HAVA H ARP OKULU
HAVACILIK VE UZAY TEKNOLOJİLERİ
ENSTİTÜSÜ

BULUT BİLİŞİMDE VEKİL AĞ SALDIRI TESPİT SİSTEMİ

YÜKSEK LİSANS TEZİ

Uçman OKTAY

Bilgisayar Mühendisliği Ana Bilim Dalı Başkanlığı

Yazılım Mühendisliği Programı

HAZİRAN 2013

HAVA HARP OKULU
HAVACILIK VE UZAY TEKNOLOJİLERİ ENSTİTÜSÜ

BULUT BİLİŞİMDE VEKİL AĞ SALDIRI TESPİT SİSTEMİ

YÜKSEK LİSANS TEZİ

Uçman OKTAY
(111102)

Bilgisayar Mühendisliği Ana Bilim Dalı Başkanlığı

Yazılım Mühendisliği Programı

Tez Danışmanı: Yrd.Doç.Dr. Özgür Koray ŞAHİNGÖZ

HAZİRAN 2013

Hava Harp Okulu Havacılık ve Uzay Teknolojileri Enstitüsünün 111102 numaralı Yüksek Lisans Öğrencisi **Uçman OKTAY**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı **"BULUT BİLİŞİMDE VEKİL AĞ SALDIRI TESPİT SİSTEMİ"** başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı :	**Yrd.Doç.Dr. Özgür Koray ŞAHİNGÖZ** Hava Harp Okulu	.....
Jüri Üyeleri :	**Doç.Dr. Olcay KURŞUN** İstanbul Üniversitesi	.....
	Yrd.Doç.Dr. Muhammed Ali AYDIN İstanbul Üniversitesi	.....

Teslim Tarihi : 31 Mayıs 2013
Savunma Tarihi : 20 Haziran 2013

Bu tez çalışmasında belirtilen görüş ve yorumlar yazara aittir. Türk Silahlı Kuvvetleri'nin ya da diğer kamu kuruluşlarının görüşlerini yansıtmaz. Ayrıca bu tez çalışması bilimsel ahlak ve etik değerlere uygun olarak yazılmış olup, yararlanılan tüm eserler kaynaklarda gösterilmiştir.

Haziran 2013 Uçman OKTAY

Aileme,

ÖNSÖZ

Tez çalışmamı hazırlama aşamasında bana her konuda yardımcı olan, her soruma sabırla yanıt veren ve tez çalışmasının eksik yönlerinin ortaya çıkarılması ve giderilmesi kapsamında değerli katkılarda bulunan tez danışmanım Yrdc.Doç.Dr. Özgür Koray ŞAHİNGÖZ'e, başta Bilgisayar Mühendisliği Ana Bilim Dalı Başkanı Yrdc.Doç.Dr. Güray YILMAZ olmak üzere, Bilgisayar Mühendisliği Ana Bilim Dalı Başkanlığı öğretim üyelerine, yüksek lisans tez çalışmam esnasında engin bilgilerinden faydalandığım hocam Yrdc.Doç.Dr. Muhammed Ali AYDIN'a, tez çalışmamı yaparken gerekli teknik altyapıyı sağlayan başta H2B Internet Technologies CEO'su Sayın Harun Barış BULUT olmak üzere, tüm H2B Internet Technologies çalışanlarına, çalışmam süresince büyük ilgi ve fedakârlıklarıyla her zaman yanımda olan aileme, bana bu imkânı sağlayan ülkem ve Türk Silahlı Kuvvetleri'ne teşekkürlerimi sunmayı bir borç bilir, sonsuz minnettarlığımı sunarım.

Haziran 2013 Uçman OKTAY

İÇİNDEKİLER

Sayfa

KISALTMALAR

AKK : Açık Kaynak Kodlu
ASCII : American Standard Code for Information Interchange
BASE : Basic Analysis and Security Engine
CaaS : Communication as a Service
CCIE : Cisco Certified Internet Engineer
CCNA : Cisco Certified Network Associate
CCNP : Cisco Certified Network Professional
CEH : Certified Ethical Hacker
CentOS : Community Enterprise Operating System
CISA : Certified Information Systems Auditor
CISSP : Certified Information Systems Security Professional
C-MaaS : Cloud Migration as a Service
COBIT : Control Objectives for Information and Related Technology
CPU : Central Processing Unit
DDR3 : Double Data Rate Type 3
DMA : Direct Memory Access
DMZ : Demilitarized Zone
Dom0 : Domain 0
DoS : Denial of Service
DDoS : Distributed Denial of Service
ENISA : European Network and Information Security Agency
e-posta : Elektronik Posta
FTP : File Transfer Protocol
GB : Gigabayt
Ghz : Gigahertz
GIAC : Global Information Assurance Certification
GPL : General Public License
GPRS : General Packet Radio Service
GPS : Global Positioning System
HIDS : Host-based Intrusiton Detection System
HTTP : Hypertext Transfer Protocol
ICMP : Internet Control Message Protocol
IaaS : Infrastructure as a Service
ID : Identification
IDC : Internet Data Corporation
IO : Input/Output
IP : Internet Protocol
ISIC : IP Stack Integrity Checker
ISO : International Organization for Standardization
IT : Information Technologies
ITIL : The Information Technology Infrastructure Library
IDS : Intrusion Detection System
IDPS : Intrusion Detection and Prevention System

kb **:** Kilobayt
KMD **:** Kernel Monitor Daemon
JAAS **:** Java Authentication Authorization Service
LAN **:** Local Area Network
MaaS **:** Monitoring as a service
MAC **:** Media Access Control
MB **:** Megabayt
NaaS **:** Network as a Service
NAPT **:** Network Address and Port Translation
NBAD **:** Network Behaviour Anomally Detection
NDA **:** Non-Disclosure Agreement
NIDS **:** Network Based Intrusion Detection System
NIST **:** National Institue of Standards and Technology
OS **:** Operating System
OSI **:** Open Systems Interconnection
OSSEC **:** Opearting System Security
PaaS **:** Platform as a Service
PARC **:** Palo Alto Research Center
PID **:** Process Identification
PVI **:** Private Virtual Infrastructure
RAM **:** Random Access Memory
RHEL **:** Red Hat Enterprise Linux
SaaS **:** Software as a Service
SaaS **:** Storage as a Service
SMTP **:** Simple Mail Transfer Protocol
SNMP **:** Simple Network Management Protocol
TC **:** Trusted Computing
TCP **:** Transmission Control Protocol
TCP/IP **:** Transmission Control Protocol/Internet Protocol
TPM **:** Trusted Platform Module
UDP **:** User Datagram Protocol
VM **:** Virtual Machine
VMI **:** Virtual Machine Introspection
VMM **:** Virtual Machine Monitor
VLAN **:** Virtual Local Area Network
VPN **:** Virtual Private Network
VSFTPD **:** Very Secure FTP Daemon
VTPM **:** Virtual Trusted Platform
XCP **:** Xen Cloud Platform

ÇİZELGE LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

BULUT BİLİŞİMDE VEKİL AĞ SALDIRI TESPİT SİSTEMİ

ÖZET

Bulut bilişim, bilgi teknolojilerinde kullanılmakta olan en yeni yaklaşımlardan biridir. Sistem yöneticileri geleneksel bilgi merkezlerinden bulut bilişime geçişte maliyetleri gözetmek durumundadır. Güvenlik, geleneksel bilgi teknolojilerinde olduğu gibi bulut bilişim yaklaşımının da en büyük sorunlarından biridir. Antivirüs yazılımları, güvenlik duvarları, bekçi sistemleri ve saldırı tespit/engelleme sistemleri gibi mekanizmalar güvenlik gerekliliklerini yerine getirmek üzere kullanılan çözümlerinden en yaygın olarak kullanılanlarıdır. Bu mekanizmalar karşısındaki en büyük engel ise özellikle CPU ve bellek olmak üzere donanım kullanımıdır.

