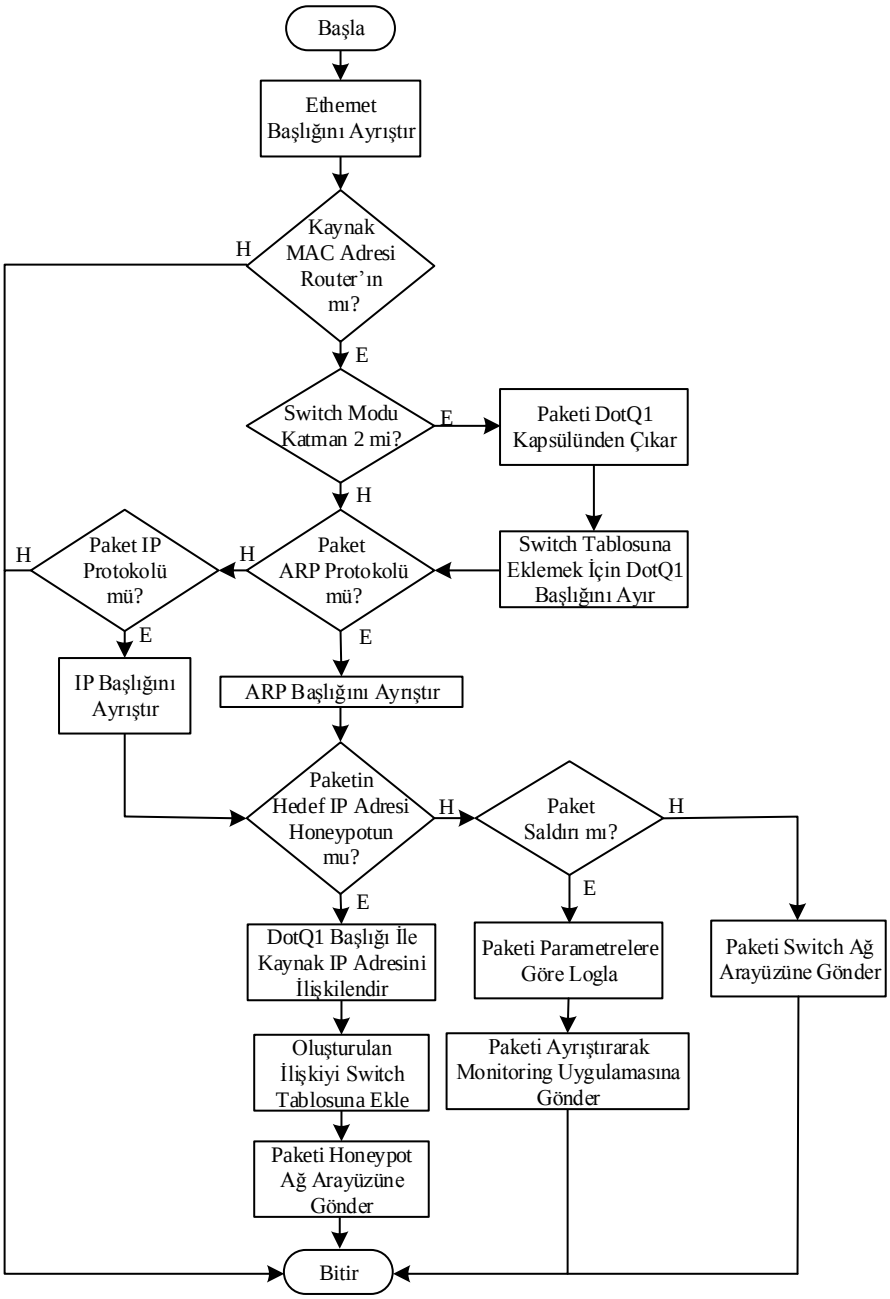
Şekil 7.6'da gösterildiği üzere arayüzden yönlendiricinin MAC adresinin girilmesi, ağda kullanılan anahtarın tipinin ve ilgili ağ arayüzlerin seçilmesi sade ve anlaşılır bir tasarım kullanılarak gerçekleştirilmektedir.


**Şekil 7.6** Yazılımsal anahtarın STS uygulamasına eklenmesi

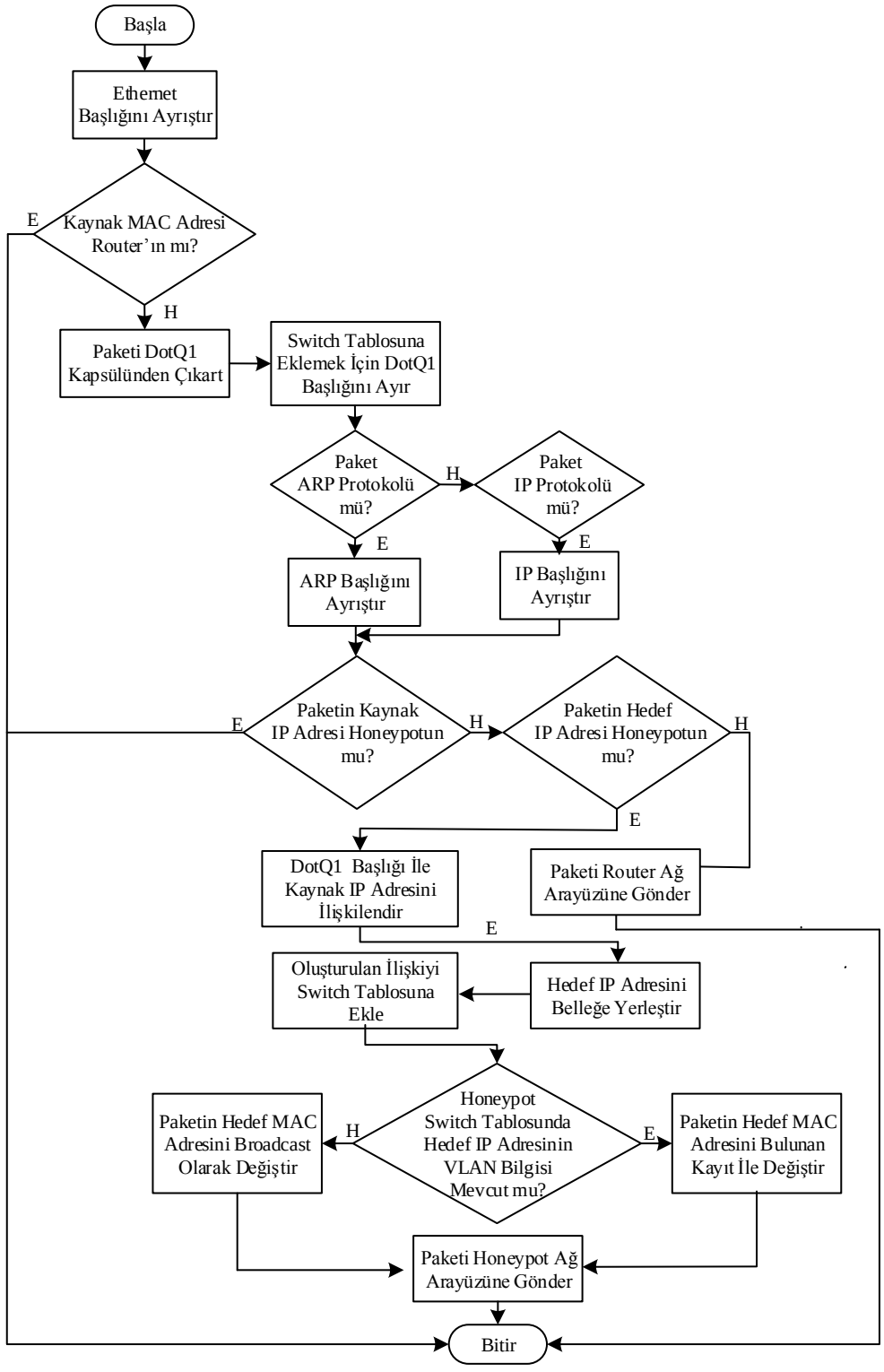
Şekil 7.7'de 2. katman ve 3. katman anahtar seçili konumdayken gerçekleşen algoritma diyagramı gösterilmektedir. Başlat butonuna basıldıktan sonra bu akış gerçekleştirilir ve asenkron olarak seçilen arayüzler dinlenerek yönlendirme işlemleri yürütülür. Şekil 7.7, geliştirilen yazılımsal anahtarın algoritma şemasını göstermektedir. VLAN ağlarına sahip kurumsal kampüs ağlarında katman-2 ve katman-3 anahtar cihazlarındaki yapılandırmaya göre, seçilen ağ arayüzlerini dinleme kabiliyetinde olan yazılımsal anahtar, merkezi olarak konumlandırılan bir IPS üzerinde çalıştırılmakta ve Şekil 7.7'de gösterildiği gibi olası senaryolara göre ağ akışını sağlayabilmektedir. Şekil 7.8'de ise yönlendiriciye bağlı ağ arayüzü için ağ akışı şematik olarak gösterilmektedir. Şekil 7.9, anahtara bağlı ağ arayüzü için akış diyagramını göstermektedir.




**Şekil 7.7** Önerilen yazılımsal anahtar algoritma şeması




**Şekil 7.8** Yönlendiriciye bağlı ağ arayüzü akış diyagramı




**Şekil 7.9** Anahtar bağlı ağ arayüzü akış diyagramı

Şekil 7.9'da gösterilen akış sadece 3. Katman anahtar tipi seçildiğinde gerçekleşmektedir. Bu şekilde, 2. katman ve 3. katman anahtarlar üzerindeki VLAN'ların tüm alt ağlarında honeypotlar, tek bir honeypot sunucu ile temsil edilerek bu çalışmanın amacına ulaşması sağlanmıştır.

## 7.3 Sonuç ve Değerlendirme

Kurumsal ağlar ve honeypot tasarımlarının incelenmesi sonucu, VLAN konfigürasyonunun kullanıldığı ağ sistemlerinde her VLAN için en azından farklı bir ağ arayüzüne sahip bir makine kullanılması gerektiği tespit edilmiştir. Her VLAN için gerçek bir makine kullanılması, özellikle kampüs ağlarını da içeren geniş kurumsal ağlarda honeypotların tüm ağda uygulanması, kurulum ve bakım maliyetlerini arttırmaktadır. Bu nedenle, büyük ölçekli kurumsal ağlarda kurulum, bakım ve yönetim maliyetlerini azaltmak için bir ağ arayüzü ile honeypotların tüm ağda faaliyet göstermesini sağlayan bir merkezi yazılımsal anahtar uygulaması geliştirilmiştir. VLAN içeren kurumsal ağlarda katman 2 ve katman 3'ü dinleyebilen özgün bir yazılımsal anahtar tasarımı yapılarak VLAN ağlarının da dinlenebilmesi sağlanmıştır.

Gerçekleştirilen yazılımsal anahtar uygulaması sayesinde IPS ve honeypot sunucular kolaylıkla etkileşim kurabilmekte, konfigürasyon, bakım, ağ izleme, ağ analizi vb. faaliyetler kolay kullanımlı bir arayüz ile rahatlıkla yapılabilmektedir. Gerçekleştirilen yazılımsal anahtar uygulaması, saldırı çekme bileşeni olarak düşük etkileşimli *honeyd* yazılımını kullanan bir honeypot uygulaması ile özgün olarak geliştirilen ve SNORT kural tabanlarından yararlanan saldırı tespit ve engelleme sisteminin (IDPS) ağ güvenliğini sağlamak üzere kullanıldığı hibrit bir senaryoda uygulanmıştır. İlgili uygulama önceki bölümlerde saldırı tespiti işleminde kullanılmıştır.

Geliştirilen yazılımsal anahtarın başarıyla uygulanması sonucu, imza tabanlı saldırı tespit yöntemlerinin dezavantajlarından olan yeni saldırı örüntülerinin (sıfır gün-zero day) tespit edilememesi, honeypot sunuculardan toplanan log kayıtlarının analiziyle minimum seviyeye düşürülmüştür. Honeypot sunuculardan toplanan log kayıtları çoğunlukla saldırı içeriğine sahip olduğu için log kayıtlarından imzalar üretilerek yeni saldırı örüntülerinin de imza tabanına eklenmesi sağlanabilmektedir. Honeypot sunuculardan toplanan logların anormallik tespiti yöntemleri ile saldırı olmayan log kayıtlarından ayıklanması, başarımı arttırmıştır. Honeypot sunuculardan toplanan saldırı log kayıtları ile anormallik tespiti yöntemlerinin dezavantajlarından olan yüksek *yanlış alarm seviyesi(false positive)* oldukça azaltılmış ve başarımı daha yüksek hibrit bir saldırı tespit yöntemi elde edilmiştir.

# 8. BULUT BİLİŞİM SİSTEMLERİ VE YENİ BİR GÜVENLİK YAKLAŞIMI

## 8.1 Giriş

Bulut bilişim, günümüz elektronik çağında dijital yaşamın, bilgi ve iletişim teknolojilerinin en önemli dönüm noktalarından birini oluşturmaktadır. Sosyal paylaşım sitelerinin gelişimi ile birlikte kullanıcıların ilgisini çekmeye başlayan bulut bilişim; iş dünyasında maliyetleri düşürmesi, e-ticaretin gelişimine katkıda bulunması ve istihdama olan etkisi nedeniyle, büyük şirketlerin ve devlet yönetimlerinin bilgi politikaları arasında öncelikli yerini almıştır. Avrupa Birliği (AB) Komisyonu Başkan Yardımcısı ve Dijital Ajanda üyesi Needie Kroes'in "Bulut Bilişim ve Veri Koruma Reformu" ile ilgili açıklamaları, AB'nin ekonomiye bulut bilişim stratejisi ile yön vermekte olduğunu göstermektedir. Küresel bulut bilişim pazarının 2010 yılındaki değeri 21,5 milyar dolar iken, 2015 yılı için gerçekleşmesi beklenen değer 73 milyar dolar olarak açıklanmıştır. Microsoft firması, 2014 yılında bulut bilişim sayesinde 11,3 milyon yeni istihdam oluşacağını açıklamıştır. Bu açıklama, bulut bilişimin ulaştığı boyutu tanımlamak açısından dikkat çekicidir [94].

Bulut bilişim sistemlerinin gelişimi artmakta ve bu gelişmeler bir takım olumsuzlukları da beraberinde getirmektedir. Özellikle yasal düzenlemelerin yetersizliğinden kaynaklanan belirsizlikler, günümüzde bulut bilişimin kullanılması noktasında çekimserlik oluşturmaktadır. Bilginin en önemli değer olarak ifade edildiği bilişim çağında; mobil iletişim ve bilgi transfer işlemlerinin yoğun olarak yapıldığı bulut sistemlerinin bilgi güvenliğini sağlama noktasındaki yetersizliği, kullanıcıların endişe duymalarının temel nedenidir. Bilgi güvenliğinin sağlanması, kişisel bilgilerin kaydedilmesi ve bilgilerin nerelerde bulunacağı ile ilgili konular; bilgi yönetiminin teknik zorlukları ile birlikte, hukuksal düzenlemelerle de yakından ilgilidir. AB ve ABD'de bulut bilişimin risklerinin ve bilgi güvenliği kapsamında alınabilecek hukuksal önlemlerin yoğun olarak tartışıldığı görülmektedir [94]. Hukuki boyutun tartışıldığı başlıca konular; verilerin farklı bir ülkeye transferi, kişisel bilgilerin korunması ve güvenlik ihlali durumunda neler yapılabileceği hakkında kullanıcının bilgilendirilmesidir.

Batı dünyasında durum böyleyken, Türkiye'de de genel bulut bilişim hizmetlerinden faydalanma oranının hızla arttığı, fakat kullanıcıları meydana gelebilecek zararlar karşısında korumaya alan kişisel verilerin korunması kanunu vb. hukuksal düzenlemenin olmadığı görülmektedir. Bu durum bulut sistemi üzerine aktarılan tüm verilerin sorumluluğunun, bulut hizmeti alan kullanıcı üzerinde bulunduğu anlamını taşımaktadır. Bu nedenle bulut sisteminin sunmuş olduğu kolaylık ve imkânlardan faydalanan kullanıcıların; bulut sisteminin kullanımı esnasında ve sonrasında oluşabilecek riskler ve bu konudaki hukuksal eksiklikler hakkında bilinçli olmaları önemlidir. Bu noktada ülkemizde konu ile ilgili olarak özellikle Google,

Microsoft ve Yahoo gibi bulut bilişim konusunda hizmet sağlayıcı olan büyük firmaların hizmet sözleşmeleri de incelenerek yeni değerlendirmelerin yapılması gerekmektedir.

Bulut bilişime olan ilgi, düşük ücretlere sağladığı yüksek esneklik ve erişilebilirlik sebebiyle son yıllarda ülkemizde de hızla artmıştır. Ayrıca, güvenlik ve gizlilik, uygulamalarını ve verilerini buluta taşırken kuruluşlar için bazı endişeler doğurduğu gözlenmiştir. Söz konusu endişelerin azaltılması için dünya genelinde pek çok ülkede standart çalışmaları yürütülmektedir. Bulut bilişimin ülkemizde de artan oranda kullanılmaya başlanması, bu alanda standartlaşma ihtiyacını gündeme getirmiştir. Bu sebeple, 2013/4890 sayılı Bakanlar Kurulu Kararı doğrultusunda, 20/06/2013 tarihli ve 28683 sayılı Resmi Gazete'de yayımlanarak yürürlüğe giren Ulusal Siber Güvenlik Stratejisi ve 2013-2014 Eylem Planı'nda güvenlik alanında standartların oluşturulması konusunda eylem sorumlusu olarak Türk Standartları Enstitüsü görevlendirilmiştir. Bulut bilişime yönelik standartların oluşturulması ihtiyacı güvenlikle sınırlı olmayıp, güvenliğin yanısıra, güvenilirlik, performans ve firma bağımlılığının önlenmesi gibi geniş bir alana yayılmaktadır. Bu nedenle TSE tarafından bir standart dökümanı hazırlanmış ve içerisine güvenlikle ilgili hususların yanı sıra güvenilirlik, performans ve firma bağımlılığı gibi konular da göz önünde bulundurularak daha geniş ve bütüncül bir bakış açısıyla hizmete sunulmuştur [95]. Oluşturulan bu standart dökümanın temel hedef kitlesi aşağıdaki kategorilerdeki kişileri içermektedir;

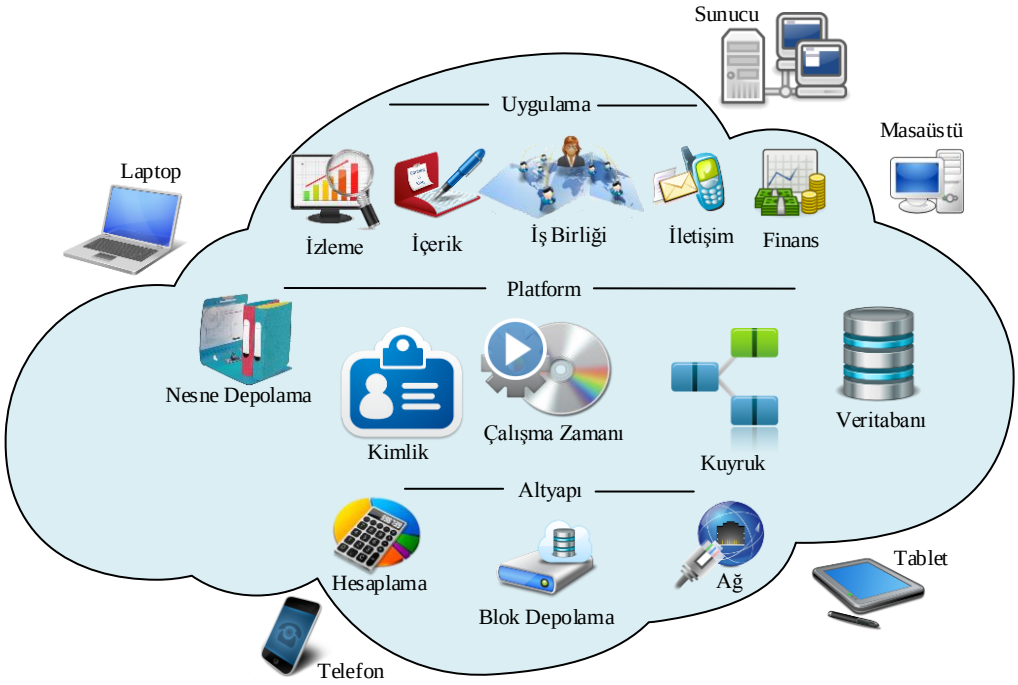
- Bulut bilişim hizmet sağlayıcıları
- Kamu kurum ve kuruluşları
- Sistem yöneticileri, çalıştırıcıları ile bulut bilişimle ilgili kararlar veren BT çalışanları
- Güvenlik uzmanları, yöneticileri, denetçileri ve güvenlikle ilgili sorumluluğu bulunan BT çalışanları
- Bulutta güvenlik ve gizlilik endişesi taşıyan BT yöneticileri
- Sistem ve ağ yöneticileri
- Bulut hizmetlerinin kullanıcıları [95].

## 8.2 Bulut Bilişim Tanımı ve Kapsamı

Bulut bilişim, üzerinde uzlaşı sağlanmış açık bir tanım bulunmamakla birlikte; uygulamaların İnternet ortamında bulunan bir uzak sunucu üzerinden çalıştırılması ya da kullanıcıya ait verilerin uzak sunucu üzerinde her an erişilebilir şekilde bulundurulmasını sağlayan bir servis yapısı olarak tanımlanabilir. Web ara yüzü bilgiyi her yerde ve herkes için ulaşılabilir hale getirirken; bulut bilişim, bilgi işlem gücünü her yerde ve herkes için kullanılabilir hale getirmiştir [94]. Uygulamaların kullanıcı bilgisayarı üzerinden çalıştırılarak

ve bilgilerin yerel bilgisayar diskinde bulundurularak yapılan uzak erişimler, merkezi bilgisayar ve ona bağlı terminal yapısıyla 2000'li yılların başından itibaren kullanılmaktadır [94]. Bu yönüyle bulut sistemi; geçmişte kullanılmış bir yöntemin günümüzün ekonomik şartlarına uyarlanarak ve geliştirilerek, tüm İnternet kullanıcılarının kullanımına olanak sağlayacak şekilde yeniden sunulmasıdır.

Bulut bilişim sayesinde, istenilen bilgiye her yerden ve her türlü bilgi iletişim cihazı (PC, Mac, iPhone, Android veya BlackBerry) kullanarak ulaşmak mümkün olabilmektedir. Donanım kaynaklı problemlerin bulunmaması, fiziksel sunuculardan daha hızlı çalışan sanal bilgisayar ile yüksek erişilebilirlik imkânının sunulabilmesi, bellek ve disk değişikliği gerektirmeyen esnek yapının kullanılması ve doğa dostu (elektrik ve yer tasarrufu) olması, bulut bilişimin ilk bakışta dikkati çeken avantajları arasında görülmektedir. Şekil 8.1'de bulut bilişim için mantıksal bir şema verilmiştir.




**Şekil 8.1** Bulut bilişim mantıksal şeması

Bütün bu avantajlı yönleri göz önüne alındığında; bilgi iletişim teknolojilerindeki gelişimin yansıması olan bulut bilişimden uzak durmak ya da alternatif yöntemlerde ısrarcı olmak akılcı bir çözüm olarak görülmemektedir. Fakat bulut bilişimin beraberinde getirdiği riskler de yok sayılamayacak kadar önemlidir.

Bulut bilişim NIST tarafından, minimum yönetim çabası ve hizmet sağlayıcısı desteği ile yayımlanabilecek ortak havuzlara ve konfigüre edilebilir kaynaklara (örneğin ağlar, sunucular, veri depoları, uygulamalar ve hizmetler) anında erişim sağlayan model olarak tanımlanmaktadır. Bulut bilişim, kaynakların esnek kullanımından dolayı ekonomik fayda,

özelleştirilebilirlik ve daha başka verimlilikler sağlar. Ancak bulut bilişimin, hala yeterince gelişmemiş ve standartlaştırılmamış dağıtık sistemlerin gün yüzüne çıkan bir kolu olduğu unutulmamalıdır [95].

Bulut bilişim, bir veri merkezindeki çok sayıda bilgisayarı birbirine bağlayıp bunlar üzerinde sanallaştırılmış bir platform oluşturarak, yazılım ve donanım servislerinin kullanımına yeni bir yaklaşım getiren bir dağıtım modelidir [96]. Kaynakların birden çok kullanıcı arasında paylaşımı temeline dayanan bulut bilişimde, kullanıcıların altyapı yatırım maliyetleri sıfıra iner ve hizmet sağlayıcının sunduğu ortak altyapı kiralanarak kullanılır. Bu durum kişisel veya kurumsal açıdan bulut bilişimin en çekici özelliklerinden biridir. Ancak bu ortak kullanım belirli güvenlik risklerini de beraberinde getirmektedir. Bulut bilişim sistemleri, dijital verilerin dağıtık olarak işlenmesine yönelik yeni bir kavram ortaya koyar. Veriler birden çok sunucu üzerinde, kullanıcıdan soyut bir şekilde, dağıtılarak işlenir. İşlem anında verilerin hangi konum ve cihazlar üzerinde yer aldığı konusunda hizmet kullanıcıları bilgi sahibi değildir. Bulut bilişim, bu dağıtık ve sanallaştırılmış yapısından dolayı kullanıcıların güvenlik kaygılarını arttırmaktadır.

Bulut bilişim konusunda hizmet veren servis sağlayıcı kurumlar, servis erişilebilirliği ve maliyet uygunluğunu sağlayabilmek için veri merkezlerini dünyanın farklı coğrafyalarında inşa edebilmektedir. Bir veri merkezinde depolanan veriler, güvenlik risklerini azaltmak için başka konumlardaki depolama ortamlarında da yedeklenir.

Bulut bilişim, İnternet tabanlı bir model olmasından dolayı güvenlik tehditleri ve gizlilik ihlallerine açıktır. Bulut bilişim sistemleri sadece kurumlar veya kişiler tarafından ve iyi niyetle kullanılmamakta, saldırganlar tarafından da kullanılmaktadır. Saldırganlar, bulut bilişim sistemlerini, işlenen suça dair delil ve kanıtlarını saklamak, yeni saldırılar planlamak ve yapmak, şifreleme anahtarlarını kırmak gibi çeşitli amaçlar için kullanmaktadırlar. Bulut bilişim servisleri içerisinde en çok kullanılanlardan biri depolama servisidir. Zaman içinde kuruluşlar, bulut bilişimin depolama hizmetini kredi kartı bilgileri ve diğer hassas verilerini de depolamak amacıyla kullanmaya yönelmişlerdir. Saldırganlar, bulut ortamındaki bu verileri hedef aldıkları zaman, karmaşık bir adli soruşturma süreci başlamakta ve dijital soruşturmacıların olayla ilgili delil elde edebilmeleri için zorlu bir yol izlemeleri gerekmektedir [96].

## 8.3 Bulut Bilişimin Karakteristik Özellikleri

Bulut bilişim, bir kurum/kuruluşun bünyesinde ya da kamusal hizmet veren bir ağ yapısında, birçok bilgisayar ve ağ cihazının birbirine bağlanmasına izin verecek şekilde oluşturulan, bir

veri merkezi üzerine kurulmuş olan sanallaştırılmış bir yapıda uygulama vb. hizmetlerin sunulduğu bir destek modelidir. Bulut bilişim, kişi ya da kurumlara bilgisayarların yeteneklerinin genişletilmesiyle ve bir veri merkezinde yer alan yazılım ve servislere, web servisi ya da tarayıcı benzeri özel yazılımlarla İnternet üzerinden erişebilmelerini sağlamaktadır.

Günümüzde bulut bilişim, gün geçtikçe teknolojik yaşamın bir vazgeçilmezi olma yolunda ilerleyen bir olgu haline gelmiştir. Özellikle İnternet kullanımının artması ve bilhassa da İnternet'e açık dijital cihazların yaygınlaşması ile birlikte bulut bilişime olan rağbet artmaktadır. Bulut bilişim ile veri ve uygulamalara, cihazdan bağımsız olarak erişim sağlayabilme özelliğinden dolayı uygulamalar yaygınlaştırılıp ölçeklenebilmektedir.

Google şirketinin yaptığı araştırmaya göre bulut bilişim ile işletme giderlerinin %10 oranında azalacağı, işletme içerisinde %35 oranında bilgi teknolojileri harcamalarının azalacağı buna karşılık; %10 oranında bir gelir ve %50 oranında bir verimlilik artışı sağlanacağı belirlenmiştir [97]. Bulut bilişim sistemlerinin bazı temel özellikleri aşağıda verilmiştir [94-97].

### 8.3.1 Sanallaştırma

Sanallaştırma; fiziksel bir kaynağı, ihtiyaçlar doğrultusunda, istenilen sayıda mantıksal parçaya bölerek, toplam sunucu verimliliğini optimize etmeye yönelik bir teknolojidir. Bulut bilişim mimarisinde, bir veri merkezinin bütün fiziksel altyapısı, sanallaştırma teknolojisi ile çok sayıda sanal sistem haline getirilip, her bir sistem farklı müşterilere hizmet olarak sunulabilmektedir.

### 8.3.2 İhtiyaca Bağlı Ölçekleme

Bulut bilişim platformları, kullanıcıların isteğine bağlı olarak, mevcut altyapıya yeni servis ve kaynakların eklenebilmesine imkân sunar. Yeni kaynak sisteme son derece hızlı bir şekilde eklenip, kullanıma hazır hale getirilebilir. Bu durum maliyet ve zaman açısından son derece önemli ve kazançlıdır.

### 8.3.3 Farklı Coğrafyalarda Kurulu Veri Merkezleri

Bulut bilişim sistemleri dünya çapında hizmet sunmak durumunda olan yapılardır. Çünkü globalleşen dünyada İnternet hizmeti bulunan her yerden erişimin sürekli bir biçimde sunulması önemlidir. Bu sebeple farklı coğrafyalarda veri merkezlerine sahip olmak büyük öneme sahiptir. Bu yapının oluşturulması; ulusal ya da uluslararası yasalar, jeopolitik fikirler, yük dengeleme, ağ gecikmelerini minimuma indirme gibi durumlar açısından gereklidir.

İnternet altyapısı kullanıldığından erişim, İnternet'i olan herhangi bir cihazla ve tarayıcılar ile kolayca sağlanabilmektedir. Ayrıca hiçbir masaüstü uygulamasına veya ekstra tarayıcı plug-in desteğine gerek duyulmamaktadır.

### 8.3.4 Sürekli Gelişim ve Kesintisiz Güncelleme

Bulut hizmeti veren kuruluşlar, sürekli olarak altyapılarını güncellemek ve geliştirmek durumundadırlar. Hem güncel gelişmelere açık olmanın bir gereği olarak hem de yazılımsal olarak güncelleştirme gerektiren durumların vaktinde sağlanması açısından bu durum önemlidir. Bu sebeple sürekli olarak son sürümlerin kullanılması adına sürümlerin her hafta yenilendiği bilinmektedir. Bilgi güvenliği sistemleri açısından temel kavramlardan birinin erişilebilirlik olduğu bilinmektedir. Bulut bilişim sistemleri, %99.9 gibi yüksek bir oranda çalışma garantisi anlaşması yaparak hizmet vermektedir.

Bu özellikler dışında hizmetlerin kullanıldığı kadarıyla ücret ödemeyi sağlayan abone modeli özelliği vardır. Ayrıca bulut bilişim sistemleri gerçek zamanlı uluslararası takım çalışması modelinden dolayı birden fazla dile çeviri hizmeti de sunmaktadır.

## 8.4 Bulut Bilişim Servisleri

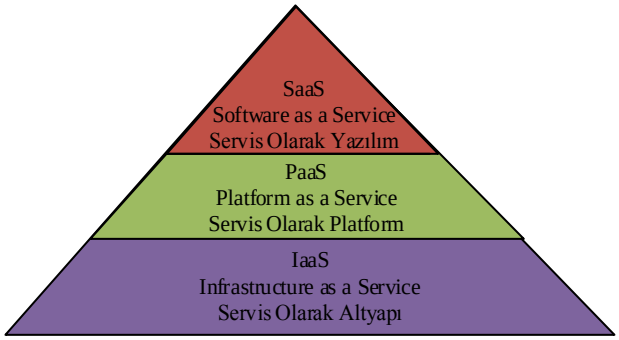
Bulut bilişim kavramı, günümüzün artan mobilite ihtiyaçları doğrultusunda, sunduğu kullanılabilirlik ve maliyet avantajları sebebiyle öngördüğü taleple ciddi bir pazar oluşturmuştur. Büyük teknoloji devleri bu pazardan payını alabilmek için ciddi yatırımlar yapmaktadır. Bulut bilişim, genel olarak kabul gören 3 farklı servis dağıtım modeli sunmaktadır. Bulut bilişim servisleri, Şekil 8.2'de gösterilmektedir.

### 8.4.1 Servis Olarak Altyapı

Servis olarak altyapı (Infrastructure as a Service - IaaS) ifadesi, bulut bilişim altyapısında, yığının en alt tabakasındaki fiziksel servisleri ifade eder. Bu tabaka; sanal makineler, yük dengeleme servisleri, ağa bağlı depolama servisleri gibi temel donanım servislerini içerir. Altyapının bir bulut servisi olarak sunulması modelinde müşteri ihtiyacı olan işlemci, depolama, ağ kaynağı ve diğer temel bilişim kaynaklarını kendisi yapılandırabilmekte ve bunların üzerine ihtiyacı olan işletim sistemi ve uygulamaları kurabilmektedir. Bu modelde bulut kullanıcısı, altyapı üzerinde yönetimi ve tam bir kontrolü olmamasına rağmen, işletim sistemi seviyesinde sisteme tam bir hâkimiyeti bulunmakta ve bazı ağ bileşenlerini (Firewall gibi) yönetebilmektedir [96, 97]. Amazon S3, SQL Azure, Amazon EC2, Zimory, Elastichosts servis olarak altyapı hizmet örnekleridir.

### 8.4.2 Servis Olarak Platform

Servis olarak platform (Platform as a Service - PaaS) bulut servisi, uygulama geliştirmek için kullanılan altyapıyı oluşturmaktadır. Bulut hizmeti alan kullanıcılar, geliştirdikleri uygulamaları, servis sağlayıcı tarafından sunulan platform üzerinde, özelleştirilmiş bir ortamda çalıştırırlar. Bu ortam çoğu zaman kısıtlanmış, düşük imtiyazlı bir yapıdadır. Bu platform uygulamanın geliştirileceği, çalıştırılacağı ortamla birlikte, tamamlayıcı servisleri ve gerekli teknolojik altyapıyı da kapsamaktadır. Kullanıcının kendi kurduğu uygulama dışında, platform altyapısını oluşturan bileşenler üzerinde herhangi bir kontrolü ve yönetim imkânı yoktur. Sistem güvenliğinin sağlanması açısından buradaki yetkilendirme dereceleri oldukça önemlidir [96, 97]. Force.com, Google App Engine, Windows Azure servis olarak platform hizmeti örnekleridir.




**Şekil 8.2** Bulut bilişim servisleri

### 8.4.3 Servis olarak Yazılım

Servis olarak yazılım (Software as a Services - SaaS), bulut bilişim kullanıcılarının bazı uygulama yazılımlarına erişmek için kendi lokal sistemlerine kurulum yapmadan İnternet'e bağlı olan herhangi bir cihaz ile bulut bilişim sistemi üzerinde kurulu bulunan yazılımlara erişmelerini yöneten bir servistir. Bu servis ile uzaktaki bulut bilişim yapısındaki yazılımlar üzerinde çalışmalar yapılabilmektedir. Uygulamalara, web tarayıcıları gibi ara yüzler (Web tabanlı e-posta gibi) aracılığı ile çeşitli kullanıcı cihazlarından erişilebilmektedir. Bu serviste kullanıcılar, altyapıdaki ağ, sunucu, işletim sistemi ve depolama aygıtları gibi bileşenleri yönetmez veya denetlemez. Ancak uygulamanın türüne göre kullanıcıya özgü bazı uygulama ayarları yapılabilir [97]. Google Drive ve Salesforce CRM servis olarak yazılım hizmeti örnekleridir.

## 8.5 Bulut Bilişim Hizmet Türleri

Bulut bilişim hizmet modelleri kullanım biçimlerine göre 4 ana grupta toplanmaktadır. Bunlar, özel bulut, genel bulut, hibrit bulut ve topluluk bulutudur [94].

### 8.5.1 Özel Bulut

Özel bulut (Private Cloud), belirli bir kurum ya da kuruluşa sunulan bulut hizmetidir. Bulut hizmet sağlayıcı, kurumun kendisi olabileceği gibi, bir başka bulut hizmet sağlayıcı da olabilir. Kurum dışından tüm erişim yolları kapatılarak sadece kurum içi hizmet verilir [94]. Bu bulut hizmet türü daha çok büyük şirketler ve veri güvenliği önemli her boyuttan şirkete hitap eder. Bu yapıda şirket dışarısından erişimlere kapalı olan bulut, şirket içerisinde ortak olarak kullanılır. Ekonomik anlamda genel bulut kadar tasarruf sağlamaz. Ancak bilişim yatırım giderlerinde avantajlar sağlar [97].

### 8.5.2 Genel Bulut

Genel bulut (Public Cloud) yapısında, hizmeti kullananların bilişim altyapıları dışarı taşınmış olur. Yani tüm bilgi işlem faaliyetleri, bulut hizmet sağlayıcı şirketlerin kurdukları altyapı üzerinde kiralanan kaynaklar üzerinde yürütülür. Bu çözüm özellikle kişisel kullanım için uygun bir çözümdür. Gmail genel bulut bilgi işlemi için en iyi örneklerden biridir. E-posta hizmetlerinden faydalanmak için hiçbir yatırım yapmanıza gerek bırakmadan, çok iyi yönetilen bir e-posta hizmetinden yararlanmanızı sağlamaktadır. Genel Bulut küçük boyutlu hatta bazen orta ölçekli işletmeler için de, düşük maliyetli çözümler sunmaktadır. Genellikle kullandığın kadar öde sistemiyle çalışır. Aylık ya da kullandıkça ödemeye dayalı sayaçlı sistemleri dahi vardır. Salesforce ve Amazon en bilinen örnekleridir [94, 97].

### 8.5.3 Hibrit Bulut

Hibrit bulut hizmeti (Hybrid Cloud), karma bulut olarak da bilinir ve özel bulut ile genel bulutun birlikte kullanıldığı bir yapıyı ifade eder. Gizlilik ya da güvenilirlik derecesinin çok önemli olmadığı bazı uygulamalar için genel bulutun, gizlilik ve güvenilirliğin daha önemli olduğu alanlarda özel bulutun kullanıldığı sistemlerdir. Örneğin veri depolama için özel bulut, kelime işlemci için ise genel bulutun kullanılmasının tercih edildiği durumlar hibrit bulut hizmetine bir örnek olarak verilebilir. Kullanıcı ihtiyacına göre hangi yapıyı seçeceğine karar verir. Genel kural kişisel kullanımlar için genel bulutun, kurumsal kullanımlar için özel bulutun kullanılmasıdır [97]. Ancak şirketin büyüklüğüne göre, tabii faaliyet alanını da göz ardı etmeden, hibrit çözümler de iyi bir alternatif olabilmektedir.