Bulut bilişim paradigması kullandığın kadar öde modeline sahiptir. Bundan dolayı kullanıcılar kendi maliyetleri için en ucuz olan çözümü tercih etmekte olup, kalifiye servis sağlayıcılar müşterilerine daha uygun çözümler sunmaktadır. Güvenlik bulut bilişimdeki en büyük ve dikkat çekici alanlardan biri olmakla birlikte, kullanıcılar için servis kalitesi değerlendirmesindeki en önemli kıstaslardan biridir.

Geleneksel bilgi sistemleri üzerinde gerçekleştirilen çok sayıda saldırı tipi bulunmaktadır. Bilgi sistemlerine zarar vererek, genel anlamda kullanılamamasını sağlayan siyah şapkalı bilgisayar korsanları, gerçekleştirilme kolaylıkları nedeniyle genellikle ağ tabanlı saldırıları kullanmaktadır.

Bu tez çalışmasında, bulut bilişimde en yaygın çözüm olarak kullanılan sanallaştırma teknolojisi üzerinde ağ tabanlı saldırı tespit sistemlerinin kullanımı farklı ağ mimarilerinin üzerinde ele alınmıştır. Sanallaştırma teknolojisinin, sağlayıcılar tarafından en çok tercih edilen çözüm olması ve aynı zamanda bulut teknolojileri yapılarının sanallaştırma teknolojileri üzerinde blade ve grid mimarilere nazaran daha kolay uygulanabilmesi nedeniyle, test ortamı olarak sanallaştırılmış bir yapı kullanılmıştır. Varsayılan ağ geçidi yaklaşımına ve konumlandırma seçeneklerine karşın daha az donanım gereksinimine sahip vekil bir ağ saldırı tespit sistemi yapısı üzerinde yoğunlaşılmıştır. Saldırı tespit görevinin korunan sanal makineler yerine farklı bir sanal makine tarafından yerine getirilmesi nedeniyle çalışmanın ismi **“Bulut Bilişimde Vekil Ağ Saldırı Tespit Sistemi”** olarak belirlenmiştir.

Sanal bir ortam içerisinde bulunan saldırı tespit sistemlerinin verimli bir konumlandırma seçeneği ortaya konmuş olup, bu çalışma sayesinde hem servis sağlayıcılar hem de bulut kullanıcıları savunma mekanizmalarını ne şekilde konumlandıracaklarına kolayca karar verebilecektir. Bunun yanında Vekil Ağ Saldırı Tespit sistemi yaklaşımıyla, servis sağlayıcılar mesh ağ yapıları içerisindeki ağ geçitlerinde de ağ tabanlı saldırı tespit sistemlerinin hangi konumlandırma seçeneğiyle kullanılması gerektiğini kolaylıkla belirleyebilecektir.

Sonuç olarak daha az kaynak kullanımıyla daha az maliyet isteyen sadece kullanıcılar değil, kendi altyapılarını daha az kaynak kullanımıyla korumak isteyen servis sağlayıcılar da bu yaklaşımı tercih edecektir.

PROXY NETWORK INTRUSION DETECTION SYSTEM IN CLOUD COMPUTING

SUMMARY

Cloud computing is the most common approach in information technologies nowadays. System administrators has to think about the expenditures while migrating from conventional data centers to the cloud computing. Security is one of the most important issues in cloud computing as in conventinonal information technologies. Antivirus software, firewalls, guard systems and intrusion detection and prevention systems are the most chosen solutions to achieve security requirements. The biggest obstacle in these mechanisms is hardware usage, and especially CPU and memory.

Cloud computing paradigm has the pay per usage model. Therefore, users choose the cheapest solution and qualified service providers are serving more elegant solutions to their customers. Security is one of the biggest and the most remarkable points in cloud computing, and it is also one of the most important criterias while evaluating service quality.

There are many types of attacks in the concept of conventional information technologies. And black hat hackers prefer the network based ones because of their convenience of realization.

In this thesis, the usage model of the network based intrusion detection systems in different network topologies with the virtualization which is the most common technology in the field of cloud computing is discussed. Virtulization is used for the test environment, because service providers are mostly choose virtualization and at the same time cloud computing infrastructres are commonly build on virtulization rather than grid and blade architectures, with the reason of realization of virtualized environmensts are easier than others relatively. Against to default gateway approach and localization opstions, a proxy network intrusion detection system is focused which needs less hardware. Because of the intrusion detection duty is appointed to a different virtual machine rather than user virtual machines, the study is called **“Proxy Network Intrusion Detection System for Cloud Computing”.**

A more efficient localization option of intrusion detection systems is presented in a virtualized environment, and with this study not only service providers but also cloud users can decide where to locate their defence mechanisms. Besides that Proxy Network Intrusion Detection System approach, service providers can determine the usage of network-based intrusion detection systems in default gateways of mesh network topologies easily.

As a result, not only the users but also the service providers will prefer this approach which has a less hardware usage and expenditure.

1. GİRİŞ

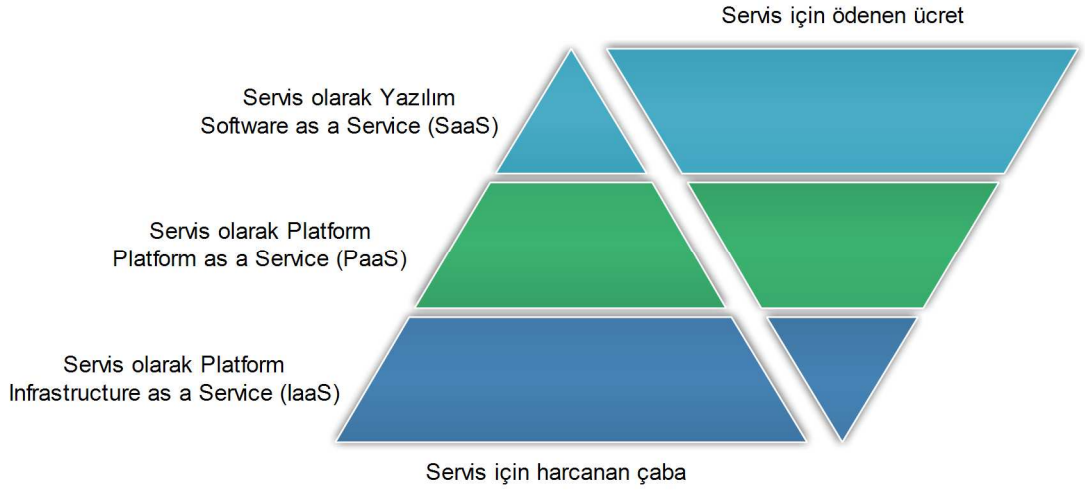
Bulut bilişim bilgi teknolojilerinde (Information Technologies - IT) sistemsel olarak kullanılmakta olan en yeni yaklaşımlardan biridir. Sistem yöneticileri geleneksel bilgi merkezlerinden bulut bilişime geçişte maliyetleri gözetmek zorundadır. Güvenlik, geleneksel IT'de olduğu gibi bulut bilişimin de en büyük sorunlarından biridir. Antivirüs yazılımları, güvenlik duvarları, bekçi sistemleri ve saldırı tespit/engelleme sistemleri gibi mekanizmaları kullanılarak güvenlik sorunlarının üstesinden gelinmeye çalışılmaktadır. Bu mekanizmalar karşısındaki performans açısından en büyük engel, özellikle Merkezi İşlemci Birimi (Central Processing Unit - CPU) ve bellek olmak üzere yoğun donanım kullanımıdır.