### 8.5.4 Topluluk bulutu

Belirli bir topluluk ya da gruba sunulan bulut hizmetidir (Community Cloud). Topluluğu oluşturan unsurlar, ortak çalışma alanında bulunan kurumlar da olabilir. Bulut bilişim hizmet

sağlayıcıları, bulut hizmeti sağlarken yazılım, platform ve altyapı hizmet modellerinden birini ya da aynı anda birkaçını kullanmaktadırlar. Fakat bazen bulut hizmet sağlayıcılarının uyguladığı roller karışıklık yaratabilmektedir [94]. Bulut bilişim günümüzde giderek artan bir kullanım oranına sahiptir. Bu artan talebe paralel olarak bilişim dünyasının en popüler alanlarından birisi olan bu alanda birçok firma hizmet yarışına girmiş durumdadır. Microsoft, Google, Apple vb. gibi bilişim sektörünün dev firmaları, bulut bilişim pastasından payına düşeni arttırmak için çalışmalarını sürdürmektedir. Kısım 8.6'da güncel bulut bilişim sistemleri açıklanmıştır [97].

## 8.6 Güncel Bulut Bilişim Sistemleri

### 8.6.1 Google Apps

Google Apps, bulut bilişim hizmeti olarak bu hizmetten yararlananların kullanabileceği, dosyalar oluşturabileceği ve kaydedip saklayabileceği kelime işlemci, tablolama ve sunum uygulamalarıyla birlikte E-mail ve takvim hizmetlerini içeren bir paket sunmaktadır. Kişisel kullanım ile beraber, dosyalar üzerindeki ortak çalışma teknolojisiyle firmalar için de inovatif bir çözüm platformu oluşturmaktadır.

### 8.6.2 iCloud

Sadece Mac ve iOS yani Apple kullanıcıları için olsa bile, dosya saklama ve cihazlar arasında senkronizasyonu her Apple ürünü gibi en basit haline indiren bir web uygulaması ve bir veri saklama hizmetidir. iCloud sayesinde, iPhone'da çekilen fotoğraflar hiç bir şey yapılmadan Mac sistemlerde görüntülenebilmekte, Mac'de güncellenen adres defterine anında iPhone'dan ulaşılabilmektedir.

### 8.6.3 Office 365

Office 365, Microsoft firmasının ofis yazılımları içinde markalaşmış olan MS Office'i bulut üzerinde çalıştıran bir altyapıdır. Bu altyapı hizmeti, Microsoft Business Productivity Online Suite (BPOS) ve Microsoft Office SharePoint Online, Exchange Online ve Lync Online ile beraber yeni bir oluşum sunmaya hazırlanmaktadır.

### 8.6.4 Dropbox

Dropbox oldukça popüler bir bulut hizmeti olup, dosyaların bulut ortamında saklanmasına olanak veren, zaman ve sabit disk kullanımında büyük ölçüde tasarruf etmemize imkân

sağlayan bir yapıdır. Ayrıca online yedekleme işlemleri için de kullanılabileceği gibi, farklı platformlarda çalışan kişilerin rahatlıkla kullanabileceği bir yapıdır.

### 8.6.5 Evernote

Evernote, not alma, bir toplantıyı sesli olarak kaydetme, İnternet'teki yazı, resim vb. dökümanları bulut üzerinde depolayan birden çok platforma destek veren bir bulut uygulamasıdır. Evernote, depolanan her türlü dosyaya, PC, Mac, iPhone, Blackberry vb. gibi her türlü cihazdan ulaşıma olanak sağlamaktadır.

### 8.6.6 Zoho

Zoho, kendi içerisinde kelime işlemci, hesap tablosu, CRM, proje yönetimi, veri tabanı, raporlama araçları gibi 29 ayrı web uygulamasını barındıran bir bulut bilişim hizmetidir. Bu hizmetin içindeki uygulamalar ile üretilen verileri kişisel olarak ya da firma hesaplarıyla saklayıp kullanmak mümkündür. Gelişmiş web uygulamalarıyla (SaaS) bulut teknolojisinin her türlü nimetinden yararlanan Zoho, kendisine bu alanda Google ve Microsoft ile bir yer edinmeye çalışmaktadır.

### 8.6.7 NetSuite

Bulut üzerinde çalışan içinde ERP, muhasebe, sipariş yönetimi, stok, CRM, profesyonel hizmet otomasyonu (PSA) ve e-ticaret de dahil olmak üzere modüller bulunan entegre bir uygulamadır. Ocak 2012 itibariyle 10.000′den fazla kullanıcısı ve 3 milyar dolar piyasa değeri bulunan NetSuite'in bir önceki adı Oracle Small Business Suite'dir.

### 8.6.8 Salesforce

Web üzerinde çalışan CRM uygulamasıyla bulut bilişimin büyükbabası olarak kabul gören bir bulut hizmetidir. Bu uygulamaya eklediği Chatter ile sosyal ağ entegrasyonunu, AppExchange ile de diğer bulut uygulamaları ile entegrasyonu sağlamaktadır.

### 8.6.9 IBM Websphere Cast Iron Cloud Integration

Bulutta ve/veya kullanıcı sunucu bilgisayarlarında bulunan dosyalar, veri tabanları, mesajlaşma sistemleri ve web hizmetlerinin yanı sıra, SAP, Siebel, SalesForce, Microsoft Dynamics, NetSuite, ve Lotus Domino gibi uygulamalar da dahil olmak üzere, uç noktalara farklı bağlantılar sağlayıp bunları entegre eden bir platform uygulamasıdır. Bu uygulama ile

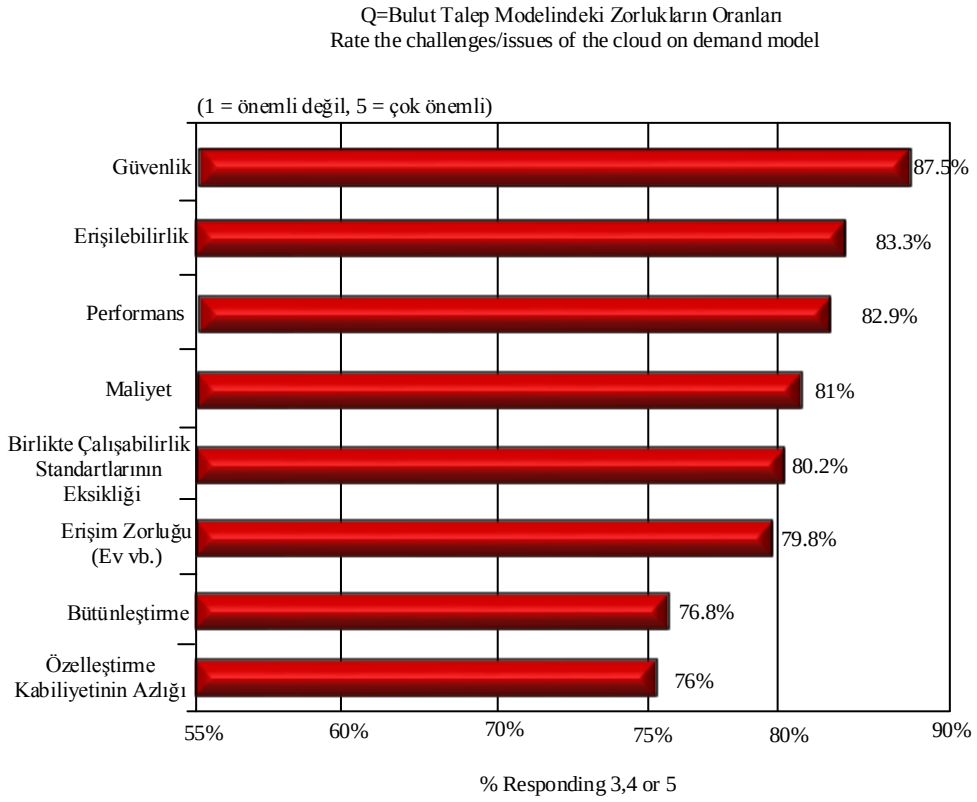
Oracle CRM'e girilmiş bir kayıt, bir başka yerdeki MySQL'e dahi anında ulaştırılabilmekte ve MySQL kullanan uygulamanın da kendi içinde bu verilere sahip olması sağlanabilmektedir.

### 8.6.10 Workday HCM İnsan Sermayesi Yönetimi

Finansal yönetim, masraf takibi, bordro gibi uygulamaları içinde barındıran ve bunları bulut üzerinden hizmete (SaaS) sunan yeni nesil bir yazılımdır. Özellikle farklı coğrafi lokasyonlarda bulunan çalışanların işlemlerini yapmak için hazırlanmış bir yapısı bulunmakla beraber bulut teknolojisi ile firmaların kendi bünyelerinde barındıracakları donanım ve yazılım risklerinin de önüne geçmiş olmaktadır [97].

## 8.7 Bulut Bilişimde Güvenlik

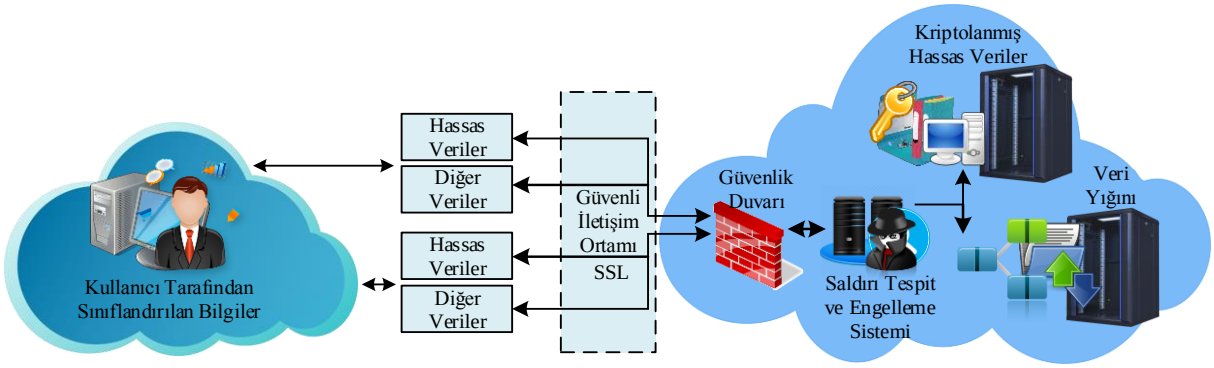
Bulut bilişim dünyasında güvenlik en önemli unsurlardan biridir. Bilginin güvenliğinin sağlanması konusu, bilgi güvenliği sistemlerinin temelini oluşturmaktadır. IDC'nin yaptığı bir araştırmaya göre bulut bilişim kullanımına başlangıçta yaşanılan en fazla çekince konusu güvenlik konusundadır. Güvenlik'ten sonra performans ve erişilebilirlik gibi unsurlar gelmektedir. Şekil 8.3'te bu unsurlar verilmiştir [97, 168].




**Şekil 8.3** Bulut bilişime geçerken çekinilen konular

Bulut bilişim sistemlerinde güvenliğin sağlanması için atılması gereken en öncelikli adımlardan biri "bilginin sınıflandırılması" 'dır. Sınıflandırmadaki amaç bilginin değerinin ortaya konulmasıdır. Bilginin herhangi bir elektronik ortamda güvenliğinin ve gizliliğinin korunabilmesi için değerinin belirlenmiş olması gerekmektedir. Bu değerin belirlenmesine örnek bir durum olarak bir bilginin "hassas bilgi" şeklinde tanımlanması ve böylece o bilginin güvenlik ya da gizliliğinin nasıl ve hangi sorumluluklarla ilgili olarak korunacağı konusunda ayrıcalık oluşturması verilebilir. Hassas bilgi, gizli tıbbi gerçekler, etnik kökenler, ırklar, siyasi görüşler, dini inançlar ya da cinsellikle ilgili bilgilerdir ve bunlar belirli bir kişisel bilgi kategorisi olarak sınıflandırılabilir [94].

Bulut hizmetleri sunulurken, bilginin sınıflandırma işleminin, kullanıcı tarafında bulut sistemine transferi öncesinde yapılması ve bulut hizmet sağlayıcılarının bu sınıflandırmayı dikkate alarak hizmet sunması; bulut bilişim üzerinde alınabilecek en etkili önlemlerden biridir. Bu yapının sağlıklı çalışabilmesi için Şekil 8.4'te gösterilen basit model önerilebilir. Bu modelde, bulut sistemine ulaşıncaya kadar olan süreçte bilginin izlediği yol, bankacılık sisteminde uygulanan güvenlik tedbirleri ile benzer niteliktedir. Bu modelde, kullanıcıların bankacılık hizmetlerinde olduğu gibi, hizmet aldıkları sunucuların nerede olduğu ve nasıl bir güvenlik yapısına sahip olduklarını bilmedikleri; fakat belirli bir standardı sağlamayan bir hizmet sağlayıcının faaliyet gösteremeyeceği öngörülmektedir. Güvenli iletişimin sağlanması için kullanılacak güvenlik protokolleri (SSL-Secure Socket Layer), giriş kısıtlamaları, kimlik doğrulama ve tek kullanımlık şifre uygulaması; bulut hizmeti veren hizmet sağlayıcılar tarafından da uygulanmalı ve bilgi güvenliği standardını karşılayacak düzeyde olmalıdır. Kullanıcı tarafından sınıflandırılmış olan bilgiler, güvenli iletişim ortamından geçerek bulut hizmetinin sunulduğu ortama ulaşır. Bu bilgiler, güvenlik duvarı ve saldırı tespit/engelleme sistemleriyle zararlı kodların tespiti ve sonrasında kimlik denetimi gibi bir dizi güvenlik kontrolünden geçirilir. Kullanıcı tarafından sınıflandırılan veriler, iki farklı sunucu ya da depolama alanına yönlendirilir. Hassas bilgilerin bulunduğu alan şifrelenmek suretiyle üst seviyede bilgi güvenliği önlemi alınır. Böylece, bu alana ilişkin olarak yapılan zararlı kod saldırıları ve yetkisiz erişimler sonucunda veri bütünlüğü ve gizliliğin sağlanması konusunda meydana gelebilecek zafiyetlerin önüne geçilir. Paylaşılan veri depolama alanlarında bilgi güvenliği risklerinin bulunması her zaman olasıdır [94].




**Şekil 8.4** Güvenli bir bulut bilişim modeli

Böyle bir modelin en basit düzeyde dahi uygulamaya geçirilebilmesi noktasında, yasal düzenlemeler ve bilgi güvenliği politikalarına ihtiyaç duyulmaktadır. Ülkemizde bankacılık sektörüyle ilgili 5411 Sayılı Bankacılık Kanunu çerçevesinde hazırlanmış düzenlemelerde (Bankalarda Bilgi Sistemleri Yönetiminde Esas Alınacak İlkelere İlişkin Tebliğ) olduğu gibi; belirli kriterlerin belirlenmesi, belirli aralıklarla bilgi güvenliği denetimlerinin yapılması ya da sertifikalandırma sisteminin oluşturulması, önemli bir zorlayıcı unsur olacaktır. Bankacılık sektöründe ve hatta İnternet toplu kullanım sağlayıcılarının faaliyet alanlarındaki sınırları belirleyen yönetmelikler dahi yapılmış ve uygulanıyorken; bulut bilişim hizmeti verecek olan şirketlerin faaliyetlerini icra ederken kullanacakları bilgi sistemlerinin yönetimi amacıyla uymak zorunda oldukları usul ve esasları gösteren bir düzenlemenin bulunması da aynı derecede önem taşımaktadır [94]. Bulut bilişim sistemlerinde güvenlik konusu oldukça geniş bir alan içermektedir. Bu alan, hizmet sunan firmaların güvenilirliği, erişim ve kimlik denetimi, erişilebilirlik, fiziksel güvenlik, legal ve operasyonel güvenlik ile veri ve altyapı güvenliği gibi çok geniş bir yapıdadır [97].

### 8.7.1 Hizmet Sunan Firmaların Güvenilirliği

Bulut bilişim sistemlerindeki güvenlik kaygılarının giderilmesine yönelik olarak ilk güvenlik unsuru bulut hizmet sağlayıcılarının güvenilirliğidir. Hizmet sağlayıcı firmanın sektördeki tecrübesi oldukça önemlidir. İlgili firmanın sektörde faaliyet gösterdiği süre ve bu sürede ortaya çıkan güvenlik sorunlarının çözülme istatistikleri, karşılaşılan vakaların düzeltilmesi için verilen hizmet kalitesi oldukça önemlidir.

Ayrıca hizmet sağlayıcı firmanın işletim sırasında kullandığı metodolojiler (ITIL, COBIT gibi.), takip ettiği regülasyonlar (ISO 9001, 27001-2, BS 25999 gibi.) ve konularında uzman (CISSP, CEH, CISA, CCIE, GIAC vs. gibi sertifikasyonlara sahip) yetişmiş personel kaynağı da hizmet kalitesi açısından önemli göstergelerdir.

### 8.7.2 Erişim ve Kimlik Denetimi

Hizmet alınan firmanın servislerine erişim sırasında mutlaka güvenli bir bağlantı yöntemi kullanılmalıdır. Bu erişimin, sadece hizmet alan kişi ya da kurum tarafından yapıldığından emin olunması için, aşağıdaki erişim kontrolü (authentication) teknolojilerinden (two factor – iki katmanlı kontrol- vs. gibi) bir veya birkaçı birden desteklenmelidir [97];

**Kimlik Doğrulama (Authentication)**

- Bildiğiniz bir şey (Something you know) – şifre (password) gibi.
- Sahip olduğunuz birşey (Something you have) – token anahtarlar gibi.
- Sizin kimliğiniz (Something you are) – biyometrik kontroller, retina kontrolü gibi.
- Bulunduğunuz yer (Someplace you are) – GPS sinyalleri ya da IP bazlı kontroller gibi.

**Yetkilendirme (Authorization)**

- Firma tarafından kullanılan hesaplarda, hangi hesabın hangi kaynaklara erişebileceği dikkatle tanımlanmalıdır ( RBAC (Role Base Access Control) ).

**Hesap Verilebilirlik (Accountability)**

- Bulut üzerindeki tüm hareketler kayıt altında olmalı ve olası soruşturmalara yardım edebilmelidir. Gereğinde yasal delil olarak da kullanılabilecek şekilde toplanabilmelidir.

### 8.7.3 Erişilebilirlik

Bulut bilişim hizmetlerinde alınan hizmetler, SLA (Service Level Agreement) ile güvence altına alınmaktadır. Örneğin hizmet alınan sunucuların aktif olarak hizmet etme süresinin %99,99 olması vs. gibi. Özellikle yüksek erişilebilirlik gereksinimi içeren projelerde, "Felaket Önleme" (Disaster Recovery) servislerinin olup olmadığı kontrol edilmelidir. Elbette hizmet sağlayıcı firmaların finansal durumları da alınacak hizmetin kesintisizliği ve kalitesi için belirleyici olacaktır [97].

### 8.7.4 Fiziksel Güvenlik

Fiziksel güvenlik, güvenlik unsurları içerisinde önemli yeri olan bir güvenlik türüdür. İhmal edilmesi durumunda olabilecek kötü olaylar (hırsızlık, yangın, su baskını vb. gibi) oldukça ciddi boyutta maddi kayba veya geri dönülemez sonuçlara neden olabilir. Bulut hizmet sağlayıcısının

veri merkezinde (Data Center) sahip oldukları fiziksel güvenlik sertifikaları (TIER 3-4 vs. gibi) ve güvenlik önlemleri (biyometrik kontroller, yanmaz duvar ve kapılar, 7x24 güvenlik, fiziksel bariyerler, sel-yangın önleme sistemleri, UPS sistemleri, soğutma sistemleri, alarm mekanizmaları) incelenmeli, gerektiği durumlarda denetlenme opsiyonlarının olup olmadığı araştırılmalıdır.