1.1 Tezin Amacı

Bulut bilişimde farklı servis ve dağıtım modelleri bulunmaktadır. Bu modellerde, gizliliğin sağlanması için güvenlik en temel gerensinimlerden biridir ve proaktif bir savunma mekanizması inşa etmenin temel yolu saldırı tespit/engellemedir.

Bulut bilişim genel anlamda kullandığın kadar öde modeline sahiptir. Herhangi bir bulut kullanıcısı, hizmet erişim oranıyla doğru orantılı olarak ödeme gerçekleştirmektedir. Bulut servisleri ve modellerin ücretleri arasındaki ölçek Şekil 1.1'de gösterilmiştir. Kullanıcılar Servis olarak Altyapı'yı (Infrastructure as a Service – IaaS) kullanmayı seçtiklerinde, sağlayıcılar donanım ve bunlara erişim arayüzlerini servis olarak sunmaktadır. Bu nedenle, sorumluluğun büyük kısmı kullanıcılarda bulunmaktadır. Sağlayıcılar, müşterileri daha az bir gayret gereksinimi ile desteklerken, servis maliyetleri düşmektedir. Ücretlerdeki bu düşüşün birçok nedeni vardır. Atlyapı için, kullanıcılar işletim sistemi (operating system – OS), yazılım veya herhangi bir ağ servisi için lisanslama ücreti ödememektedir. IaaS'ta, ayrılmış donanım (ağ, depolama, CPU, sunucu vb.) ve sanallaştırılmış ortam sağlayıcılar tarafından desteklenmektedir. Servis olarak Platform'da (Platform as a Service – PaaS), IaaS'a ilave olarak ara katman yazılımı ve yürütme ortamı sağlayıcılar tarafından işletilmekte, Servis olarak Yazılım'da (Software as a Service –

SaaS) ise, PaaS'a ilave olarak veri ve ağ servisleri sağlayıcılar tarafından idame ettirilmektedir. Maliyetler, sağlayıcı tarafından yönetilen kısımlar arttıkça yükselmektedir.



Şekil 1.1 : Bulut servislerine harcanan çaba ve ödenen ücret karşılaştırması.

Sanallaştırma teknolojileri, CPU ve bellek gibi bilişim kaynaklarını soyutlamaktadır. Bu kaynaklar, Saldırı Tespit Sistemleri (Intrusion Detection System – IDS) tarafından saldırılar incelenirken kullanılmaktadır. Eğer ortamda ağ saldırı tespit sistemi kullanılıyorsa, bunlara ilave olarak ağ bileşenlerine de ihtiyaç duyulmaktadır. Kullanıcılar, her bir sanal makine (virtual machine – VM) için ayrı ayrı IDS'ler kullanırlarsa; CPU, bellek ve ağ trafiği kullanımı tek bir IDS kullanımına göre artmaktadır.

IDS'ler bir topoloji içerisinde temel olarak üç farklı konumda teşkil edilebilir. Basitçe, her bir sunucu üzerine birer IDS yapılandırılabilir. Bu yaklaşımın yönetilebilirliği, tek bir IDS'yi yönetmeye göre daha zordur. Tek bir IDS yaklışımı tercih edilirse, IDS'yi konumlandırmak için iki farklı senaryo ortaya çıkmaktadır. Bunlardan biri ağ içerisinde farklı bir sunucu üzerinde, diğeri de varsayılan ağ geçidi üzerinde IDS konumlandırmaktır. İlk senaryoda, ağın tüm trafiği IDS'nin bulunduğu sunucu üzerine yansıtılacak ve yüksek ağ trafiğinden dolayı birçok darboğaz meydana gelecektir. Ağ geçidi senaryosunda, CPU ve bellek kullanımı, ayrı bir sunucu üzerinde konumlandırılacak IDS ile yakın olacak ancak ağ trafiği yükü meydana gelmeyecektir. Ağdaki tüm trafik ağ geçidi üzerinden geçecek, bu nedenle, IDS ağ trafiğine ilave bir darboğaz oluşturmayacaktır.

Bu çalışmada, IDS'yi konumlandırırken asgari donanım kullanımını gerçekleştiren senaryo tespit edilerek, diğer IDS konumlandırma senaryolarıyla karşılaştırılması ve uygun konumlandırma yaklaşımının belirlenmesi amaçlanmıştır. Böylelikle, müşterilerin ödediği miktarda masraf azalacak ve kullanım olarak IaaS modeli seçilirse sağlayıcıların desteği asgari seviyede olacaktır. Sonuç olarak, müşterilerin ödediği miktarın yanında sağlayıcıların yönetim maliyeti de azalacaktır. Bunun yanında, kullanıcılar tüm VM'ler ve IDS'ler üzerinde daha fazla yönetimsel ve yapılandırma yükü üstlenecektir.

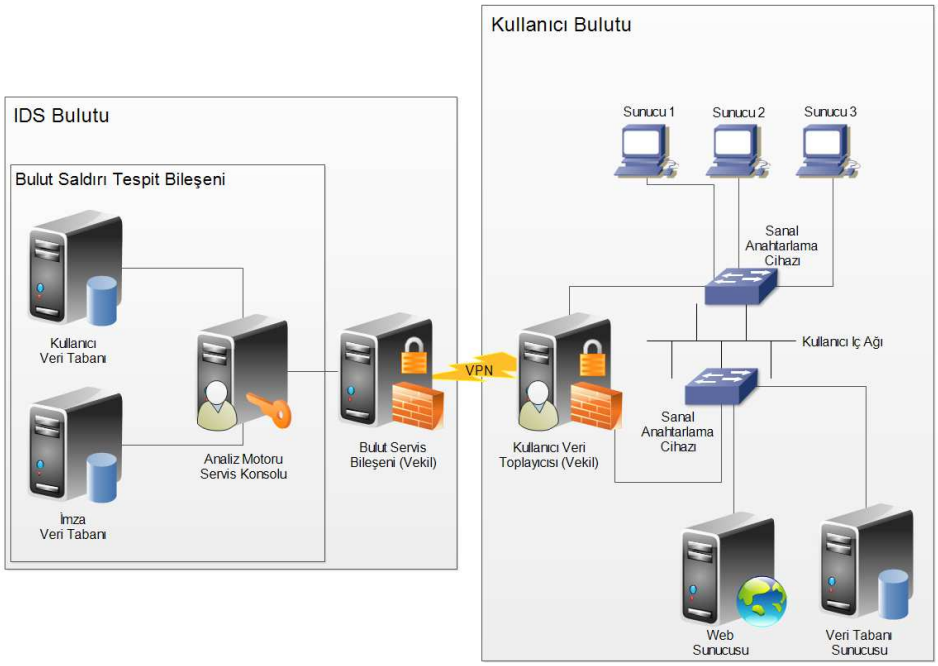
2. LİTERATÜR ARAŞTIRMASI

Collabra adı verilen dağıtık IDS yaklaşımında sanallaştırma sayesinde her bir VM, donanıma sanal makine denetleyicisi (Virtual Machine Monitor – VMM) üzerinden bir hiper çağrı (hypercall) ile erişmekte ve Collabra bir VM'e ait olan her bir çarıyı filtrelemektedir [1]. VMM'leri atlatmak amacıyla yeni nesil bazı saldırı çeşitleri, kendilerini geleneksel IDS'lerden gizlemek için hypercall'lar içerisinde gizlenmektedir. Collabra bu tarz akıllı saldırıları engellemeyi hedeflemektedir. Eğer çağrılar kararlı değil ve bütünlükleri zararlı olarak işaretlenmiş ise Collabra bunu saldırı girişimi olarak tanımlamakta ve sanal ortamı farklı yollarla güvenlik altına almaktadır.