### 8.7.5 Legal ve Operasyonel Güvenlik

Bulut bilişim alanında alınan hizmetlerin lisans durumları ve yasal yükümlülükler, sözleşmeler ile güvence altına alınmalıdır. Ülkemizde SPK, BDDK gibi kuruluşların yönergeleri ve yasal yükümlülükler, yine güncel yasalardan 5651 vs. gibi yönetmeliklerin sorumlulukları takip edilmelidir. Hizmet veren firmanın operasyonel hizmetlerde çalışan personelleri ile imzaladıkları NDA vs. gibi sözleşmeler sorgulanmalı ve personel özgeçmiş soruşturmaları yapılıp yapılmadığı eğer mümkünse bilgi/taahhüt olarak talep edilmelidir [97].

### 8.7.6 Veri ve Altyapı Güvenliği

Bulut bilişim hizmet sağlayıcısının veri ve altyapı güvenliğini üst düzeyde sağlaması oldukça önemlidir. Bu açıdan yedekleme ve veri silme/yok etme biçimleri sorgulanmalıdır. Kullanımı biten verilerin imha edilme metodolojileri sorgulanmalı, mümkünse ayrı bir servis olarak alınabilmelidir. Hizmet sağlayıcı firmanın bilgilerinin ya da kurumsal önem içeren sunucuların verilerinin mutlaka bir şifreleme algoritması ile saklanması, gerektiğinde servis olarak alınabilmesi sağlanmalıdır [97].

## 8.8 Güncel Bulut Çalışmaları

Kumar N. ve diğ. yaptıkları çalışmada bulut bilişim ortamında saldırı tespiti için denetimsiz yönteme dayalı, etkin bir dışsal saptama kavramı olan DenOD (Density Based Outlier Detection)'u önermişlerdir. Kullanılan denetimsiz dışsal saptama teknikleri; ağ saldırı tespit, arıza tespiti ve sahtekarlık tespiti gibi çeşitli uygulama alanlarında rol oynayan bir yöntemdir. DenOD bulut düğümleri, STS ve son kullanıcı gibi bileşenlere sahip IDCC (Intrusion Detection in Cloud Computing) framework'ü üzerinde uygulanmıştır [98].

Ficco M. ve diğ. yaptıkları çalışmada bulut ortamlarında dağıtık saldırı tespitini desteklemek için temel araştırma konularını belirlemişler, ayrıca bulut sağlayıcılara servis olarak güvenlik çözümleri sunan, bulut bilişimde saldırı tespiti sağlayan dağıtık bir mimari sunmuşlardır. Bu mimari, bulut ortamında karmaşık olay ilişki analizi gerçekleştirmek için güvenlik bileşenleri kullanarak bilgi toplayan çok katmanlı ve hiyerarşik bir yapıdır [99].

Yassin W. ve diğ. çalışmalarında bulut tabanlı saldırı tespit servisi (CBIDS) olarak adlandırılan bulut bilişime dayalı yeni bir framework tanıtmışlardır. Bu model, klasik saldırı tespit sistemlerinin eksikliğinin üstesinden gelmektedir ve ağın farklı noktalarındaki zararlı faaliyetlerin tanımlanmasını sağlamaktadır. CBIDS, özel ve kamu bulutlarındaki saldırıların çeşitliliği tespit etmek için uygulanabilmektedir [100].

Han Li. ve diğ. yaptıkları çalışmada saldırı davranışlarını tespit etmek ve karmaşık saldırıları tespit kabiliyetini geliştirmek için bulut teorisine dayanan bir saldırı tespit algoritması önermiş ve bulut platformunun elastik özellikleri kullanılarak DDoS saldırılarına karşı bir savunma stratejisi tanıtmışlardır. Bu stratejiyle üretilen model, *saldırı tespit aracı* alt sistemi ve *veri toplama* alt sistemi tarafından oluşmuştur [101].

Mehmood Y. ve diğ. yaptıkları çalışmada bulutdaki farklı saldırılara genel bir bakış sunmuşlardır. Yaptıkları çalışmada, bazı mevcut bulut tabanlı saldırı tespit sistemlerini; türlerine, konumlandırılmalarına, tespit tekniğine, veri kaynağına, tespit zamanlarına ve tespit edilebilme özelliklerine göre analiz etmişlerdir [102].

Nikolai J. ve diğ. çalışmalarında hipervizör performans ölçümlerini kullanarak saldırı tespit güvenliği için bulut bilişimin özündeki sanallaştırma teknolojisinden yararlanarak bir mimari ve yaklaşım önermişlerdir. Bu yaklaşımla engelleme aygıtı okuma/yazma istekleri, paketlerin iletimi/alımı ve CPU kullanımı gibi hipervizörlerden toplanan sanal makine performans ölçütlerinin kullanımı sayesinde sanal makineler içinde çalışan işletim sisteminin ayrıntılı bilgisi olmadan profili çizilebilen şüpheli faaliyetler gösterilmiş ve doğrulanmıştır [103].

Li Z. ve diğ. çalışmalarında buluttaki herhangi tek bir makineye aşırı yükleme yapmadan mevcut kaynaklardan tam olarak faydalanmak için uygulanabilir bir mimari ile dağıtılmış bir sistem olan YSA tabanlı bir STS önermişlerdir. Ayrıca geliştirilen çalışmada, YSA makine öğrenmesi yeteneği ile önerilen STS yapısının oldukça doğru sonuçlar ile saldırıların yeni türlerini algılayabileceği savunulmuştur [104].

Vanathi R. ve diğ. yapmış oldukları çalışmada bulut sistemlerdeki en ünlü ağ tabanlı STS araçlarından olan Snort, Tcpdump ve Network Flight Recorder incelenmiş ve karşılaştırılmıştır. İncelemeler sonucunda, Snort'un uygulanmasının diğer mevcut ticari ve açık kaynak araçlardan daha kolay olduğu sonucuna varılmıştır. Snort'un mali, teknik ve/veya idari açılardan uygun olduğu belirtilmiştir [105].

Mohamed H. ve diğ. yapmış oldukları çalışmada dağıtık STS ve SES yapılarının fonksiyonlarını içeren işbirlikçi bir model önermişlerdir. Bu modelde karşılaşılan saldırıların sorunlarını ele almak için hibrit bir algılama tekniği kullanmışlardır. Önerdikleri güvenlik

sisteminin işleyişini geliştirmek amacıyla yeni bir saldırı üretmek için İmza Apriori Algoritması entegrasyonunu da sisteme dahil etmişlerdir [106].

Donadio Pasquale'nin çalışmasındaki amacı dağıtık ağlarda ağ güvenliği ve güvenilirliği ilkelerini uygulayan, güvenlik risklerini en aza indiren bir işlem ve yeni bir yapı ortaya koymaktır. Tanımlanan mimarinin temelinde bulut tabanlı STS prototipi uygulamaları ortaya konulan tezi doğrulamıştır. Bu prototip, OpenNebula ve DRAGON olmak üzere iki farklı teknolojinin entegrasyonu sayesinde oluşturulmuştur [107].

Roschke S. ve diğ. yaptıkları çalışmada bulut sistemlerde STS yapılarının dağıtık kullanımı için çeşitli gereksinimleri özetlemiş ve dağıtık bir bulut altyapısında kolaylıkla kullanılabilecek genişletilmiş bir STS mimarisi önermişlerdir [108].

Jun-Ho L. ve diğ. yaptıkları çalışmada hem güvenlik hizmetinin sistem kaynağını ve gücünü kullanan hem de verimliliği üst düzeyde elde eden bulut bilişim sistemi sağlayan bir yöntem önermişlerdir. Çalışmada, bulut bilişim sistemlerinde etkin STS uygulanması için kullanıcı davranışlarına dayalı log yönetim metodu ve çok seviyeli bir STS önerilmiştir [109].

Gupta S. ve diğ. yapmış oldukları çalışmada bulut ortamında anormallik tespiti için hypervisor sanal makineler tarafından üretilen sistem çağrı dizilerinin analizi üzerine kurulu bir yaklaşım sunmuşlardır. Önerilen uygulama ile sıklıkla çalıştırıldığı bilinen programları değiştirmeye çalışan zararlı sanal makine (VM) kullanıcıları önlenmektedir [110].

Aljurayban N.S. ve diğ. çalışmalarında izlenen bulut trafiği arasındaki normal trafiğin varlığını belirlemek amacıyla bulut bilişimin farklı katmanlarında uygulanabilen katmanlı saldırı tespit frameworkü (LIDF) denilen etkin bir framework önermişlerdir. Önerilen framework yapısında özellikle doğruluk, hız ve ölçeklenebilirlik sağlayan, Yapay Sinir Ağı ve Veri Madenciliği kullanılmıştır [111].

Sijin He. ve diğ. çalışmalarında IaaS bulut altyapılarında saldırı tespiti için yeni bir güvenlik izleme frameworkü önermişlerdir. Önerilen framework, sanal makinenin hem içerisinde hem de dışarısında izlenen veriler üzerinde istatistiksel sapma tespiti tekniklerini kullanmaktadır [112].

Modi C.N. ve diğ. çalışmalarında Bayes, Associative ve karar ağaçları gibi sınıflandırıcıları ve Snort'u kullanan hibrid ağ saldırı tespit sistemi olan yeni bir framework önermişlerdir. Bu framework ile performans ve hizmet kalitesi sağlanırken, ağ trafiğini izleyerek buluttaki ağ saldırılarının tespit edilmesi hedeflenmiştir. Bulut yapısında uygulanabilirliği sağlamak için H-NIDS'in performansını ve algılama etkinliğini değerlendirmişlerdir [113].

Kumar N. ve diğ. yaptıkları çalışmada bulut bilişim ve saldırı tespit sistemlerinde DDoS saldırıları ile başa çıkmak için kullanılabilecek bir çalışma üzerine odaklanmışlardır [114].

Wang X. ve diğ. çalışmalarında geleneksel saldırı tespit yöntemlerinin eksikliğini telafi edebilecek ve saldırıları büyük ölçüde önleyebilecek bulut tabanlı yeni bir saldırı tespit sistemi geliştirmişlerdir. Bununla beraber herhangi bir web sitesinde, yerel algılama motoru, saldırı örüntülerini bulmak için bulut merkezi tarafından toplanan veriler analiz edilmiş ve oluşturulan tüm uyarılar ağ güvenliğinin global bir görünümünü vermek ve daha karmaşık saldırıları bulmak için global saldırı tespit motoru tarafından işlenmiştir [115].

Wang D. ve diğ. çalışmalarında bulut modelini saldırı tespitine uygulamış ve saldırı veri kümelerini verimli şekilde gözlemlemişlerdir [116].

Lo C. ve diğ. yaptıkları çalışmada bulut ortamlarındaki güvenlik konularından biri olan DoS veya DDoS saldırısının etkisinin nasıl azaltılacağı konusuna dikkat çekmişler ve bu tür saldırılara karşı koymak için, yardımcı saldırı tespit sistemli bir framework önermişlerdir. Bu beceriyi kazandırabilmek için bulut hesaplama sistemindeki STS'ler, kendi uyarılarıyla diğerlerininki arasında değişim yaparlar. Bu sistemde, her bir STS, diğer STS'lerden gönderilen uyarıyı kabul edip etmediğini belirlemek için kullanılan bir birleşik ajana sahiptir. Bu yolla bulut STS'ler, benzer tipte meydana gelen saldırıları önleyebilmektedirler [117].

Andreolini M. ve diğ. çalışmalarında, bir saldırganın durumsal modunda çalışan en modern ağ saldırı tespit sistemleri dahil, hiçbir bölümü mevcut defans mekanizmaları tarafından kabul edilemeyen, kötü niyetli bir yükü bölen, taşınabilirlik tabanlı sızma denilen gizli ağ saldırılarının yeni bir sınıfını tanıtmışlardır. Ayrıca çalışmalarında sızma tespiti için taşınabilirlik tabanlı sızmayı önleyebilen özgün bir işbirliği sağlayan framework önermişlerdir. Çalışmada Mobil IPv4, IPv6 Mobil ve WiFi protokolleri uygulanan bir prototip ile önerilen çözümün canlılık ve performansı gösterilmiştir [118].

Modi C. ve diğ. yapmış oldukları çalışmada bulut kaynaklarının ve hizmetlerinin kullanılabilirliği, gizliliğini ve bütünlüğünü etkileyen farklı müdahalelere yönelik bir araştırma yapmışlardır. Bulut sistemlerde STS ve SES yapılarını içeren önerileri incelemiş ve yeni nesil ağlarda istenen güvenliği sağlamak için bulut ortamında STS/SES yapılarının konumlandırılmaları için önerilerde bulunmuşlardır [119].

Donadio P. ve diğ. yapmış oldukları çalışmada sanal saldırı tespit sistemleri üzerine odaklanmışlardır. V-IDS (Virtual Intrusion Detection System) adını verdikleri çalışmalarında [120], bulut bilişim ağlarını korumak için, bulut bilişimin temel karakteristik servislerini ve sanallaştırma teknolojilerini kullanan yeni bir mimari yaklaşım sunmuşlardır. Riyo J. ve diğ. [121] ve Tari Z. ve diğ. [122], bulut bilişim çevreleri için güvenlik ve gizlilik hususlarının sağlanması için, vizyon, trend ve zorluk noktalarını ortaya koymuşlardır.

Varadharajan V. ve diğ. yapmış oldukları çalışmada [123], bir bulut sağlayıcının altyapısal anlamda bir parçası olabilecek ve sağlayıcı tarafından müşterilerine koruma amaçlı önerilebilecek güvenlik servisleri üzerinde durmuşlardır. Yapılan çalışma, bir bulut güvenlik mimarisi tasarımının yanısıra, önerilen yaklaşımın kullanılması ile hangi tür saldırıların nasıl etkisiz hale getirilebileceği konusunda da bilgiler sunmaktadır.

Chen Z. ve diğ. yaptıkları çalışmada [124], ağ güvenliği adli bilişim analizi için, bulut bilişim temelli bir güvenlik modeli önermişlerdir. İlgili çalışmada, bulut bilişim trafiğinin toplanması ve bu toplanan trafik verilerinin analiz edilmesi sonucu kötücül aktivitelerin yakalanması şeklinde bir yaklaşım önerilmiştir.

Xuexiu Chen ve diğ. yaptıkları çalışmada [125], sınıflandırma ve derecelendirme temelli bir bulut güvenlik değerlendirme göstergesi önermişlerdir. Önerilen yapı, rasyonelliğin ve kullanılabilirliğin doğrulanabilmesi için, ileri doğru ve geri doğru değerlendirme yöntemlerini birlikte kullanan kapsamlı bir değerlendirme metodu kullanmaktadır.

Birçok araştırmacı bulut bilişim çevrelerinde veri ve bilgi güvenliğinin sağlanması konusunda çalışmalar yapmışlardır [126]. Choo K. ve diğ. yaptıkları çalışmada, bir bulut güvenliği risk yönetimi stratejisi geliştirmişlerdir. İlgili çalışmada, bir bulut güvenliği risk yönetimi stratejisi dökümanının, paydaşlar tarafından düzenli bir biçimde gözden geçirilmesi gerektiği ve organizasyonun ihtiyaçlarına göre amaç ve politikaların belirlenmesi gerektiği vurgulanmıştır [127].

Qian Wang ve diğ. yaptıkları çalışmada [128], bulut bilişim sistemlerinde, veri depolamada bütünlüğün sağlanması problemi üzerinde durmuşlardır. Gongjun Yan ve diğ. yaptıkları çalışmada [129], araç bulut bilişim sistemlerinin güvenlik zorlukları üzerinde durmuşlardır. Yapılan çalışmanın temel katkısı, güvenlik zorluklarının ve potansiyel gizlilik tehditlerinin tanımlanması ve analiz edilmesi olarak belirtilmiştir.

Zhifeng Xiao ve diğ. yaptıkları çalışmada bulut bilişim sistemleri için, güvenlik ve gizlilik üzerine kapsamlı bir araştırma sunmuşlardır [130]. Yapılan araştırmanın dayandığı temel özellikler; gizlilik, bütünlük, erişilebilirlik, hesap sorulabilirlik ve korunabilirlik şeklindedir.

Madria S. ve diğ. yaptıkları çalışmada [131], bulut servis sağlayıcısı tarafından önerilen güvenliği ölçmek için bir çevrimdışı risk değerlendirme framework yapısı geliştirmişlerdir. Bir başka çalışmada ise yazarlar, bulut bilişimde veri gizliliğinin elde edilme yollarını tartışmışlardır [132]. Karnad K. ve diğ. yaptıkları çalışmada, bulut bilişimin güvenli olup olmadığı sorusunu irdelemişlerdir. Çalışmada, bulut bilişim sisteminin güvenlik testi ile üç katmanlı mimari için güvenlik testinin farklılıkları gibi açılardan bulut güvenliği incelenmiştir [134].

Lin Chen ve diğ. yaptıkları çalışmada, yazılım tanımlamalı ağ, sanal makine trafik yönlendirmesi, ağ politikası tanımlama yönetimi, yazılım tanımlı izolasyon ağları, açıklık tarayıcı ve yazılım güncelleme teknolojileri üzerine inşa edilmiş bir dinamik ağ güvenlik mimarisi önermişlerdir [135].

## 8.9 Özel Bulut Ağlarının Güvenliği İçin Yeni Bir Yaklaşım

Önceki bölümlerde honeypot sistemlerinin kurumsal ağlar üzerinde konumlandırılması ve VLAN içeren ağlarda tek bir merkezden kontrolü için yazılımsal anahtar tasarımı gerçekleştirilmesi gibi konular üzerinde durulmuştu. Bu bölümde, özel bulut ağları üzerinde, tasarlanan honeypot ve saldırı engelleme sisteminin kullanılması incelenmiştir.

### 8.9.1 Özel Bulut Ağlarında Honeypot STS'nin Kullanılması

Önceki çalışmalarda tasarlanan honeypot ve saldırı engelleme sisteminin, gözlenmek istenen ağa seri olarak bağlanması gerekmekteydi. Aynı fiziksel ağ altyapısını kullanan fakat farklı sanal ağlarda yer alan altyapıları izleyebilmek için yazılımsal anahtar tasarımı gerçekleştirilerek, VLAN teknolojisini barındıran 3. katman anahtarlarda kullanılmak üzere bir çalışma yapılmıştı. Yapılan bu çalışmanın sonucunda fiziksel olarak yalıtılmamış olan ağların tüm trafik akışının tek bir merkez üzerinden kontrolü sağlanmıştır. Böylece, aynı fiziksel ağ altyapısı içinde bulunan özel bulut sistemleri, sanallaştırma sistemleri, DMZ bölgesi ve iç ağ trafiğinin bir merkez üzerinden güvenliğinin oluşturulması sağlanmıştır.

Bir kurumsal ağ, bir tek fiziksel ağdan oluşabileceği gibi birden fazla yalıtılmış fiziksel ağdan da oluşabilmektedir. Özellikle özel bulut sistemlerinin kurulduğu altyapılar, farklı fiziksel ağlarda çalışan sistemlerin birleşiminden oluşabilir. Bu durumda, kurum içi ağ trafiğinin kontrolü için iki merkezli bir kontrol mekanizması veya birden fazla sistemin birleştiği bir şebeke kurulmalıdır.

Geliştirilmiş olan honeypot temelli saldırı engelleme sistemi, her merkezde birbirinden bağımsız olarak çalışmaktadır. Birbirine göre fiziksel olarak yalıtılmış ağların her biri için bir honeypot sistemi kurulmak zorundadır. Kurulan honeypot sistemlerinin izlenmesinin iki farklı merkez tarafından yapılması gerekmektedir. Bu durum özellikle birbirinden yalıtılmış olan fiziksel ağlarda bulunan özel bulut sistemlerinin izlenmesinde sorun oluşturacaktır.

Honeypot saldırı engelleme sistemi ile kurumsal ağ içindeki yalıtılmış fiziksel ağların güvenliğinin sağlanmasında oluşan bu gereksinim, geliştirilen Muxer uygulaması sayesinde sağlanmıştır. Muxer uygulaması, çalışmakta olan birden fazla honeypot saldırı tespit ve

engelleme sisteminin bir merkez üzerinden kontrol edilmesini ve saldırı faaliyetlerinin izlenmesini sağlamaktadır.

### 8.9.2 Özel Bulut Güvenliği İçin Önerilen Yaklaşım: Muxer

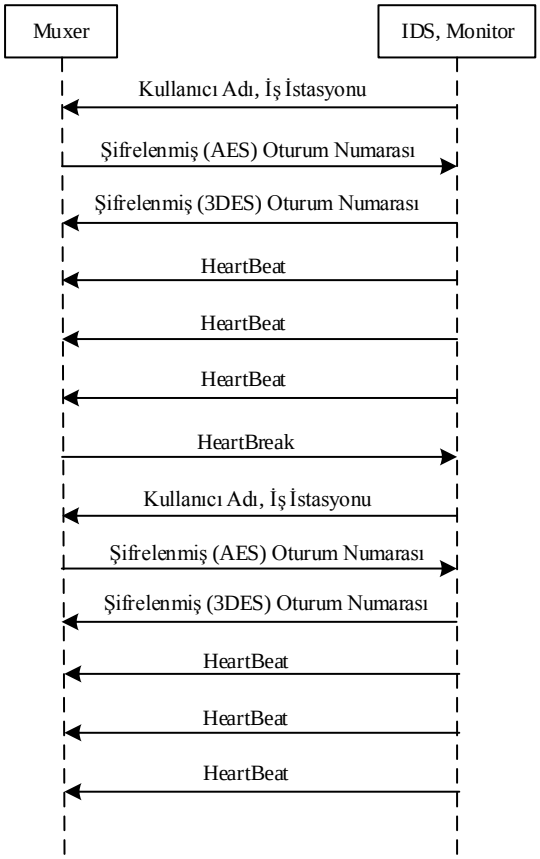
Muxer uygulamasının ana görevi, sistem içinde görev alan IDS uygulamasının kendisine kayıt olduktan sonra bağlantılı olan IDS sistemlerin çıktılarını tek bir Monitör uygulamasına yönlendirebilmektir. IDS uygulaması, tüm ağ ve honeypot faaliyetlerinin izlendiği uygulamadır. Uygulama, girilen kullanıcı adı ve şifre ile Muxer uygulamasına kayıt olur. Daha sonra yakaladığı şüpheli paket bilgilerini kurduğu bağlantı üzerinden gönderir. Bu noktada Muxer uygulaması ile IDS uygulamasının arasında kurulan iletişimin doğruluğunun sağlanması oldukça önem kazanmaktadır.

Diğer taraftan Muxer uygulamasının Monitör uygulaması ile de bağlantı kurması gerekmektedir. Monitör uygulaması girilen kullanıcı adı ve şifreyi kullanarak Muxer uygulamasına kayıt olur. Bu şekilde Muxer uygulaması aynı kullanıcıya ait IDS uygulamalarından gelen verileri kullanıcıya ait Monitör uygulamasına iletir. Bu şekilde tüm özel bulut ağ trafiğinin bir merkez üzerinden izlenebilmesi sağlanmış olur.

**Uygulamalar Arası İletişimin Kurulması:** Uygulamaların iletişime başlayabilmeleri için öncelikle kendilerini Muxer uygulamasına kayıt ettirmeleri gerekmektedir. Bunun için Muxer uygulamasında öncelikle bir şirket oluşturulmakta ve kullanıcı adı ve şifre belirlenmektedir. Bu şekilde Muxer uygulaması, diğer uygulamalar arası iletişimi ilişkilendirebilmektedir. Muxer uygulaması, iki farklı kanal üzerinden diğer uygulamaların kendine bağlanmasını beklemektedir. Birinci kanal üzerinden IDS uygulamaları bağlanırken ikinci kanal üzerinden Monitör uygulamaları bağlanmaktadır. Kurulacak iletişim ve bağlantının güvenliği için bu kanallar TCP protokolü kullanılarak gerçekleştirilmiştir. IDS ve Monitör uygulamalarının Muxer uygulamasına bağlandıktan sonra bağlantılarının doğruluğunu sağlamak için bir protokol geliştirilmiştir.

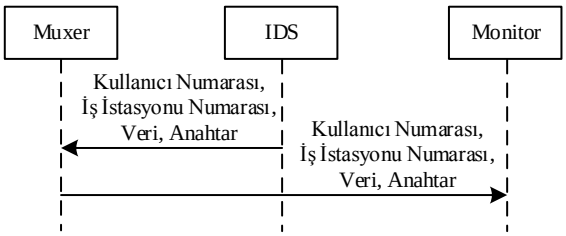
Muxer uygulamasına istemci uygulama olarak kayıt olmak isteyen uygulama öncelikle kullanıcı adı ve iş istasyonu numarasını gönderir. Muxer uygulaması iş istasyon numarası ve kullanıcı adıyla ilişkilendirdiği rastgele bir oturum numarası oluşturur ve kullanıcı adına ait şifre ile bunu simetrik bir şifreleme yöntemi kullanarak şifreler ve istemciye gönderir. İstemci oturum şifresini kendinde bulunan şifre ile deşifre ederek oturum numarasını öğrenir. Farklı bir simetrik şifreleme yöntemi kullanarak oturum numarasını yeniden şifreler ve Muxer uygulamasına gönderir. Muxer uygulaması şifreyi çözerek oturum numarasını doğrular. Bu şekilde Muxer ile istemci arasında kurulan bağlantı doğrulanmış olur. Muxer uygulaması

kurulan bağlantı üzerinden belli aralıklar ile HeartBreak sinyali göndererek yeni oturum numarası ile oturumun tazelenmesini sağlar. Uygulamanın oturumunu devam ettirmesi için istemci uygulamadan belli aralıklar ile HeartBeat sinyali göndermesini bekler. Uygulamadan zaman aşımı süresi boyunca HeartBeat sinyali alınamamış ise geçerli oturum sonlandırılır. Şekil 8.5, uygulamalar arası bağlantı protokolünü göstermektedir.



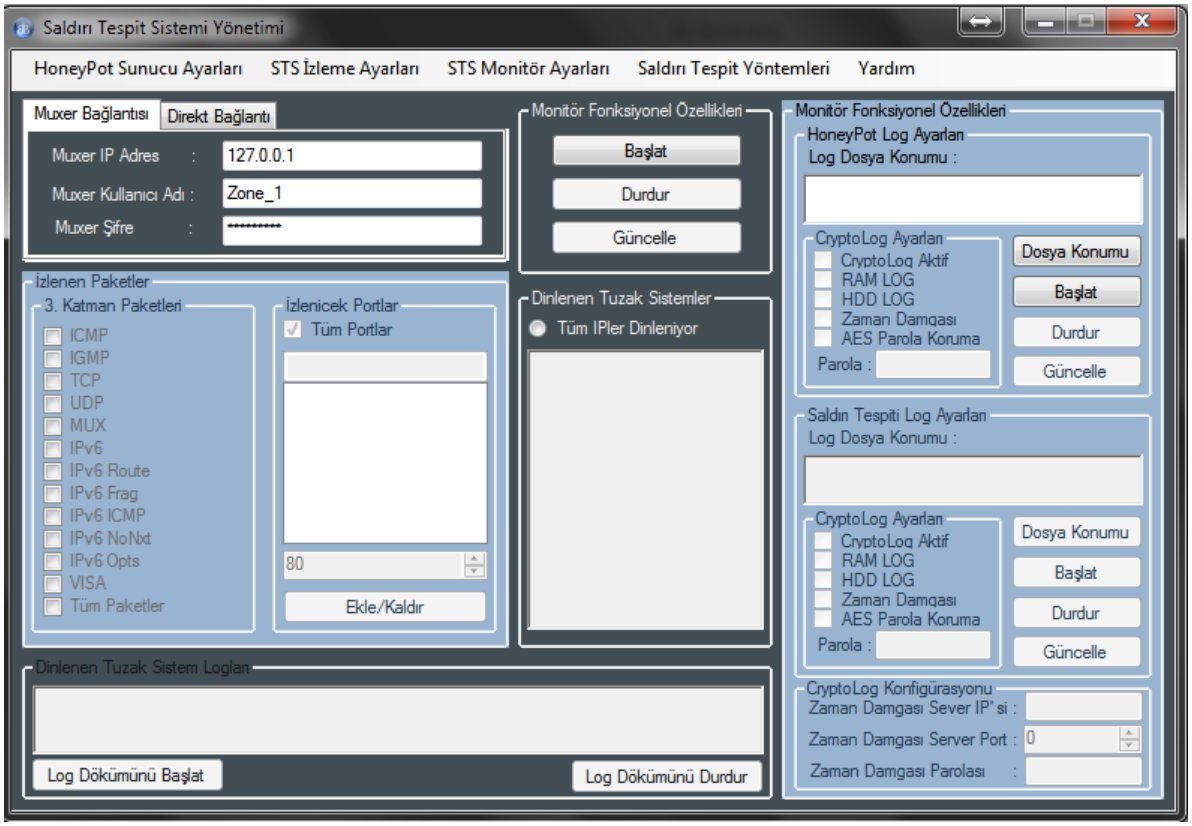

**Şekil 8.5** Muxer uygulaması bağlantı doğrulama protokolü

Muxer uygulaması, istemci ile veri iletimini sağlamak için bir UDP kanalı kullanır. İstemci göndereceği veriyle birlikte kullanıcı numarasını, iş istasyon numarasını ve mesaja özgü oluşturduğu anahtarı ekler. Bu özel anahtar mesajın hash değeri ile oturum numarasının hash değerinin XOR işleminden geçirildikten sonra şifre ile simetrik şifreleme yöntemi kullanarak şifrelenmesi ile oluşur. Muxer uygulaması mesajı aldığında, mesajın geldiği iş istasyonu numarasına ait oluşturduğu oturum numarasının hash değeri ile mesajın içeriğinin hash değerini XOR işleminden geçirerek şifreler ve mesajın doğruluğundan emin olur. Gelen veriyi doğruladıktan sonra aynı şekilde diğer istemcinin oturum numarası ile mesaja özel anahtarı yeniden oluşturur. Bu şekilde diğer istemcinin de mesajı doğrulayabilmesi sağlanmış olur. Şekil 8.6'da, uygulamalar arası veri iletişim protokolü gösterilmektedir.



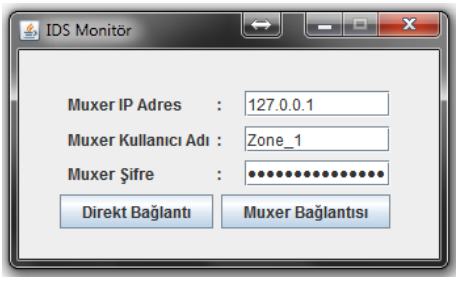

**Şekil 8.6** Uygulamalar arası veri iletişim protokolü

IDS uygulaması, yazılımsal anahtar tasarımı ile birlikte çalışarak saldırı olarak tespit edilmiş olan paketlerin hedeflerine ulaşmasını da engelleyebilmekte ve bu paket bilgilerini monitör uygulamasına göndermektedir. Uygulamanın Muxer uygulamasına entegre edilebilmesi için tasarlanan iletişim protokolünün istemci tarafının uygulamaya eklenmesi gerekmektedir. IDS uygulamasının Muxer uygulamasına kayıt olma işlemi ve iletişimin devamı sırasında kullanacağı bilgiler uygulamanın arayüzünde yeni bir panel aracılığıyla kullanıcıdan alınmaktadır. Burada kullanıcıya iki seçenek sunulur. İstenirse monitör uygulamasına direkt olarak bağlantı kurulabileceği gibi Muxer uygulamasına da kayıt olunabilir. Bu şekilde saldırı paket bilgileri dış ağa çıkmadan da geliştirmeden önceki şekilde de izlenebilir. Şekil 8.7, Muxer uygulamasının entegrasyonu için geliştirilen arayüz tasarımını göstermektedir.



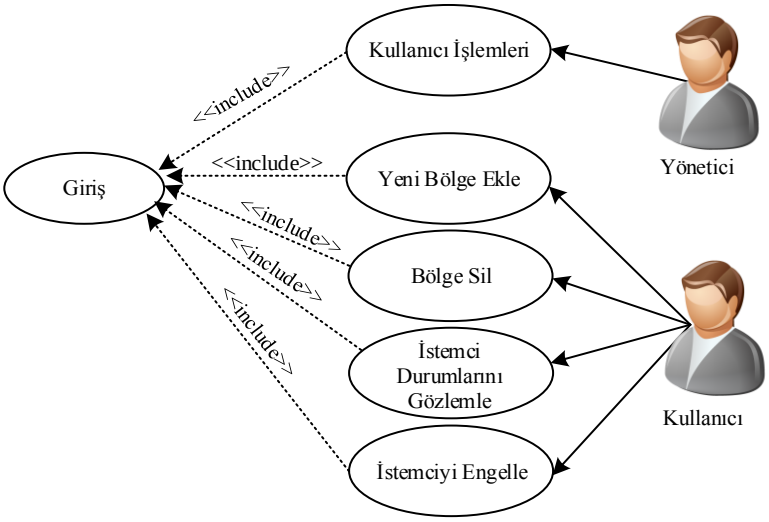

**Şekil 8.7** Muxer-STS bağlantı arayüzü

Monitör uygulaması (monitoring-analyzing), IDS üzerinden gelen saldırı paket bilgilerini analiz ederek üzerinde kayıtlı olan konumlandırma bilgilerini kullanarak ilgili animasyonu gerçekleştirir. Bunun için sürekli IDS uygulaması için açtığı portu dinleyerek IDS uygulamasının göndereceği mesajları dinler. Monitör uygulamasına birden fazla IDS uygulaması mesaj gönderebilir. Bu durumda, Monitör uygulaması mesajın hangi IDS yazılımından geldiğine bakmaksızın gelen paketin içeriğini analiz ederek üzerinde kayıtlı olan konumlandırma bilgilerini kullanır. Monitör uygulamasında da IDS uygulamasında olduğu gibi kurulacak iletişim hakkında kullanıcıya iki seçenek sunulmaktadır. Bunlardan birincisi, IDS ile direkt olarak iletişimin kurulması, ikincisi ise Muxer uygulamasına bağlanılmasıdır. İkinci seçenek söz konusu olduğunda, kullanıcıdan uygulamanın Muxer uygulamasına bağlanması ve sonrasında kurulacak iletişimin devamında kullanacağı bilgileri almak için bir giriş paneli tasarlanmıştır. Şekil 8.8'de giriş panelinin ekran görüntüsü yer almaktadır.



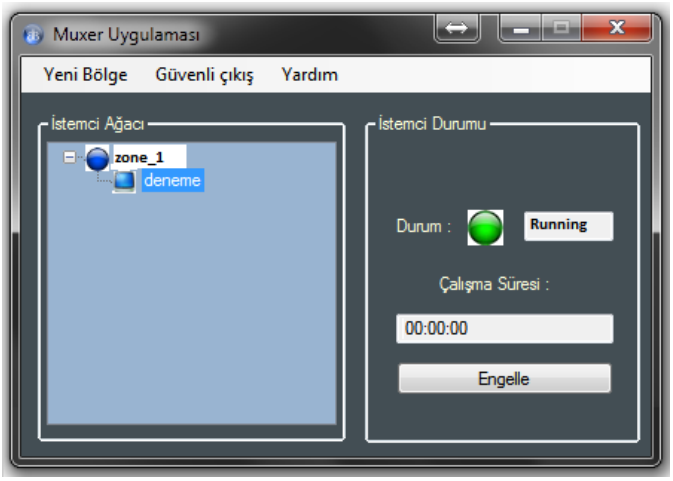

**Şekil 8.8** Muxer-Monitör bağlantı arayüzü

Muxer uygulaması, IDS ve Monitör uygulamaları arasında belirlenen protokolleri kullanarak iletişimi sağlar. Bunun için daha önceden Muxer uygulamasına, kendine bağlanacak olan IDS ve Monitör uygulamalarına ait bilgilerin verilmesi gerekmektedir. Uygulama daha sonra çalıştığında, bu bilgileri kullanarak kendine bağlanan IDS ve Monitör uygulamalarını kayıt işlemi sırasında tasarlanan protokol ile doğrulayarak otomatik olarak algılar. Muxer uygulamasında, kullanıcı tarafında yapılabilecek işlemler Şekil 8.9'daki usecase diyagramında verilmiştir.




**Şekil 8.9** Muxer uygulaması use-case diyagramı

Kullanıcının tanımlanan fonksiyonları gerçekleştirebilmesi ve istemci durumlarını gözlemleyebilmesi için sade ve kullanışlı bir arayüz tasarımı yapılmıştır. Panelin sol tarafından aktif ve inaktif istemci durumları gözlenirken sağ tarafında seçilen istemciye ait bilgiler ve yapılabilecek işlemler yer almaktadır. Üst menü çubuğundan yeni kayıt, güvenli çıkış ve yardım işlemleri yapılabilmektedir. Bu durum Şekil 8.10'da görülmektedir.



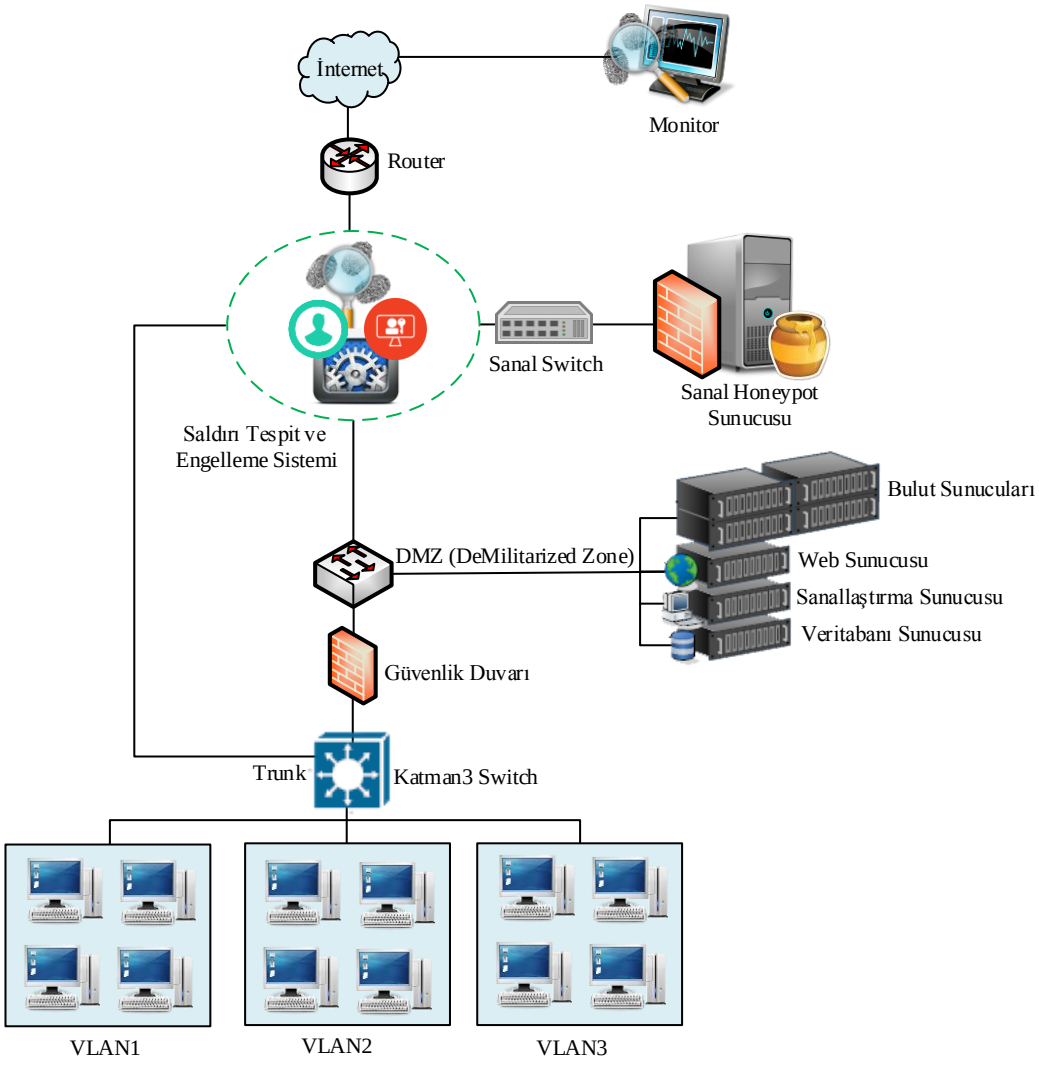

**Şekil 8.10** Muxer uygulaması arayüzü

### 8.9.3 Özel Bulut Güvenliği İçin Farklı Senaryolar

Bu çalışmada geliştirilen Muxer uygulamasının kullanımına dair üç senaryo üzerinde durulmuştur. Bu senaryolar sırası ile, yalıtılmamış fiziksel ağ altyapısından oluşan özel bulut sistemleri, yalıtılmış fiziksel ağlarda bulunan özel bulut sistemleri ve ağın kendi içinde yeniden yalıtılarak oluşturulan kurumsal ağlardır.