Bulut tabanlı olarak geliştirilen ve ev sahibi buluta bir sanal özel ağ (Virtual Private Network – VPN) aracılığıyla bağlanan IDS'de, ev sahibi bulut ve IDS bulutu, bulut altyapısında iç içe geçen bir mimaridedir [2]. VPN bağlantısının ev sahibi bulut ve IDS bulutu taraflarının ikisinde de verileri IDS ve ev sahibi bulut arasında ileten birer vekil sunucu bulunmaktadır. Şekil 2.1'de bulut tabanlı IDS'ye ait mimari yapı sunulmuştur. "Kullanıcı Veri Toplayıcısı", ev sahibi bulut vekil sunucusu arkasında kullanıcı bulutuna ait veri toplamakta, IDS tarafındaki vekil sunucu ise bu verileri almakta ve analiz yapmak üzere IDS motoruna göndermektedir. Daha sonra analiz motoru toplanan verilere ait durumları kullanıcı bilgileri ve imza veritabanıyla karşılaştırarak denetlemektedir.

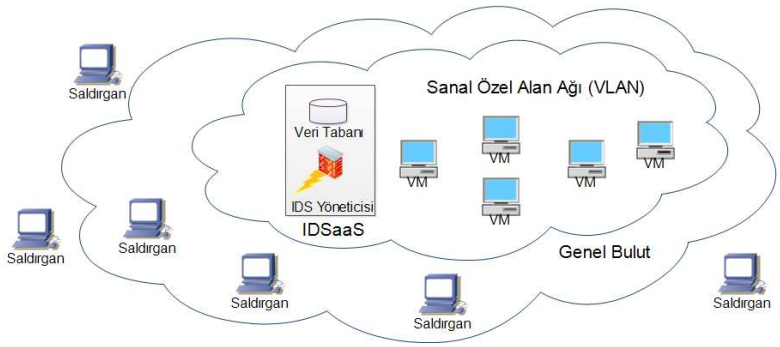
Sistem yöneticileri tarafından geleneksel IT altyapılarında en fazla tercih edilen üç Ağ Tabanlı IDS (Network Based Intrusion Detection System – NIDS)'in karşılaştırıldığı çalışmada; Snort, TCPDump ve Network Flight Recorder'ın iyi ve kötü yanlarıyla, analiz ve tespit mekanizmalarının arasındaki farklılıklar ortaya konmuştur [3]. Snort'un diğer NIDS'lere göre artılarıyla daha iyi olduğu sonucuna varılmış olup, ticari bir ürün olmamakla birlikte Genel Kamu Lisansı (General Public License – GPL)'na sahip olması avantajları arasında gösterilmiştir. Böylelikle bir Snort kullanıcısı bir altyapıda IDS kullanmak için yüksek meblağlar ödemek zorunda

kalmamaktadır. Ayrıca Snort'un sistem yönetiminin piyasadaki diğer GPL sahibi veya ticari NIDS'lere göre daha kolay ve kullanıcı dostu olduğu vurgulanmıştır.



Şekil 2.1 : Bulut altyapısına uygun olarak geliştirilen IDS mimarisi.

Genel Bulutlarda Servis olarak Saldırı Tespit Sistemi (IDSaaS: Intrusion Detection System as a Service in Public Clouds) isimli çalışmada, IaaS'da genel bulut içerisinde soyutlanan bir ortamda çalışan henüz gerçeklenmemiş bulut tabanlı bir IDS tanıtılmıştır [4]. Bu çaışmada, kullanıcı IDS üzerindeki tüm kontrolü elinde bulundurmakta ve sağlayıcının saldırı tespit mekanizması üzerinde herhangi bir sorumluluğu bulunmamaktadır. Farklı coğrafi konumlarda bulunan bulutlar ile bu bulutlar bünyesinde teşkil edilmiş alt bulutlar, birbirleriyle VPN'ler aracılığıyla iletişim kurmaktadır. Şekil 2.2'de iç içe geçmiş bir yapıda bulunan bulut mimarisi gösterilmiştir. IDSaaS taşınabilirlik, elastikiyet, ölçeklenebilirlik ve istenildiğinde erişim gibi birçok bulut özelliğini desteklemektedir.



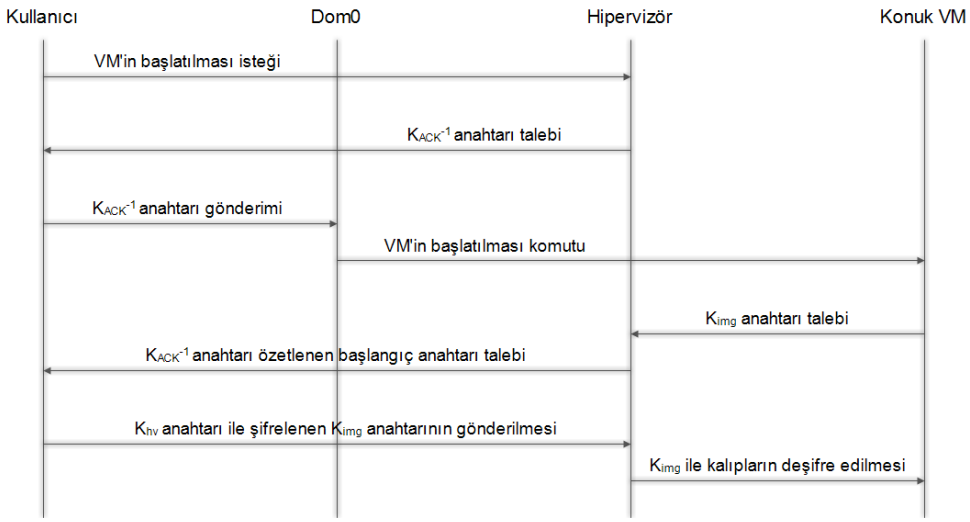
Şekil 2.2 : İç içe inşa edilmiş bulut altyapısında IDSaaS.

Sanallaştırma üzerinde çalışan ve servis sağlayıcısının bulut ortamını izleyerek güvence altına almasına yardımcı olan bir izleme tekniğinin önerildiği modelde VM'lerin durumlarını izleyerek güncelleme amacıyla Java Kimlik Doğrulama Yetkilendirme Servisi (Java Authentication Authorization Service – JAAS) kullanılmıştır [5]. Bir saldırı tespit edildiğinde, VM'ler kolaylıkla göç edebilmekte, taşınabilmekte ve sanal ortamdan izole edilebilmektedir. Bu model çok kolay ve temel bir yaklaşım kullanmakta olup, proaktif bir güvenlik sağlayamamaktadır. Güncel saldırıları tespit ederek bunlara karşı koyabilmek amacıyla önerilen model istenildiğinde reaksiyon gösterebilecek şekilde geliştirilmelidir.

Özel Sanal Altyapı (Private Virtual Infrastructure – PVI) yaklaşımında, tehditler karşısında durumsal farkındalığı artırmak amacıyla sorumluluk servis sağlayıcı ve kullanıcı arasında paylaştırılmaktadır [6]. PVI'da veri merkezi yeni bir kavrama sahip olarak öne çıkmaktadır. Bir veri merkezi fiziksel olarak müşteriye ait bir yerde bulunmakta, ancak servis sağlayıcı tarafından desteklenmektedir. Bunun yanında müşteriler güvenlik ayarları ve yapılandırmalarına erişebilmektedir. PVI durumsal farkındalığı doğrulamak amacıyla Güvenilir Platform Modülü (Trusted Platform Module – TPM) tekniklerini kullanmaktadır. Sanal TPM (Virtual TPM – VTPM)'ler bulut içerisindeki VM'lere yerleştirilmiştir. Bu VTPM'ler Güvenilir Bilişim (Trusted Computing – TC) ihtiyaçlarını karşılamaktadır. Fiziksel katmanda sağlayıcı tarafından, mantıksal katmanda ise kullanıcı tarafından kontrol edilmekte ve yönetilmekte, böylece güvenlik sorumluğu paylaşılmış olmaktadır. Servis sağlayıcının altyapıyı her anlamda belirgin bir şekilde gerçeklemesi gerekmekte olup, böylelikle müşterinin yapılandırmayı izleyebilmesi, kontrol edebilmesi ve müdahale edebilmesi amaçlanmaktadır. PVI yatırım maliyetlerini azaltırken güvenlik sorunlarının üstesinden sorumluluk paylaşımı yoluyla gelmektedir. Müşteriler altyapıdaki sorunları izleyebilmek ve kontrol edebilmek için gerekli bilgi ve donanıma sahip olmak zorundadır.

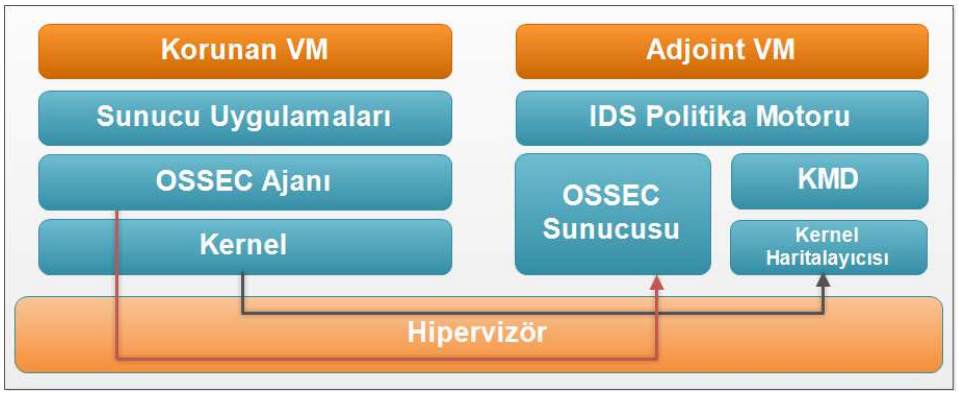
VM İç Gözlemi için Şekilsel bir Model (Virtual Machine Introspection – VMI)'de, VM'lerin hipervizör içerisindeki durumlarını gözlemleyen ve analiz eden bir yöntem tanımlanmıştır [7]. Tanımlanan bu modelde, VM'lerin durum bilgisi, ilişkisel durum bilgisi, durumların türlerine göre sınıflandırması ve ayrıca VMI tabanlı IDS'in geliştirilmesindeki zorluklar açıklanarak, VMI tabanlı bir güvenlik uygulamasının nasıl gerçekleştirilmesi gerektiği ortaya konmuştur.

Güvenilmeyen bir sunucu üzerinde bulunan VM'lerin gizliliğini sağlamak amacıyla yapılan çalışmada, konuk VM'lerin boot edilmesi esnasında kullanılan güvenli bir protokol geliştirilmiştir [8]. Bu protokol, kullanıcıların çekirdek (kernel) ve başlangıç bellek sürücülerini (initial ram disk) bir K_{img} anahtarıyla şifreleyerek saklamalarına imkân tanımaktadır. Daha sonra hipervizörün kalıpları deşifrelemesini tetikleyen bir kernel kalıbıyla birlikte hypercall kodunu paketlemektedir. Farklı kullanıcıların kullanımı için, hipervizör farklı asimetrik anahtar çiftleri üretmektedir. İlk olarak, kullanıcı sunucuya bir istek göndererek bilgilendirmekte ve hipervizör TPM sorgusunu gerçekleştirerek K_{AIK}^{-1} özel anahtarını talep etmektedir. Eğer istek başarılı olursa, sonrasında kullanıcı yetkili işletim sistemine (Domain 0 – Dom0) başka bir istek göndermekte ve Dom0 bu isteğin VM kimlik bilgisiyle (identification - ID) eşleşip eşleşmediğini kontrol etmektedir. Konuk VM anahtarı çalıştırmakta ve hipervizöre kernel kalıbı ve bellek sürücüsünün deşifrelenmesi için bir istek göndermektedir. Hipervizör, başlangıç (boot) isteği ve anahtar istek mesajını K_{hv}^{-1} anahtarıyla şifreleyerek paketlemekte ve gönderme olup, kullanıcı bu isteği aldıktan sonra K_{img} anahtarını K_{hv} ile birlikte geri göndermektedir. K_{img} anahtarı hipervizöre ulaştıktan sonra kernel ve başlangıç bellek sürücüsünün deşifrelenmekte ve kullanıcıya VM'in boot işleminin başlatıldığını bildirmektedir. Bu protokolde, kullanıcı ile hipervizör arasındaki haberleşmenin güvenli bir şekilde gerçekleştirilmesi amaçlanmıştır. K_{img} VM'lerin kalıplarını korumakta ve Dom0'ın kalıplara erişerek istismar etmesini engellemektedir. Şekil 2.3'te protokolün işleyişi gösterilmiştir.



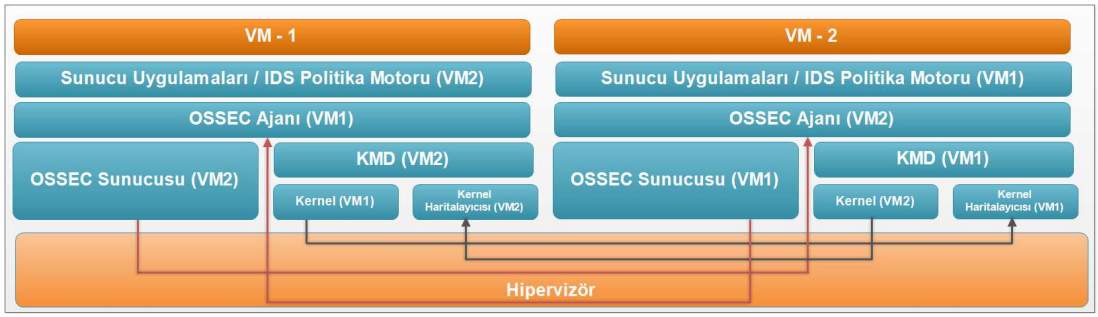
Şekil 2.3 : Güvenli VM başlangıç protokolü.

Yardımcı VM (AdjointVM) isimli çalışmada gizliliği sağlanmak istenen her VM için bir VM daha yapılandırılarak, ilave olarak yapılandırılan bu VM'in asıl gizliliği sağlanmak istenen VM'i güvence altına alması sağlanmıştır [9]. Gizliliğinin sağlanması istenen VM'in boot edilmesi sırasında güvenli VM başlangıç protokolü kullanılmıştır [8]. VM'in güvenliğinin sağlanması amacıyla melez bir IDS yapısı oluşturulmuş olup, sunucu tabanlı bir IDS olan Opearting System Security (OSSEC) ile işletim sistemi çekirdeğini izleyen Kernel Monitor Daemon (KMD) kullanılmıştır. OSSEC'in sunucu istemci mimarisinden dolayı OSSEC sunucusu eşlenik olarak ortama eklenen VM üzerinde, istemci durumundaki OSSEC ajanları da asıl güvenliğinin sağlanması amaçlanan VM üzerinde teşkil edilmiştir. Saldırılara ait imzalar OSSEC sunucusu üzerinde bulunmakta ve ajanlar saldırı bilgilerini sunucu bünyesindeki saldırı imzaları veri tabanından almaktadır. KMD ise güvenliğinin sağlanması amaçlanan VM'in çekirdeğini haritalayarak, işletim sisteminin arka planında kendini bir sistem servisi gibi gösteren gizli ve kötü niyetli bir yazılım olup olmadığını kontrol etmektedir. KMD üzerinde rutin ve olması gereken sistem servislerine ilişkin bilgiler bulunmakta, servis kimlik numaralarına ait özet değerler kaydedilmekte ve herhangi bir anormallik, işletim sistemi çekirdeğinin belirli zamanlarda denetlenerek mevcut bilgilerle karşılaştırılması sonucunda tespit edilmektedir. Güvenli VM başlangıç protokolünün kullanılmasıyla, güvenilmeyen bir ortamdaki servis sağlayıcı veya sistem yöneticisi tarafından kullanıcı VM'lerine ait sabit ya da sanal sürücülere istem dışı erişimin önüne geçilmiştir. Geliştirilen melez IDS yapısı şekil 2.4'te gösterilmiştir.