**Senaryo 1:**

Bu senaryoda izlenmek istenen ağ aynı fiziksel altyapıya sahiptir. Ağ altyapısı içinde özel bulut ve sanallaştırma sunucuları, DMZ bölgesinde hizmet veren diğer sunucular ve kurum içi ağ altyapısı bulunabilir. Buradaki tüm ağ faaliyetleri bir tane honeypot sunucu ile izlenebilir. Haricen Muxer uygulamasının kullanılmasına gerek yoktur. Bu çalışma senaryosu, Şekil 8.11'de gösterilmektedir.

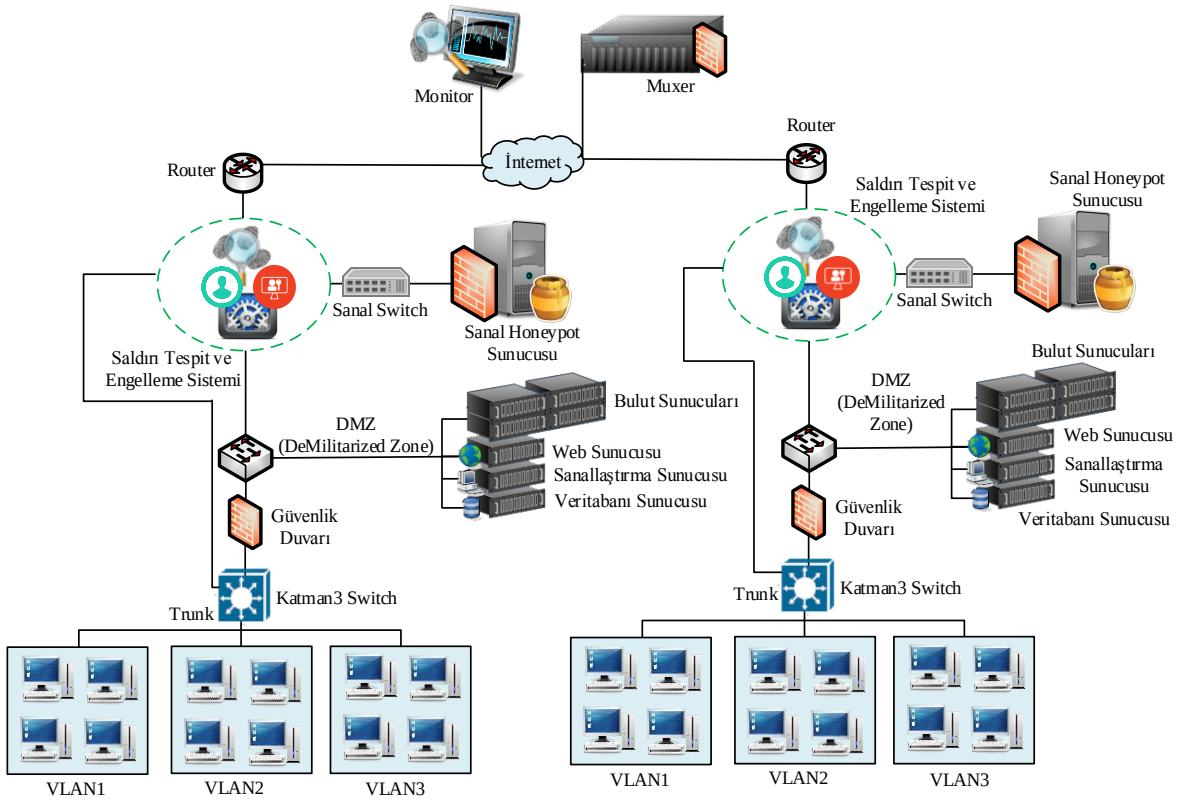



**Şekil 8.11** Senaryo 1 için çalışma diyagramı

**Senaryo 2:**

Bu senaryoda, şirketin birbirinden yalıtılmış iki fiziksel ağı bulunmaktadır. Burada, ağ altyapısı ortak özel bulut ve sanallaştırma sunucuları, DMZ bölgelerinde bulunan diğer sunucular ve iki ağda bulunan kurum içi ağlardan oluşabilir. Bu durumda, yalıtılmış iki fiziksel ağın

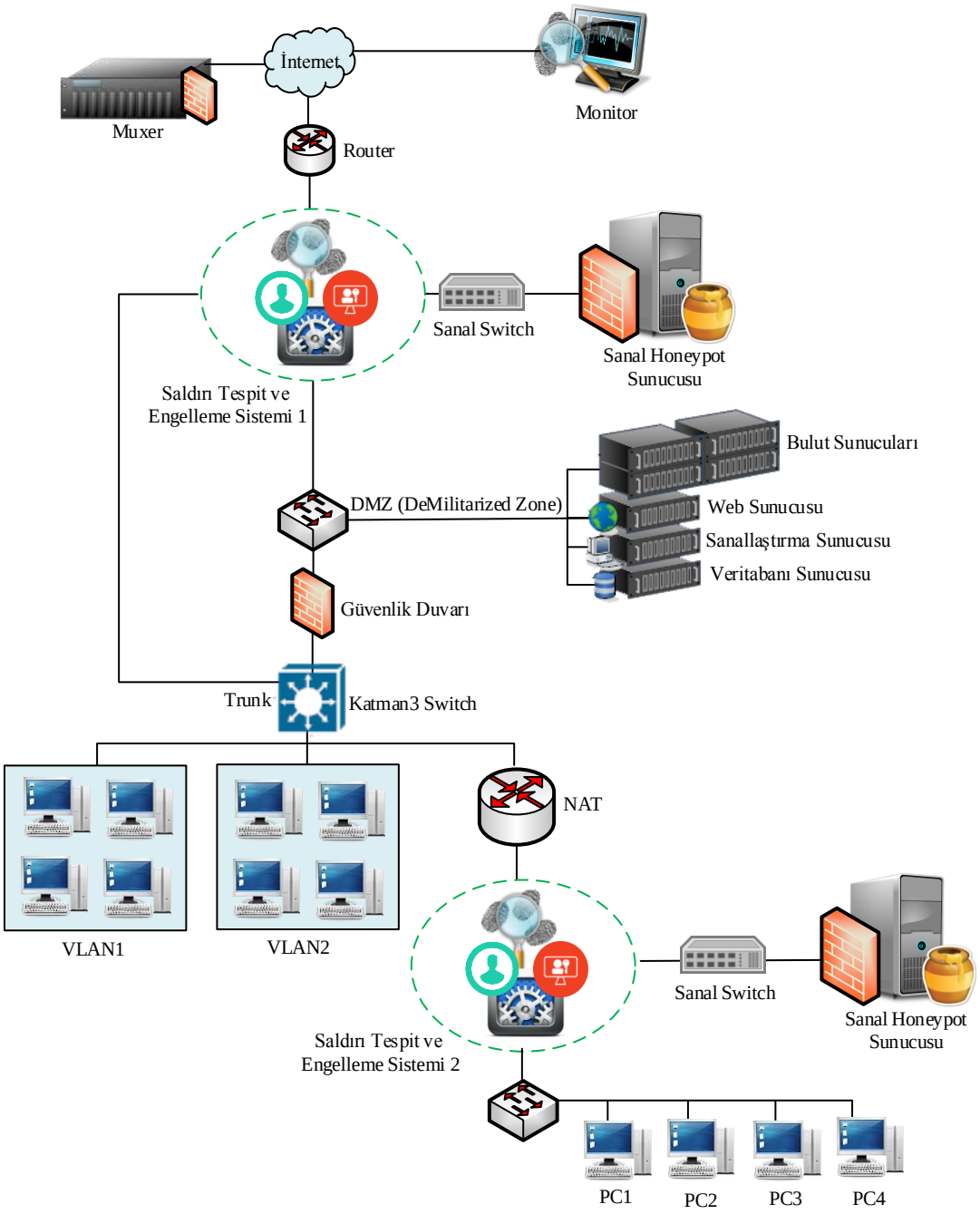
faaliyetlerini izleyebilmek için Muxer uygulamasına ihtiyaç duyulmaktadır. Senaryo 2 için çalışma diyagramı Şekil 8.12'de gösterilmiştir.




**Şekil 8.12** Senaryo 2 için çalışma diyagramı

**Senaryo 3:**

Bu senaryoda fiziksel olan ağ kendi içinde NAT ile tekrar yalıtılarak oluşturulmuştur. Bu durumda ağ içi faaliyeti izlenmek istenen her alt ağ için bir honeypot sunucu kullanılması gerekecektir. Kurulan honeypot sunucuların iletişimi, Muxer uygulaması ile sağlanarak tüm ağ içi trafik bir noktadan izlenebilmektedir. Senaryo 3 için çalışma diyagramı, Şekil 8.13'te verilmiştir.




**Şekil 8.13** Senaryo 3 için çalışma diyagramı

Bu çalışmada geliştirilen Muxer uygulaması ile önceki bölümlerde gerçekleştirilen çalışmalara ek olarak honeypot temelli saldırı engelleme sisteminin *dağıtık özel bulut* ağlarında da kullanılabilmesi hedefine varılmıştır. Ortaya konulan ağ altyapısı senaryoları ile uygulamanın kullanım kapsamı belirlenmiştir. Geliştirilen Muxer uygulaması ile ağ analizinde kullanılabilecek bir merkezi işlem birimi altyapısı oluşturulmuştur. Çalışmaların devamında Muxer uygulamasının üzerinden akan trafiği analiz ederek yeni saldırı yöntemlerinin tespit edilmesi ve aktarılan verilerin şifrelenerek loglanması sağlanabilecektir.

# 9. SONUÇLAR VE DEĞERLENDİRME

Bu tez çalışmasında, bilgi güvenliği sistemlerinin vazgeçilmez araçlarından biri haline gelen saldırı tespit ve engelleme sistemleri üzerine yeni yaklaşımlar geliştirilmiş ve önerilmiştir. Önerilen yeni yaklaşımlarla STS'lerin en önemli dezavantajlarından biri olan yanlış alarm seviyesinin düşürülmesi amaçlanmıştır. Bu amaç doğrultusunda bu sistemlerin honeypot sistemlerle entegrasyonunun sağlanması ve önerilen bu yeni yaklaşımların özel bulut bilişim ağlarında kullanılması sayesinde yüksek performanslı yeni bir bulut güvenlik yaklaşımı sağlanması amaçlanmıştır.

Tez çalışması kapsamında, ağ güvenliğinde kullanılabilecek gerçek zamanlı saldırı tespit ve engelleme sistemleri için honeypot temelli yaklaşımlar önerilmektedir. Önerilen bu özgün yaklaşımlar için etkin yazılımlar geliştirilmiştir. Geliştirilen sistem, düşük ve yüksek etkileşimli honeypotların üstün özelliklerini alarak tek bir yapıda birleştiren, başarımı daha yüksek, hibrit bir honeypot sistemi olma özelliğine sahiptir. Geliştirilen sistem, gerçek zamanlı olarak bir kampüs ağı üzerinde test edilmiş ve başarılı sonuçlar gözlenmiştir. Bu çalışma ile honeypot sistemler, ilk defa geniş kampüs ağlarını içeren yerel ağ sistemlerinde güvenliği tamamlayıcı bir unsur olarak uygulanmıştır. Bu kapsamlı çalışmada, honeypotların kurumsal ağlarda kullanımı incelenerek kurulum, bakım ve yönetim maliyetini düşürmek için sanallaştırma teknolojileri kullanılmıştır. Honeypot sunucularda oluşan ağ trafikleri, animasyon şeklinde görsel olarak izlenebilmektedir. Böylece, sistem durumu hakkında daha kolay bilgi sahibi olma imkânı sağlanmıştır.

Kurumsal ağlar ve honeypot tasarımları incelendiğinde, VLAN konfigürasyonunun kullanıldığı ağ sistemlerinde her VLAN için en azından farklı bir ağ arayüzüne sahip bir makine kullanılması gerekliliği tespit edilmiştir. Her VLAN için gerçek bir makine kullanılması, özellikle kampüs ağlarını da içeren geniş kurumsal ağlarda honeypotların tüm ağda uygulanması, kurulum ve bakım maliyetlerini arttırmaktadır. Bu nedenle, büyük ölçekli kurumsal ağlarda kurulum, bakım ve yönetim maliyetlerini azaltmak için bir ağ arayüzü ile honeypotların tüm ağda faaliyet göstermesini sağlayan merkezi bir honeypot sunucu uygulaması geliştirilmiştir. VLAN içeren kurumsal ağlarda, katman-2 ve katman-3'ü dinleyebilen özgün bir *yazılımsal anahtar* tasarımı yapılarak VLAN ağlarının da dinlenebilmesi sağlanmıştır.

Gerçekleştirilen STS uygulaması, honeypot sunucuları dinlemenin yanı sıra, ağa bağlanarak saldırı engelleme sistemi olarak da kullanılabilmektedir. Bunun için imza tabanlı saldırı tespit yöntemleri sisteme entegre edilebilmektedir. Geliştirilen STS uygulamasında, açık kaynak kodlu SNORT saldırı tespit sisteminin paket yakalama (Sniffing) modülü baz alınarak kötüye

kullanım tespiti yapabilen uygulama gerçekleştirilmiştir. STS uygulaması yakaladığı paketlerin imza tabanındaki saldırılarla uyuşması durumunda paketin iletimini keserek, ağı saldırılardan koruyabilmektedir.

İmza tabanlı saldırı tespit yöntemlerinin dezavantajlarından olan yeni saldırı örüntülerinin (sıfır gün -zero day) tespit edilememesi, honeypot sunuculardan toplanan log kayıtlarının analiziyle minimum seviyeye getirilmiştir. Honeypot sunuculardan toplanan log kayıtları, çoğunlukla saldırı içeriğine sahip olduğu için log kayıtlarından imzalar üretilerek yeni saldırı örüntülerinin de imza tabanına eklenmesi sağlanmaktadır. Honeypot sunuculardan toplanan logların anormallik tespiti yöntemleri ile saldırı olmayan log kayıtlarından ayıklanması, başarımı arttırmıştır. Toplanan log kayıtları ile anormallik tespiti yöntemlerinin dezavantajlarından olan yüksek *yanlış alarm seviyesi* oldukça azaltılmış ve başarımı daha yüksek hibrit bir saldırı tespit yöntemi elde edilmiştir.

Ayrıca tez çalışması kapsamında, yukarıda anlatılan honeypot temelli yapı, güncel web açıklıklarına yönelik olarak başarılı bir biçimde uygulanmıştır. Gerçekleştirilen bu uygulama, PHP ve ASP.NET ile geliştirilmiş, web tabanlı sistemlere yapılabilecek olası saldırıları tespit edebilecek şekilde tasarlanmıştır. Honeypot temelli saldırı tespit ve engelleme sistemi ile birlikte kullanılan bu uygulama, düşük ve yüksek etkileşimli honeypot sistemlerin özelliklerini birlikte kullanarak başarımı arttırmıştır. Uygulamalar, gerçek zamanlı analizler sonucu, grafiksel kullanıcı arayüzleri ile sunulan sonuçlar sayesinde sistem yöneticilerine kolaylık sağlayabilecek niteliktedir. Geliştirilen *Honeypot-IDPS* mimarisi, imza tabanlı STS yapılarına entegre olabilecek şekilde tasarlanmıştır. Honeypot sistemlerin üzerine gelen trafiğin analiz edilmesi ile yeni saldırı imzaları üretilebilmektedir. Böylece sistem, sıfır gün ( zero-day) adı verilen, önceden bilinmeyen yeni saldırı türlerini de yakalayabilmektedir. Herhangi bir kötüye kullanım ya da anormallik tespiti durumunda paket iletimi kesilerek ağ, saldırılara karşı korunabilmektedir.

Geliştirilen yazılımlar, ağ trafiğini gerçek zamanlı olarak inceleyip yakaladığı paketlerden gelen isteklerin analizine göre *SQL Enjeksiyon, XSS, CSRF, Başlık Enjeksiyonu, LDAP Enjeksiyonu, Dizin Atlatma, Uzak/Yerel Dosya Ekleme, Hizmet Engelleme, Kaba Kuvvet* gibi saldırıları tespit edebilmektedir. Geliştirilen bu saldırı tespit ve engelleme sistemi yazılımları, yapısı itibariyle kural tabanlı ve anormallik tespit yaklaşımlarını birlikte sunan hibrit bir yapıda ve gerçek zamanlı olarak çalışan bir STS'dir. Gerçekleştirilen STS, kural tabanlı kötüye kullanım tespiti yaklaşımını kullandığı saldırı tespitlerinde %100 başarımla çalışmaktadır. Anormallik tespitine göre çalıştığı DoS benzeri saldırılarda ise değişen ağ trafik şartlarına göre

birçok parametreye göre kendini adapte etmek zorunda olduğundan başarımı daha düşüktür. Gerçekleştirilen saldırı tespit ve engelleme sistemi yazılımları, gerçek zamanlı veriler üzerinde çalışacak ve bilinen bütün saldırı türlerini tespit edebilecek bir seviyededir.

Geliştirilen honeypot sistem ile toplanan log kayıtlarının anormallik tespiti yöntemleri ile saldırı olmayan diğer kayıtlardan ayrıştırılması başarımı arttırmıştır. Bu log kayıtları ile STS'lerin dezavantajlarından olan yanlış alarm seviyesinin yüksek olması problemi çözülmüş olur. Çünkü honeypot sistemler genel itibariyle üzerinde kötücül aktiviteler barındırmaktadır. Böylece honeypota düşen saldırganların kimlikleri tespit edilip, gerçekleştirdikleri aktiviteler de analiz edilerek yeni saldırı örüntülerinin tespit edilmesi ve böylece yanlış alarm seviyesinin düşürülmesi mümkün olmaktadır.

Bu tez çalışması kapsamında ayrıca, bilgi sistemleri ve güvenliği açısından çok önemli bir yer teşkil eden log tutma ve tutulan log dosyalarının güvenli ve doğru bir biçimde saklanması, yetkisiz kişiler tarafından değiştirilmemesinin garanti edilmesi üzerinde durulmuştur. Yapılan çalışmada, log dosyalarının kaydedilmesinde bilgi güvenlik unsurlarını sağlayarak, adli bilişim sürecinde loglara objektif kanıt niteliği kazandırılmıştır. Yapılan çalışmada log kayıtlarının doğrulanması için kullanılan karşılaştırma işlemi belirlenen zaman periyotları ile yapılmaktadır. Fakat belirlenen zaman periyodu boyunca sistemin toplayacağı log dosyasının boyutu tahmin edilemeyebilir. Paralel olarak log kayıtlarının RAM'de tutulması bellek taşması gibi sorunlara neden olabilir. Bu sorunların önüne geçebilmek için zaman periyotları yerine bellek sınırı kullanılabilir. Bu şekilde uygulama yalnızca kendine tahsis edilen bellek üzerinde çalışarak sistem yükünü azaltacak ve ayrılan bellek kapasitesini doldurduğunda log dosyalarının doğrulanması için karşılaştırma işlemini gerçekleştirecektir. Karşılaştırma işleminden sonra tekrar tahsis edilmiş RAM bloğu üzerinde çalışmaya devam edecektir.