Şekil 2.4 : AdjointVM melez IDS modeline genel bakış.

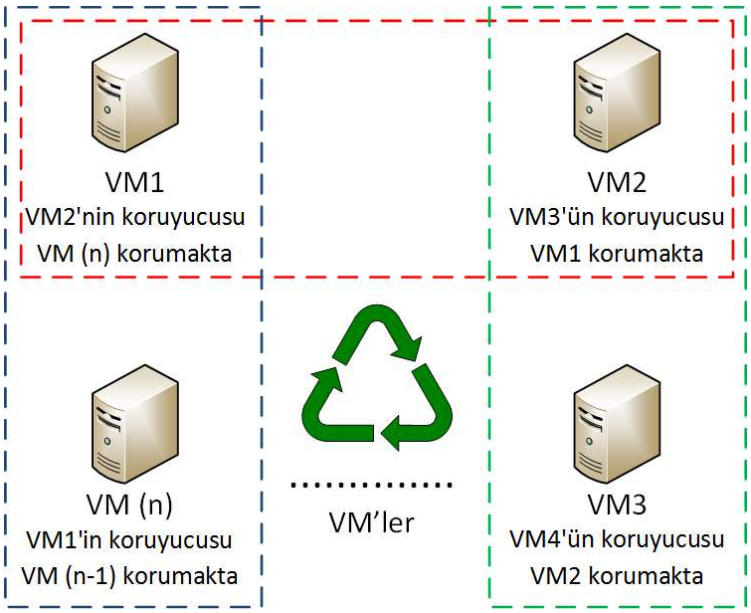
AdjointVM'de Dairesel VM Zinciri'nde, AdjointVM güvenlik yaklaşımında bulunan bazı eksikliklerin giderildiği bir model sunulmuş ancak gerçeklenmemiştir [10]. Bu modelde; AdjointVM'nin yanlış pozitif oranını düşürmek, AdjointVM'nin istismar edilme direncini güçlendirmek, AdjointVM'yi iç ve dış saldırılara karşı, özellikle de kötü niyetli sistem yöneticilerine karşı güvenli kılmak, esnek bir güvenlik politikası oluşturmak, sadece gizlilik ihtiva eden bilginin bulunduğu VM'yi değil, ayrıca bu VM'in güvenliğini sağlayan diğer VM'in de güvenliğini sağlamak ve gizlilik ihtiva eden bilgiyi koruma amacıyla çalışan VM sayısını düşürerek, sistemin toplam maliyetini de azaltmak amaçlanmıştır. AdjointVM'deki koruyucu VM'nin ana görevi korunan VM'nin güvenliğini sağlamak iken koruyucu VM saldırılara karşı hala zayıf durumdadır. AdjointVM üzerinde önerilen geliştirmeler yapıldığı takdirde, çift içerisindeki her VM korunacak ve birbirleriyle eş olacaktır. Her biri diğerinin belleğini haritalayabilecek ve birbirlerinin OSSEC sunucu görevlerini yerine getirebilecektir. Çiftler içerisindeki herhangi bir VM'ye saldırı gerçekleştirildiği takdirde, bu durum çiftler tarafından kaçırılmayacaktır. Önerilen bu model sadece internet üzerinden bulut hizmeti sunan ve güvenilmeyen bulut sağlayıcılarına karşı değil, ayrıca kendi bulut hizmetlerine sahip kurumsal işletmelerde görevli kötü niyetli bilgi teknolojileri çalışanlarına karşı da etkili olacaktır. Önerilen geliştirilmiş güvenlik modeli Şekil 2.5'te gösterilmiştir.



Şekil 2.5 : Geliştirilmiş AdjointVM melez güvenlik modeli.

Önerilen model üzerinden devam edildiğinde, koruyucu VM ve korunan VM yapısına ilave olarak bir VM diğer bir VM'yi korurken, üçüncü bir VM'nin de ilk VM'yi koruduğu bir zincir yapısı teklif edilmiştir. Böylece *n* adet VM'nin korunması gereken bir sistemde, ilave herhangi bir VM'ye ihtiyaç duyulmaması planlanmıştır. Dinamik bir yaklaşımla güvenlik zincirine bir VM eklemek ve çıkarmak ölçeklenebilir bir şekilde mümkündür. Yeni bir VM (VM_n) bulut sistemine eklenmek istediğinde, ilk olarak sistemdeki bir VM (VM_a)'nin sorumluluğunu almak zorundadır. Daha sonra

VM_a'nın koruyucusu olan VM_b, kontrol ettiği VM'yi yeni VM_n olarak değiştirecektir. Eğer zincirden (bulut sisteminden) bir VM çıkarılmak istenirse, koruyucu VM (VM_a)'nin koruduğu VM (VM_b), VM_n sistem dışarısına çıkarıldıktan sonra değiştirilmelidir. AdjointVM zincir güvenlik modeli Şekil 2.6'da gösterilmiştir.



Şekil 2.6 : AdjointVM zincir güvenlik modeli.

Örnek olarak, AdjointVM yaklaşımında bir kullanıcının bulut içerisinde dört adet VM çalıştıracağı varsayılırsa, bu VM'lerin güvenliğini sağlamak amacıyla dört adet AdjointVM çalıştırması gerekecektir. Bunun yerine geliştirilmiş olan zincir modeli kullanılırsa, zincir içerisindeki herbir VM diğer bir VM'nin güvenliğini sağlayacak ve böylelikle ilave bir VM'ye ihtiyaç olmadan, dört VM ile yine bu dört VM'nin güvenliği sağlanmış olacaktır. AdjointVM ile geliştirilen modelin karşılaştırması Çizelge 2.1'de sunulmuştur.

Açık kaynaklı bir bulut ortamında bilinen IDS'lerin farklı dağıtım modelleriyle kullanılması sonucu daha ucuz ve hızlı bir çözüm bulmak üzere yapılmış olan diğer bir çalışmada, dağıtım seçimine bağlı olarak farklı yarar ve eksiklikler ortaya konmuştur [11]. Kümeleme Denetçisi'ne yakın bir konumda tesis edilen tek bir IDS üzerinden tüm trafik akacak ve gözlemlenebilecektir. Bu yaklaşımda, tek bir IDS üzerine çok fazla yük binecek, buna bağlı olarak ta koordineli olarak gerçekleştirilen gelişmiş saldırılarda, oluşturulabilecek yoğun trafik sonucunda IDS devre dışı kalabilecektir. Diğer bir taraftan ele alınacak olunursa, IDS'nin fiziksel makineler üzerine dağıtımlarının yapılması ve her IDS'nin trafiğin bir kısmını kontrol etmesi

sağlanırsa, yoğun bir trafiğin oluşması durumunda bunun üstesinden gelinebilecek ve IDS devre dışı kalmayacaktır.

Çizelge 2.1 : AdjointVM ile geliştirilen güvenlik modelinin karşılaştırılması.

Özellik	AdjointVM	Geliştirilen Model
Bilginin güvenliğinin sağlanması	Evet	Evet
Düşük yanlış pozitif oranı	Evet	Evet
İstismarlara karşı dayanıklılık	Evet	Evet
Saldırganlar tarafından tespit edilebilirlik	Hayır	Hayır
Kötü niyetli sistem yöneticisi ve servis sağlayıcılara karşı koruma	Hayır, Ancak eksik kısımlar bulunmakta	Evet, Eksik kısımların üstesinden geliyor
Yapılandırılabilir güvenlik politikası	Evet	Evet
Ölçeklenebilirlik	Kolay	AdjointVM'e göre daha zor
Sadece korunan VM'yi değil, koruyan VM'nin de güvenliği	Sadece korunan VM	Hem korunan, hem de koruyucu VM
n VM'yi koruyan VM sayısı	2n	n
Maliyet	Düşük	Daha düşük

3. BULUT BİLİŞİM

Yaygın bilişimin, teknolojik imkânlar aracılığıyla bilgiye erişilebilirliğinin, sürekli ve görünmez bir biçimde gerçekleşmesi gerektiği, kavramsal olarak ilk defa Xerox Palo Alto Research Center (PARC) Incorporated araştırmacısı Mark Weiser tarafından, Philip K. Dick'in 1969 yılında yayınlanan Ubik isimli romanından esinlenmesiyle ortaya çıkmıştır [12, 13, 14]. Ubik romanında, bütün nesnel varlıklar zeki birer varlık olarak birbirleriyle iletişim halindedir. Bu iletişim çevresel tüm etkenlerle ilişkili olarak gerçekleşmektedir. İletişim ağları, ortamdan bağımsız olarak sürekli bilgi erişimine imkân vermektedir.

Yaygın Bilişim yaklaşımının gündelik hayata dâhil olmasıyla birlikte, gerçek zamanlı ve mekân bağımsız olarak interaktif bir iletişim ortamı kullanılmaya başlanmıştır [14]. Mevcut iletişim altyapısına sadece kişisel bilgisayarlar değil, sunucu bilgisayarlar, tablet bilgisayarlar, cep telefonları, akıllı telefonlar, televizyonlar, arabalar ve gündelik yaşantıda yeri olan bütün elektronik cihazlar dâhil olmuştur. Mobil cihazların ve mobil iletişim teknolojileri kullanımının yaygınlaşması, eş zamanlı ve mekân bağımsız interaktif yaşamın hayat dinamikleri içerisinde yer edinmesini sağlamıştır. IT, önceleri sadece sabit bir kaynaktan erişilen merkezi bir yapıya sahip iken, yaygın bilişim ile birlikte merkezileşen yapının dışında, zaman ve mekân bağımsız interaktif etkileşim yetenekleri ortaya çıkmıştır. [15]

Bulut bilişim, eş zamanlı olarak birden fazla mekânda teşkil edilmiş, kullanımı kolay, istenildiği anda erişilmesi mümkün, ayarlanabilen bilişim kaynaklarından oluşan ve söz konusu kaynakların yönetimsel sürdürülebilirliği amacıyla en düşük seviyede müdahaleye ihtiyaç duyan karşılıklı etkileşime dayalı bir iletişim modelidir [16]. Günümüzde bulut bilişim, kullanıcılarının ihtiyaç duyduğu hizmetleri almak üzere servis sağlayıcılara ait bilişim kaynaklarını kullandıkları, bu kaynaklardan faydalandıkları kadar ücret ödedikleri, sürekli ve hızlı bir gelişim sürecine sahip bir teknolojidir [17]. Bulut bilişim; uygulamalar ve kullanıcılar için iletişim ağı üzerinde merkezileştirilmiş veri merkezlerinde kümelenmiş hesaplama gücü, depolama alanı ve uygulamaları, ağ uygulaması geliştiricileri için ağ üzerinde ölçeklendirilmiş bir

yazılım geliştirme ve yürütme ortamını, altyapı sağlayıcıları ve sistem yöneticileri için büyük ve dağıtık veri merkezlerinin ağ üzerinden birbirlerine bağlı bulunmalarıdır [18]. Farklı kaynaklar tarafından birbirine yakın tanımlamalara sahip bulut bilişim özet olarak, alınan hizmetin zaman ve mekân bağımsız olarak, kullanıcının ihtiyaçlarına göre şekillendirilebilen, yönetimi asgari seviyede müdahale gerektiren ve kullanılan servis miktarına göre ücretlendirilen günümüz bilişim teknolojisi olarak tanımlanabilir.