Log kayıtlarının doğrulanması için kullanılan karşılaştırma işlemi paralel programlama kullanılarak maksimum verim alınması amaçlanmıştır. Fakat ikili arama algoritması ile log kayıtlarının hash değerlerinin karşılaştırılarak var olan farkların bulunmasıyla yapılacak karşılaştırma sayısı en iyi durum için 1'e en kötü durum için ise n/2'ye düşecektir. Bu şekilde karşılaştırma işleminin maliyeti yarıya indirilebilir. Yapılan çalışmada, ağ katmanından IP protokolü ve taşıma katmanından UDP ve TCP protokollerinden log toplanmıştır. Uygulamaya diğer protokoller de tanıtılarak daha kapsamlı bir şekilde log toplanması sağlanabilir. Yapılan çalışma, olabilecek adli olaylarda kurumsal ve kişisel anlamda logların değişmediğinin ve delil olarak kullanılacak dijital bilgilerin güvenlik ve bütünlüğünün korunduğunu garanti eden bir yapı sunmaktadır.

Kurumsal ağlar ve honeypot tasarımlarının incelenmesi sonucu, VLAN konfigürasyonunun kullanıldığı ağ sistemlerinde her VLAN için en azından farklı bir ağ arayüzüne sahip bir makine kullanılması gerektiği tespit edilmiştir. Her VLAN için gerçek bir makine kullanılması, özellikle kampüs ağlarını da içeren geniş kurumsal ağlarda honeypotların tüm ağda uygulanması, kurulum ve bakım maliyetlerini arttırmaktadır. Bu nedenle, büyük ölçekli kurumsal ağlarda kurulum, bakım ve yönetim maliyetlerini azaltmak için bir ağ arayüzü ile honeypotların tüm ağda faaliyet göstermesini sağlayan bir merkezi yazılımsal anahtar uygulaması geliştirilmiştir. VLAN içeren kurumsal ağlarda katman 2 ve katman 3'ü dinleyebilen özgün bir yazılımsal anahtar tasarımı yapılarak VLAN ağlarının da dinlenebilmesi sağlanmıştır.

Gerçekleştirilen yazılımsal anahtar uygulaması sayesinde IPS ve honeypot sunucular kolaylıkla etkileşim kurabilmekte, konfigürasyon, bakım, ağ izleme, ağ analizi vb. faaliyetler kolay kullanımlı bir arayüz ile rahatlıkla yapılabilmektedir. Gerçekleştirilen yazılımsal anahtar uygulaması, saldırı çekme bileşeni olarak düşük etkileşimli *honeyd* yazılımını kullanan bir honeypot uygulaması ile özgün olarak geliştirilen ve SNORT kural tabanlarından yararlanan saldırı tespit ve engelleme sisteminin (IDPS) ağ güvenliğini sağlamak üzere kullanıldığı hibrit bir senaryoda uygulanmıştır. İlgili uygulamalar yukarıda anlatıldığı gibi saldırı tespiti işleminde kullanılmıştır.

Bu tez çalışması kapsamında yukarıda detaylarıyla anlatılan uygulamalar, geliştirilen Muxer uygulaması ile *dağıtık özel bulut* ağlarında da başarıyla uygulanmıştır. Geliştirilen Muxer uygulaması ile ağ analizinde kullanılabilecek bir merkezi işlem birimi altyapısı oluşturulmuştur. Ortaya konulan ağ altyapısı senaryoları ile uygulamanın kullanım kapsamı belirlenmiş ve yukarıda bahsedilen honeypot temelli saldırı tespit ve engelleme sistemleri, özel bulut sistemlerinde kullanılabilecek şekilde yeni bir bulut güvenlik mimarisinde birleştirilmiştir. Gelecek çalışmalarda Muxer uygulamasının üzerinden akan trafiği analiz ederek yeni saldırı yöntemlerinin tespit edilmesi ve aktarılan verilerin şifrelenerek loglanması sağlanabilecektir.

## Yayınlar

Bu tez çalışması kapsamında, 10 adet uluslararası konferans [6, 39, 85, 141-147], 1 adet uluslararası dergi [148] olmak üzere 11 adet makale yayınlanmıştır. Ayrıca SCI-Expanded kapsamındaki uluslararası dergilere 6 adet makale gönderilmiştir. Bu makale çalışmalarının değerlendirme süreçleri devam etmektedir.

## KAYNAKLAR


[1] 'Information', "Harrod's Librarians' Glossary and Reference Book", Tenth Edition Compiled by Ray Prytherch, p. 349, 2005.

[2] Sağsan, M., "Gelişmişliğin Vazgeçilmez Unsuru: Ulusal Bilgi Politikası", Stratejik Analiz (15), 2001.

[3] Vigo, R., "Representational information: A new general notion and measure of information", Information Sciences, 181, 21, pp.: 4847-4859, 2011, Elsevier.

[4] Naisbitt, J., "Megatrends", New York, Warner Books, s. 10, 1982.

[5] Hız, G., "Avrupa Birliği Bilgi Politikası: Gelişmiş Ülkelerdeki Bilgi Politikalarının Az Gelişmiş Ülkelere Işık Tutması", A. Yıldızeli, HK Bahşişoğlu, B. Bulgun, ve F. Başar (Yay. Haz), ÜNAK 5 (2005): 280-293.

[6] Baykara, M., Daş, R., Karadogan, İ., "Bilgi Güvenliği Sistemlerinde Kullanılan Araçların İncelenmesi", 1st International Symposium on Digital Forensics and Security (1. Uluslararası Adli Bilişim ve Güvenlik Sempozyumu)", 231-239, 20-21 Mayıs 2013, Elazığ - Turkey.

[7] Anderson, J.P. (1980). "Computer Security Threat Monitoring and Surveillance", Technical Report. James P. Anderson Co., Fort Washington, PA.

[8] Güven, E., N., "Zeki Saldırı Tespit Sistemlerinin İncelenmesi, Tasarımı ve Gerçekleştirilmesi", Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, 2007.

[9] Sancak, S., "Saldırı Tespit Sistemi Tekniklerinin Karşılaştırılması", Yüksek Lisans Tezi, Gebze Yüksek Teknoloji Enstitüsü, 2008.

[10] Sazli, H., M., Tanrıkulu, H., "Saldırı Tespit Sistemlerinde Yapay Sinir Ağlarının Kullanılması", XII. Türkiye'de İnternet Konferansı 8-10 Kasım, Ankara, 2007.

[11] Yuqing Mai; Upadrashta, R.; Xiao Su, "J-Honeypot: a Java-based network deception tool with monitoring and intrusion detection", Information Technology: Coding and Computing, Proceedings. ITCC 2004. International Conference on , vol.1, no., pp.804,808 Vol.1, 5-7 April 2004.

[12] Yun Yang; Jia Mi, "Design and implementation of distributed intrusion detection system based on honeypot", Computer Engineering and Technology (ICCET), 2010 2nd International Conference on , vol.6, no., pp.V6-260,V6-263, 16-18 April 2010.

[13] I. Mokube, M. Adams, "Honeypots: concepts, approaches, and challenges", Proceedings of the 45th Annual Southeast Regional Conference, pp. : 321– 326., March 2007.

[14] McGrew, R., "Experiences With Honeypot Sytems: Development, Deployment, and Analysis", Proceedings of the 39th Hawaii International Conference on System Sciences, vol.9, no., pp.220a,220a, 04-07 Jan. 2006.

[15] Gökırmak Y., Bektaş O., Soysal M., Yiğit S., "Sanal IPv6 Balküpü Ağı Altyapısı: Kovan", Ulusal IPv6 Konferansı, 2011.

[16] Gökırmak Y., Yüce E., Bektaş O., Soysal M., Orcan S., "IPv6 Balküpü Tasarımı", Tübitak Ulakbim, Ankara, 2011.

[17] Provols, N., Holz T., "Virtual Honeypots", Addison-Wesley Professional; 1 edition (July 26, 2007).

[18] Riden, J., "Using Nepenthes Honeypots to Detect Common Malware", 2007.

[19] Portokalidis, G., Slowinska, A., Bos, H., "Argos: an emulator for fingerprinting zero-day attacks for advertised honeypots with automatic signature generation", ACM SIGOPS Operating Systems Review. Vol. 40. No. 4. ACM, 2006.

[20] Balas, E., "Know Your Enemy: Sebek", A kernel based data capture tool, The Honeynet Project, 2003.

[21] Malanik D., Kouril L., "Honeypot as the Intruder Detection System", In Proceedings of the 17th WSEAS International Conference on Computer, Kos(GR), pp. 96-101, 2013.

[22] Riboldi Jordao da Silva Vargas, I.; Kleinschmidt, J.H., “Capture and Analysis of Malicious Traffic in VoIP Environments Using a Low Interaction Honeypot”, Latin America Transactions, IEEE (Revista IEEE America Latina), vol.13, no.3, pp. 777, 783, March 2015.
[23] Shukla, R., Singh, M., “PythonHoneyMonkey: Detecting malicious web URLs on client side honeypot systems”, 3rd International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), 2014, pp.1,5, 8-10 Oct. 2014.
[24] Koniaris, I., Papadimitriou, G., Nicopolitidis, P., Obaidat, M., “Honeypots deployment for the analysis and visualization of malware activity and malicious connections”, 2014 IEEE International Conference on Communications (ICC), pp.1819-1824, 10-14 June 2014.
[25] Song Li, Qian Zou, Wei Huang, “A new type of intrusion prevention system”, 2014 International Conference on Information Science, Electronics and Electrical Engineering (ISEEE), vol.1, no., pp.361-364, 26-28 April 2014.
[26] Chawda, K., Patel, A.D., “Dynamic & hybrid honeypot model for scalable network monitoring”, 2014 International Conference on Information Communication and Embedded Systems (ICICES), pp.1-5, 27-28 Feb. 2014.
[27] Xiangfeng Suo, Xue Han, Yunhui Gao, “Research on the application of honeypot technology in intrusion detection system”, 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), pp.1030-1032, 29-30 Sept. 2014.
[28] Paul, S., Mishra, B.K., “Honeypot based signature generation for defense against polymorphic worm attacks in networks” 2013 IEEE 3rd International Advance Computing Conference (IACC), pp.159-163, 22-23 Feb. 2013.
[29] Beham, M., Vlad, M., Reiser, H.P., “Intrusion detection and honeypots in nested virtualization environments”, 2013 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp.1-6, 24-27 June 2013.
[30] Liu Dongxia, Zhang Yongbo, “An Intrusion Detection System Based on Honeypot Technology”, 2012 International Conference on Computer Science and Electronics Engineering (ICCSEE), vol.1, pp.451-454, 23-25 March 2012.
[31] Pomsathit, A., “Effective of Unicast and Multicast IP Address Attack over Intrusion Detection System with Honeypot”, 2012 Spring Congress on Engineering and Technology (S-CET), pp.1-4, 27-30 May 2012.
[32] Jiang Zhen, Zhenxiang Liu, “New honeypot system and its application in security of employment network”, 2012 IEEE Symposium on Robotics and Applications (ISRA), pp.627-629, 3-5 June 2012.
[33] Akiyama, M., Kawakoya, Y., Hariu, T., “Scalable and Performance-Efficient Client Honeypot on High Interaction System”, 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet (SAINT), pp.40-50, 16-20 July 2012.
[34] Buvaneswari, M., Subha, T., “IHoneycol: A distributed collaborative approach for mitigation of DDoS attack”, International Conference on Information Communication and Embedded Systems (ICICES), pp.340-345, 21-22 Feb. 2013.
[35] Mehta, V., Bahadur, P., Kapoor, M., Singh, P., Rajpoot, S., "Threat prediction using honeypot and machine learning," International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), pp.278-282, 25-27 Feb. 2015.
[36] Li Li., Sun Hua., Zhenyu Zhang., "The research and design of honeypot system applied in the LAN security," IEEE 2nd International Conference on Software Engineering and Service Science (ICSESS), pp.360-363, 15-17 July 2011.
[37] Parziale, Lydia, et al. TCP/IP Tutorial and Technical Overview. s.l.: IBM, 2006. IBM Redbooks.

[38] Wei Chen, Dit-Yan Yeung, "Defending Against TCP SYN Flooding Attacks Under Different Types of IP Spoofing", International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, ICN/ICONS/MCL2006, pp.38-38, 23-29 April 2006.
[39] Karadoğan, İ., Daş, R., Baykara, M. "Scapy ile Ağ Paket Manipülasyonu", 1. Uluslararası Adli Bilişim ve Güvenlik Sempozyumu. 2013, s. 196-201.
[40] Ahsan, Kamran. Covert Channel Analysis and Data Hiding in TCP/IP. University of Toronto. 2002.
[41] Önal, H., "TCP/IP Ağlarda Parçalanmış Paketler - Parçalanmış Paketler ve Güvenlik Sistemlerine Etkileri", 2009.
[42] Millî Eğitim Bakanlığı. Meslekî ve Teknik Eğitim Programlar ve Öğretim Materyalleri. Elektrik Elektronik Teknolojisi, Ağ Güvenliği ve Ağ Protokolleri- 481BB0007, 2011.
[43] Vacca, John R. Computer and Information Security Handbook. s.l.: Morgan Kaufman Publishers, 2009.
[44] Orebaugh, Angela, et al. Wireshark & Ethereal Network Protocol Analyzer Toolkit. s.l.: Syngress Publishing, 2007. ISBN-13: 978-1-59749-073-3.
[45] Orebaugh, A., Pinkard B., "Nmap in the enterprise: your guide to network scanning", Syngress, 2011.
[46] Matsunaka, T., Yamada, A., Kubota, A., "Passive OS Fingerprinting by DNS Traffic Analysis", 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), pp.243-250, 25-28 March 2013.
[47] Fussell, R.S., "Protecting information security availability via self-adapting intelligent agents", Military Communications Conference, IEEE, 2977s., 2005.
[48] Tekerek M. 2008, Bilgi Güvenliği Yönetimi, KSÜ Fen ve Mühendislik Dergisi 11(1).
[49] Marcinkowski, S.J., Stanton, J.M., "Motivational aspects of information security policies", IEEE International Conference on Systems, Man and Cybernetics, 3: 2528s., 2003.
[50] Campbell, S., "Supporting digital signatures in mobile environments. Enabling Technologies: Infrastructure for Collaborative Enterprises", Twelfth IEEE International Workshops Proceedings, 238s., 2003.
[51] Shephard, B., "Information security-whocares?", Fifth International Conference on Power System Management and Control, (Conf. Publ. No. 488), 126s., 2002.
[52] Moffett, J. 1990. Network security management. Security and Networks, IEE Colloquiumon, 4-6.
[53] Karaarslan E., Teke A., Şengonca H. 2003. Bilgisayar Ağlarında Güvenlik Politikalarının Uygulanması. Akademik Bilişim, Çukurova Üniversitesi, 1s.
[54] Anonim, Bilişim Güvenliği Kitapçığı. ProG Bilişim Güvenliği ve Araştırma Ltd, 7s., 2003.
[55] Nen-Fu Huang, Chuang Wang, I-Ju Liao, Che-Wei Lin, Chia-Nan Kao, "An OpenFlow-based collaborative intrusion prevention system for cloud networking", 2015 IEEE International Conference on Communication Software and Networks (ICCSN), pp.85-92, 6-7 June 2015.
[56] Tan H., Aktaş Z. A. 2011 Bir Kuruluşun Bilgi Sistemi Güvenliği İçin Bir Yaklaşım, 4. Ağ ve Bilgi Güvenliği Sempozyumu, Ankara.
[57] Sommerville, I., "Software Engineering", Addison-Wesley, 2007.
[58] Datasheet, "FireEye Web Malware Protection System", FireEye, Inc., 2011.
[59] Cascade® Pilot Reference Manual, Version 10.0.6, Mart 2013.
[60] Kevin J. Connolly (2003). Law of Internet Security and Privacy. Aspen Publishers. p. 131. ISBN 978-0-7355-4273-0.
[61] Slay, J., Turnbull, B., "The Uses and Limitations of Unidirectional Network Bridges in a Secure Electronic Commerce Environment", INC 2004 Conference, Plymouth, UK, 6–9 July 2004.

[62] Pasin, Ş., "Çok Algılayıcılı Saldırı Tespit Sistemleri", Gebze Yüksek Teknoloji Enstitüsü, Bilgisayar Mühendisliği Bölümü, Veri ve Ağ Güvenliği Dersi Projesi, 2002.
[63] Dayıoğlu, B., Özgit, A., "Internet'de Saldırı Tespiti Teknolojileri", İletişim Teknolojileri I. Sempozyumu ve Fuarı, 17-21 Ekim 2001, Ankara.
[64] Weijian Huang; Yan An; Wei Du, "A Multi-Agent-Based Distributed Intrusion Detection System", 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), vol.3, no., pp.V3-141-V3-143, 20-22 Aug. 2010
[65] E. Biermann, E. Cloete, L. M. Venter, "A Comparison of Intrusion Detection Systems", Computers & Security, vol. 20 (2001), pp. 676-683.
[66] Depren, O., Topallar, M., Anarim, E., Ciliz, M.K., "An Intelligent Intrusion Detection System (IDS) for Anomaly and Misuse Detection in Computer Networks", Expert System with Application vol. 29 (2005), pp. 713-722.
[67] Patcha, A., Park, J.M., "An overview of anomaly detection techniques: Existing solutions and latest technological trends", Computer Networks, 51(12): pp. 3448-3470 (2007).
[68] Denning, D.E., "An intrusion detection model", IEEE Transactions on Software Engineering, 13(2): 118–131 (1987).
[69] Abd Almonim Nour Albashir, A., "Detecting unknown vulnerabilities using Honeynet," First International Conference on Anti-Cybercrime (ICACC), pp.1-4, 10-12 Nov. 2015.
[70] G. Canbek, Ş. Sağıroğlu, "Bilgisayar Sistemlerine Yapılan Saldırılar ve Türleri: Bir İnceleme", Erciyes Üniversitesi Fen Bilimleri Enstitüsü Dergisi 23(1-2)1-12(2007).
[71] Chimphlee, W., Abdullah, A.H., Noor Md Sap, M., Srinoy, S., Chimphlee, S., "Anomaly-Based Intrusion Detection using Fuzzy Rough Clustering", ICHIT '06. International Conference on Hybrid Information Technology, vol.1, no., pp.329-334, 9-11 Nov. 2006.
[72] Waksman, A., Sethumadhavan, S., "Silencing Hardware Backdoors", 2011 IEEE Symposium on Security and Privacy (SP), pp.49-63, 22-25 May 2011.
[73] Wonghirunsombat, E., Asawaniwed, T., Hanchana, V., Wattanapongsakorn, N., Srakaew, S., Charnsripinyo, C., "A centralized management framework of network-based Intrusion Detection and Prevention System", 10th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp.183-188, 29-31 May 2013.
[74] Kumar, G., Ahmad, K., Saurabh, A.K., Doja, M.N., "Data prevention from unauthorized access by Unclassified Attack in Data Warehouse", 2014 International Conference on Computing for Sustainable Global Development (INDIACom), pp.734-738, 5-7 March 2014.
[75] İlktan, A.R., "Nüfuz Tespit Sistemleri", 2010.
[76] Özgür, A., Erdem, H., "Saldırı Tespit Sistemlerinde Kullanılan Kolay Erişilen Makine Öğrenme Algoritmalarının Karşılaştırılması", Bilişim Teknolojileri Dergisi, Cilt:5, Sayı:2, Mayıs 2012.
[77] Cisco 2014 Annual Report, Cisco Systems, Inc. San Jose, CA.
[78] Karen S., Peter M., "Guide to Intrusion Detection and Prevention Systems", NIST Special Publication 800-94, 2007.
[79] Zemene, M.S., Avadhani, P.S., "Implementing high interaction honeypot to study SSH attacks", International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp.1898-1903, 10-13 Aug. 2015.
[80] Mudzingwa, D., Agrawal, R., "A study of methodologies used in intrusion detection and prevention systems (IDPS)", in Southeastcon, 2012 Proceedings of IEEE , vol., no., pp.1-6, 15-18 March 2012.
[81] Erol M., "Saldırı Tespit Sistemlerinde İstatistiksel Anormallik Belirleme Kullanımı", 2005.
[82] Paxson V., "Bro: A System for Detecting Network Intruders in Real-Time", Lawrence Berkeley National Laboratory, 1998.

[83] Özhan, E., “Paket ve Port Analizi İle Ağ Saldırı Tespit Sistemleri”, Yüksek Lisans Tezi, Bilgisayar Mühendisliği Anabilim Dalı, Trakya Üniversitesi Fen Bilimleri Enstitüsü, 2006.
[84] Joshua S., Whitea, Thomas T., Fitzsimmonsb, Jeanna N. Matthewsc, “Quantitative Analysis of Intrusion Detection Systems: Snort and Suricata”, Wallace H. Coulter School of Engineering Department of Computer Science, 2011.
[85] Demirol D., Daş R., Baykara M., "SQL Enjeksiyon Saldırı Uygulaması ve Güvenlik Önerileri", 1.Uluslararası Adli Bilişim ve Güvenlik Sempozyumu, 20-22 Mayıs 2013, Fırat Üniversitesi, Elazığ.
[86] Vural, Y., Sağıroğlu, Ş., “Kurumsal Bilgi Güvenliği ve Standartları üzerine bir İnceleme”, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, Cilt 23, No 2, s.507-522, Haziran 2008.
[87] Jovanovic, N., Kirda, E., Kruegel, C., “Preventing Cross Site Request Forgery Attacks”, In Securecomm and Workshops, pp.1-10, Aug. 28 -Sept. 1 2006.
[88] Kavak İ., Türker G. F., “LDAP ile Güvenli Kullanıcı Kontrol Sistemi”, Akademik Bilişim 2014, Mersin.
[89] Bernd-Jan de Rooij, “Augmented Reality & E-Learning Possible Solutions For The Nautical Industry”, VDR-Ballastwasser-Symposium, 8-9 September 2015.
[90] Hekim H., Başıbüyük O., “Siber Suçlar ve Türkiye’nin Siber Güvenlik Politikaları”, Uluslararasi Güvenlik ve Terörizm Dergisi, 4.2 (2013).
[91] Pfleeger, C.P., "The fundamentals of information security", Software, IEEE , vol.14, no.1, pp.15,16, 60, Jan/Feb 1997.
[92] Canbek G., Sağıroğlu Ş., ”Bilgisayar sistemlerine yapılan saldırılar ve türleri: Bir inceleme”, Erciyes Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 23.1-2, s.1-12, 2007.
[93] Demirol, D., Web Erişim Kütüklerinin Temizlenmesine Yönelik Yazılım Geliştirme, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2013.
[94] Henkoğlu, T., Külcü, Ö., ”Bilgi Erişim Platformu Olarak Bulut Bilişim: Riskler ve Hukuksal Koşullar Üzerine Bir İnceleme”, Bilgi Dünyası 14.1 (2013): 62-86.
[95] “Bulut Bilişim Güvenlik ve Kullanım Standardı”, Türk Standardları Enstitüsü (TSE), Mart 2014.
[96] Selvi, O., Küçüksille, U., “Bulut Ortamında Adli Bilişim”, 6. Uluslararası Bilgi Güvenliği ve Kriptoloji Konferansı, 20-21 Eylül 2013, Ankara.
[97] Yüksel, H., “Bulut Bilişim El Kitabı”, Çözümpark, Ocak 2012.
[98] Kumar, M., Mathur, R., “Unsupervised outlier detection technique for intrusion detection”, International Conference for cloud computing in Convergence of Technology (I2CT), 2014.
[99] Ficco, M., Tasquier, L., Aversa, R., "Intrusion Detection in Cloud Computing", Eighth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013.
[100] Yassin, W., Udzir, N.I., Muda, Z., Abdullah, A., Abdullah, M.T., "A Cloud-based Intrusion Detection Service framework", International Conference on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec), 2012.
[101] Han Li., Qiuxin Wu, "A distributed intrusion detection model based on cloud theory" IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS), 2012.
[102] Mehmood, Y., Habiba, U., Shibli, M.A., Masood, R., "Intrusion Detection System in Cloud Computing: Challenges and opportunities", 2nd National Conference on Information Assurance (NCIA), 2013.
[103] Nikolai, J., Yong Wang, "Hypervisor-based cloud intrusion detection system", International Conference on Computing, Networking and Communications (ICNC), 2014.

[104] Zhe Li., Weiqing Sun., Lingfeng Wang, "A neural network based distributed intrusion detection system on cloud platform", IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS), 2012.
[105] Vanathi, R., Gunasekaran, S., "Comparison of Network Intrusion Detection Systems in cloud computing environment", International Conference on Computer Communication and Informatics (ICCCI), 2012.
[106] Mohamed, H., Adil, L., Saida, T., Hicham, M., "A collaborative intrusion detection and Prevention System in Cloud Computing", in AFRICON, 2013.
[107] Donadio, Pasquale, "Virtual intrusion detection systems in the cloud", Bell Labs Technical Journal, 2012.
[108] Roschke, S., Feng Cheng, Meinel, C., "Intrusion Detection in the Cloud", DASC '09. Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2009.
[109] Jun-Ho Lee, Min-Woo Park, Jung-Ho Eom, Tai-Myoung Chung, "Multi-level Intrusion Detection System and log management in Cloud Computing", 13th International Conference on Advanced Communication Technology (ICACT), 2011.
[110] Gupta, S.; Kumar, P., Sardana, A., Abraham, A., "A fingerprinting system calls approach for intrusion detection in a cloud environment", Fourth International Conference on Computational Aspects of Social Networks (CASoN), 2012.
[111] Aljurayban, N.S., Emam, A., "Framework for cloud intrusion detection system service", 2nd World Symposium on Web Applications and Networking (WSWAN), 2015.
[112] Sijin He; Ghanem, M., Li Guo, Yike Guo, "Cloud Resource Monitoring for Intrusion Detection", IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom), 2013.
[113] Modi, C.N., Patel, D., "A novel hybrid-network intrusion detection system (H-NIDS) in cloud computing", IEEE Symposium on Computational Intelligence in Cyber Security (CICS), 2013.
[114] Kumar, N., Sharma, S., "Study of intrusion detection system for DDoS attacks in cloud computing", Tenth International Conference on Wireless and Optical Communications Networks (WOCN), 2013.
[115] Xin Wang, Huang Ting-lei, Liu Xiao-yu, "Research on the Intrusion detection mechanism based on cloud computing", International Conference on Intelligent Computing and Integrated Systems (ICISS), 2010.
[116] Deguang Wang, Zhigang Zhou, "Application of Cloud Model in Intrusion Detection", 2nd International Conference on e-Business and Information System Security (EBISS), 2010.
[117] Chi-Chun Lo, Chun-Chieh Huang, Ku, J., "A Cooperative Intrusion Detection System Framework for Cloud Computing Networks", 39th International Conference on Parallel Processing Workshops (ICPPW), 2010.
[118] Mauro Andreolini, Michele Colajanni, Mirco Marchetti, "A collaborative framework for intrusion detection in mobile networks", Information Sciences, Volume 321, Pages 179-192, 10 November 2015.
[119] Chirag Modi, Dhiren Patel, Bhavesh Borisaniya, Hiren Patel, Avi Patel, Muttukrishnan Rajarajan, "A survey of intrusion detection techniques in Cloud", Journal of Network and Computer Applications, Volume 36, Issue 1, Pages 42-57, January 2013.
[120] Donadio, Pasquale, "Virtual intrusion detection systems in the cloud", in Bell Labs Technical Journal, vol.17, no.3, pp.113-128, Dec. 2012.
[121] Ryoo, J., Rizvi, S., Aiken, W., Kissell, J., "Cloud Security Auditing: Challenges and Emerging Approaches", in *Security & Privacy, IEEE*, vol.12, no.6, pp.68-74, Nov.-Dec. 2014.

[122] Tari, Z., Xun Yi, Premarathne, U.S., Bertok, P., Khalil, I., "Security and Privacy in Cloud Computing: Vision, Trends, and Challenges", in Cloud Computing, IEEE, vol.2, no.2, pp.30-38, Mar.-Apr. 2015.
[123] Varadharajan, V., Tupakula, U., "Security as a Service Model for Cloud Environment", in Network and Service Management, IEEE Transactions on , vol.11, no.1, pp.60-75, March 2014.
[124] Chen, Zhen, Han, Fuye, Cao, Junwei, Jiang, Xin, Chen, Shuo, "Cloud computing-based forensic analysis for collaborative network security management system", in Tsinghua Science and Technology, vol.18, no.1, pp.40-50, Feb. 2013.
[125] Xuexiu Chen, Chi Chen, Yuan Tao, Jiankun Hu, "A Cloud Security Assessment System Based on Classifying and Grading", in Cloud Computing, IEEE, vol.2, no.2, pp.58-67, Mar.-Apr. 2015.
[126] Kaufman, L.M., "Data Security in the World of Cloud Computing", in Security & Privacy, IEEE , vol.7, no.4, pp.61-64, July-Aug. 2009.
[127] Choo, K.-K.R., "A Cloud Security Risk-Management Strategy", in Cloud Computing, IEEE, vol.1, no.2, pp.52-56, July 2014.
[128] Qian Wang, Cong Wang, Kui Ren, Wenjing Lou, Jin Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", in Parallel and Distributed Systems, IEEE Transactions on, vol.22, no.5, pp.847-859, May 2011.
[129] Gongjun Yan, Ding Wen, Olariu, S., Weigle, M.C., "Security challenges in vehicular cloud computing", in Intelligent Transportation Systems, IEEE Transactions on , vol.14, no.1, pp.284-294, March 2013.
[130] Zhifeng Xiao, Yang Xiao, "Security and Privacy in Cloud Computing", in Communications Surveys & Tutorials, IEEE, vol.15, no.2, pp.843-859, Second Quarter 2013.
[131] Madria, S., Sen, A., "Offline Risk Assessment of Cloud Service Providers", in Cloud Computing, IEEE, vol.2, no.3, pp.50-57, May-June 2015.
[132] Padilha, R., Pedone, F., "Confidentiality in the Cloud", in Security & Privacy, IEEE, vol.13, no.1, pp.57-60, Jan.-Feb. 2015.
[133] Gupte, N., Jia Wang, "Secure Power Grid Simulation on Cloud", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol.34, no.3, pp.422-432, March 2015.
[134] Karnad, K., Nagenthram, S., "Cloud security: Can the cloud be secured?", 2012 International Conference for Internet Technology and Secured Transactions, pp.208-210, 10-12 Dec. 2012.
[135] Lin Chen, Xingshu Chen, Junfang Jiang, Xueyuan Yin, Guolin Shao, "Research and practice of dynamic network security architecture for IaaS platforms", in Tsinghua Science and Technology, vol.19, no.5, pp.496-507, Oct. 2014.
[136] Ficco, M., Rak, M., "Stealthy Denial of Service Strategy in Cloud Computing", IEEE Transactions on Cloud Computing, vol.3, no.1, pp.80-94, Jan.-March 1 2015.
[137] Idziorek, Joseph, Tannian, Mark F., Jacobson, Doug, "The Insecurity of Cloud Utility Models", in IT Professional, vol.15, no.2, pp.22-27, March-April 2013.
[138] Salah, K., Alcaraz Calero, J.M., Zeadally, S., Al-Mulla, S., Alzaabi, M., "Using Cloud Computing to Implement a Security Overlay Network", in *Security & Privacy, IEEE* , vol.11, no.1, pp.44-53, Jan.-Feb. 2013.
[139] Amoroso, E.G., "From the Enterprise Perimeter to a Mobility-Enabled Secure Cloud", in *Security & Privacy, IEEE* , vol.11, no.1, pp.23-31, Jan.-Feb. 2013.
[140] Zhiyuan Tan, Nagar, U.T., Xiangjian He, Nanda, P., Ren Ping Liu, Song Wang, Jiankun Hu, "Enhancing Big Data Security with Collaborative Intrusion Detection", in *Cloud Computing, IEEE* , vol.1, no.3, pp.27-33, Sept. 2014.

[141] Baykara, M., Daş, R., "A Steganography Application for Secure Data Communication", 10th International Conference on Electronics, Computer and Computation (ICECCO 2013), Turgut Özal University, pp.313-317, 7-9 November 2013, Ankara.
[142] Bürhan, Y., Daş, R., Baykara, M., "Sosyal Ağ Ortamlarında Karşılaşılan Tehditlerin Analizi", The Third International Symposium on Digital Forensics and Security (ISDFS 2015), pp.194-200, 11-12 May, 2015, Gazi University, ANKARA.
[143] Daş, R., Baykara, M., Tuna, G., (2015), "Novel CAPTCHA Approaches to Protect Web Forms against Bots", The Third International Symposium on Digital Forensics and Security (ISDFS 2015), pp.85-88, 11-12 May, 2015, Gazi University, ANKARA.
[144] Tuna G. , Örenbaş H. , Daş R. , Kogias D. , Baykara M. , Gulez K., "Information security threats and an easy-to-implement attack detection framework for wireless sensor network-based smart grid applications", 5th International Conference on Materials and Applications for Sensors and Transducers, September 27-30, 2015, Mykonos, Greece.
[145] Tuna G. , Daş R. , Tuna A. , Örenbaş H. , Baykara M. , Gulez K., "A secure and easy-to-implement web-based communication framework for caregiving robot teams", 5th International Conference on Materials and Applications for Sensors and Transducers, September 27-30, 2015, Mykonos, Greece.
[146] Baykara, M., Daş, R., "A Survey On Honeypot Technologies Used In Intrusion Detection System", 16th International Conference on Science and Innovative Engineering (ICSIE), , pp. 14-19 November 2015, Prague.
[147] Baykara, M., Daş, R., "Real Time Face Recognition and Tracking System", 10th International Conference on Electronics, Computer and Computation (ICECCO 2013), Turgut Özal University, pp.163-167, 7-9 November 2013, Ankara.
[148] Baykara, M., Daş, R., "A Survey on Potential Applications of Honeypot Technology in Intrusion Detection Systems", International Journal of Computer Networks and Applications (IJCNA), 2(5), 203-211, 2015.
[149] Chebrolu S., Abraham A. and Thomas J.P, 2005, "Feature deduction and ensemble design of intrusion detection systems", Computers & Security, Volume 24, Issue 4, pp.295-307.
[150] Aickelin and Greensmith, 2007, "Sensing danger: Innate immunology for intrusion detection", Information Security Technical Report, Vol. 12 Issue 4, pp. 218-227.
[151] Axelsson, S., "Intrusion detection systems: A survey and taxonomy", Technical Report 99-15, Dept. of Computer Eng., Chalmers University of Technology, Göteborg, Sweden, 1-23 (2000).
[152] Michael E. Whitman; Herbert J. Mattord (2009), "Principles of Information Security", Cengage Learning EMEA. ISBN 978-1-42.
[153] Smaha, S.E., "Haystack: an intrusion detection system", In Aerospace Computer Security Applications Conference, 1988., Fourth, pp.37-44, 12-16 Dec 1988.
[154] Lunt, T.F., Tamaru, A., Gilham, F., Jagannathan, R., Neumann, P.G., Jalali, C., "IDES: a progress report [Intrusion-Detection Expert System]", Proceedings of the Sixth Annual Computer Security Applications Conference, 1990., pp.273-285, 3-7 Dec 1990.
[155] Peddabachigari, S., Abraham, A., Grosan, C., Thomas, J., "Modeling intrusion detection system using hybrid intelligent systems", Journal of Network and Computer Applications, Elsevier, 30:114–132 (2007).
[156] Vaccarro, H. S. ve Liepins, G. E. "Detection of Anomalous Computer Session Activity", IEEE Symposium on Research in Security and Privacy, Oakland, California, 280-289 (1989).
[157] Christoph, G.G., Jackson, K.A., Neuman, M.C., Siciliano, C.L.B., Simmonds, D.D., Stallings, C.A., Thompson, J.L., "UNICORN: Misuse Detection for UNICOS", Proceedings of the 1995 ACM/IEEE conference on Supercomputing, San Diego, California, United States, 56-80 (1995).

[158] Ilgun, K., "Ustat: A real-time intrusion detection system for Unix", Proceedings of the 1993 IEEE Symposium on Research in Security and Privacy, Oakland, California, 16-28, (1993).
[159] SNORT Users Manual Version 2.9.7, 2014, ( http://manual.snort.org/).
[160] McRee R., "BroIDS with Logstash and Kibana", pp. 37-39, ISSA Journal, September 2013.
[161] Bray R., Cid D., Hay A., "OSSEC Host-Based Intrusion Detection Guide", Foreword by Stephen Northcutt, President The SANS Technology Institute, 2008.
[162] Manev, P., "Windows Installation Guide for Suricata IDS/IPS", Open Information Security Foundation, Document Version 1.3, 2012.
[163] Azizul, I., Manirul, I., "A Novel Signature-Based Traffic Classification Engine To Reduce False Alarms In Intrusion Detection Systems", International Journal of Computer Networks & Communications (IJCNC) Vol.7, No.1, January 2015.
[164] Phpids, PHP-Intrusion Detection System, https://github.com/PHPIDS/PHPIDS , (Erişim Tarihi: 27.10.2015).
[165] .NETIDS, Intrusion detection system for .NET based on phpids, https://code.google.com/p/dotnetids/, (Erişim Tarihi: 27.10.2015).
[166] Leach J., Tedesco G., "Firestorm Network Intrusion Detection System", Firestorm Documentation 2003.
[167] İnternet: "DARPA Intrusion Detection Data Sets", http://www.ll.mit.edu/ideval/data/, (Erişim Tarihi: 25.10.2015).
[168] Kavitha, K., "Study on Cloud Computing Model and its Benefits, Challenges", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 1, pp. 2423-2431, 2014.

# ÖZ GEÇMİŞ

**Muhammet BAYKARA**

Fırat Üniversitesi

Teknoloji Fakültesi

Yazılım Mühendisliği

23119, Elazığ

Tel: 0 424 – 2370000 - 4318

E-posta: mbaykara@firat.edu.tr

| | |
|---|---|
| 2001-2006 | Fırat Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü'nden mezun oldu. |
| 2005-2007 | Elazığ Belediyesi Bilgi İşlem Müdürlüğü'nde çalıştı. |
| 2007 | Fırat Üniversitesi Rektörlüğü Bilgi İşlem Daire Başkanlığı'nda Okutman olarak göreve başladı. |
| 2008-2011 | Fırat Üniversitesi Maden Meslek Yüksekokulu Bilgisayar Programcılığı'nda Okutman olarak görev yaptı. |
| 2009 | Fırat Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında yüksek lisans eğitimini tamamladı. |
| 2011- | Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü'nde Araştırma Görevlisi olarak çalışmaktadır. |
| 2012- | Fırat Üniversitesi Fen Bilimleri Enstitüsü Yazılım Mühendisliği Anabilim Dalında Doktora eğitimine devam etmektedir. |