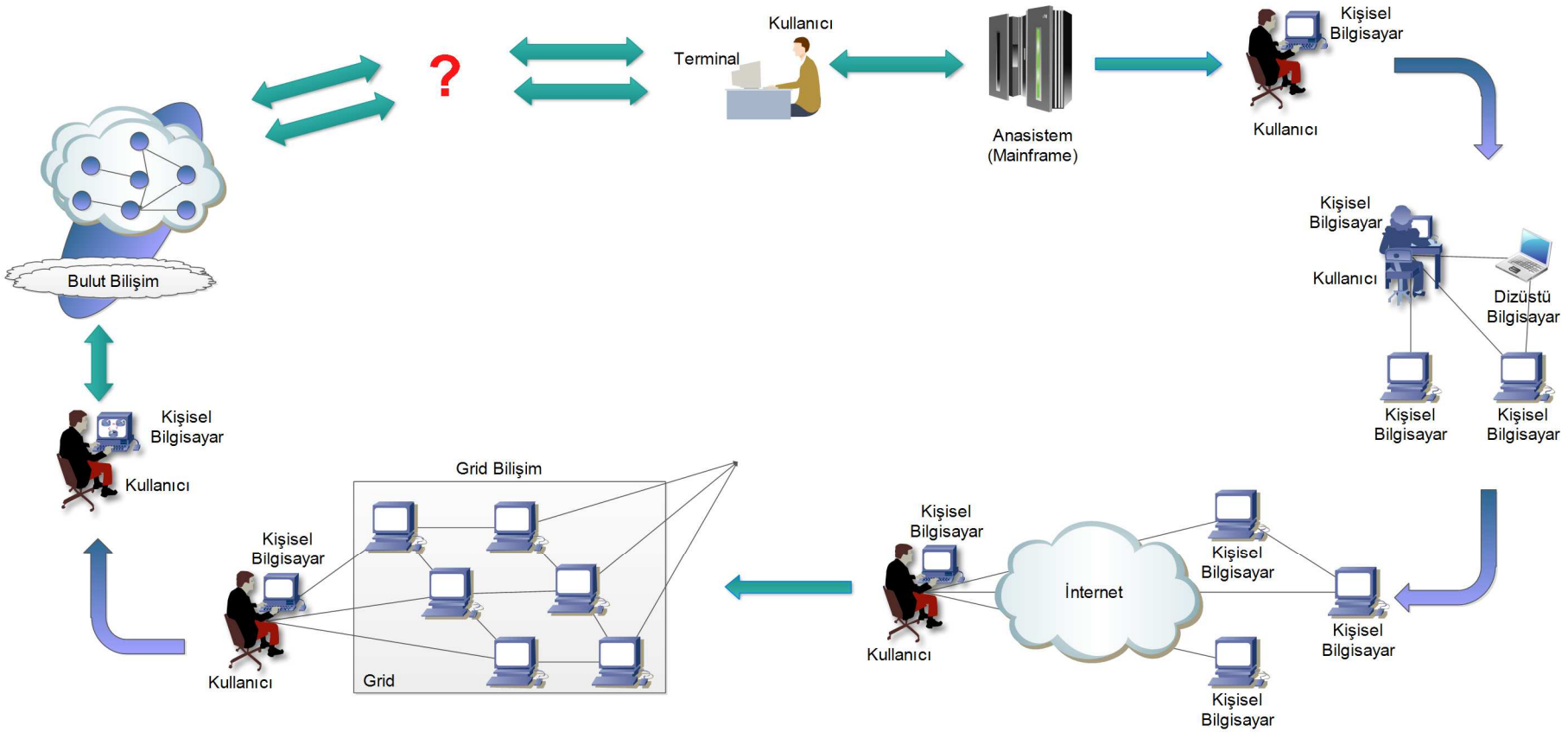
IT'nin son elli yıl içerisindeki gelişimi ve kullanımı incelendiğinde, altı aşamalı bir sınıflandırma yapılabilir [19]. Birinci aşamada, kullanıcılar ana çerçeve (mainframe) olarak adlandırılan ana sistemlere bağlanmak amacıyla diğer kullanıcılarla ortak olarak kullandıkları ekran ve klavyelerden oluşan terminaller kullanılmaktaydı. İkinci aşamada, bağımsız bilgisayarlar kullanıcıların günlük yapmaları gereken işleri karşılayabilecek güce sahip bulunuyorlardı ve ana sistemlerin kullanılmasına gerek kalmamıştı. Üçüncü aşamada, birbirine bağlı bilgisayarlardan oluşan bilgisayar ağları ile kullanıcılar birbirleri arasında iletişim yeteneğine kavuşmuş, aynı yerel alan ağı (Local Area Network – LAN) içerisinde paylaşımda bulunan kaynaklara erişebilmekteydiler. Dördüncü aşamada, LAN'lardaki teknolojik gelişim bu ağların dünya çapında bir altyapı aracılığıyla iletişim kurabilmelerini sağlamış ve internet ortaya çıkmıştır. Böylelikle kullanıcılar, fiziksel olarak uzakta bulunan uygulama ve kaynaklara erişim ve bu kaynakları kullanma imkânına sahip olmuşlardır. Beşinci aşamada, paylaşılan hesaplama gücü ve dağıtık bir şekilde teşkil edilmiş olan depolama alanlarının oluşturduğu grid mimari ortaya çıkmış, kullanıcılar, kişisel bilgisayarlar aracılığıyla grid mimari içerisinde bulunan bileşik kaynaklara erişim imkânına kavuşmuşlardır. Altıncı ve şu anda içerisinde bulunulan aşama **Bulut Bilişim**'dir. Bulut bilişim, uygun tüm kaynaklara internet üzerinden ölçeklenebilir ve kolay bir şekilde erişilebilmesini sağlamaktadır. İnternetin son derece güçlü bir bulut olarak ele alınması durumunda, kişisel bilgisayarlar buluta erişim için kullanılan önemsiz birer terminal halini almaktadır. Bu açıdan ele alındığında bulut bilişim, özgün ana sistem yaklaşımına geri dönüş gibi gözükebilmektedir. Ancak bulut bilişim yaklaşımı bu kadar basite indirgenmemelidir. Bulut bilişim, ana sistemlerden farklı olarak tek bir fiziksel cihazın sonlu bilişim gücü yerine, internet üzerinde bulunan sonsuz güç ve kapasiteye sahip bir yaklaşımdır. Kişisel bilgisayarlar, ana sisteme bağlanan bir ara yüze sahip basit birer terminal olmalarının yanında, bulut bilişim

yaklaşımı içerisinde dikkate değer gücün birer parçası olarak yerel bilişim ve ön belleğe alma işlemlerini gerçekleştirmektedir. Şekil 3.1'de IT teknolojilerindeki evrimsel süreç gösterilmiştir [19].

İhtiyaç duyulan servis ve kaynaklara internet üzerinde bir süredir erişilebilmekte olup, günümüzde bulut bilişim üç önemli değişikliğe odaklanmaktadır [20]. Ölçeklenebilirlik; bulut bilişim sistemleri birkaç veya çok daha fazla veri merkezi üzerindeki altyapıyı kullanabilmektedir. Basitlik; ilk bulut bilişim servisleri, yüksek performanslı ve dağıtık olarak teşkil edilmiş bilişim yaklaşımına sahip ve son derece karmaşık bir yapıya sahip olarak, genellikle grid tabanlı servislerde kod geliştirmek, düğümler arasında mesaj aktarımı ve diğer özel yöntemleri kapsamaktaydı. Ücretlendirme; bulut bilişim genellikle hizmet alınan servislerin ihtiyaç duyduğu kaynakların kullanım miktarlarına göre fiyatlandırılmaktadır. Örneğin, kullanıcının bir saat içerisinde ilave olarak 1,000 adet işleme ihtiyacı varsa, bu 1,000 adet işlemi bir saat içerisinde kullanma durumuna göre ücretlendirilmektedir. Herhangi bir yatırım harcaması yapılmasına ihtiyaç duyulmamaktadır.

3.1 Neden Bulut Bilişim?

Büyük işletmelerin geleneksel IT kullanım modellerinde yüklü miktarda yazılım ve donanım yatırımları yapmaları ve yaşayan bir süreç içerisinde bu yatırımların tecrübeli ve profesyonel çalışanlar tarafından bakım ve idamelerini sürdürmeleri gerekmektedir [18]. Bulut bilişim sistemleri içerisinde IT servislerine ait kaynak ve hizmetler kullanıldığı kadar maliyetlere sahip oldukları için, geleneksel IT kullanım modellerine göre kayda değer bir maliyet avantajına sahiptir. Örneğin; Microsoft tarafından sunulan Exchange Online hizmeti, zaman ve mekân bağımsız olarak çoklu cihazlar üzerinden elektronik posta (e-posta) erişimi, takvim ve kişiler kullanımına ayda $10, SharePoint Online hizmeti ise aylık $7.25 ücret karşılığında kullanılabilmektedir. Bu ücretler geleneksel IT kullanım yaklaşımlarında personel ücretleri ve diğer giderler sonucunda bulut bilişim sistemlerine oranla çok daha yüksek maliyetlerin oluşmasına neden olabilmektedir. Çizelge 3.1'de IBM tarafından gerçekleştirilen geleneksel IT ve bulut bilişim tabanlı yönetimsel görevlere ait zaman karşılaştırması gösterilmiştir [18].



Şekil 3.1 : Bilişim dünyasının son elli yıl içerisindeki evrimi.

Özellikle kurumsal kullanıcıların maliyet etkin bir yönetime sahip olmayı istemelerinden dolayı, günümüzde geleneksel IT harcamalarını (sadece finansal değil, zaman ve personel gücü gibi diğer etkenler de dâhil olmak üzere) düşürmek amacıyla kullanıcılar bulut bilişim teknolojilerini kullanmaya ve geleneksel IT altyapılarını bulut bilişim altyapısına geçirmeye başlamışlardır.

Çizelge 3.1 : IBM tarafından sekiz temel yönetimsel IT görevi için geleneksel ve bulut bilişim tabanlı dağıtım sürelerinin karşılaştırması.

Yönetimsel IT Görevleri	Geleneksel Dağıtım	Otomatik Dağıtım
Sunucu Tahsis Etmek	3 gün	< 1 saat
Yazılım Yüklemek	5-10 gün	< 1 saat
Ağ ve güvenlik parametrelerini ayarlamak	5-10 gün	< 1 saat
İşletim sistemini yedeklemek	2 saat	½ saat
İşletim sistemini geri almak	2 saat	1 saat
İşletim sistemi yaması yüklemek	2 saat	1 saat
Bilişim kaynaklarını dinamik olarak tahsis etmek	1 saat	2 dakika
Farklı servisler için işletim sistemi parametrelerini düzenlemek	10 dakika	1 dakika
Toplam	**14-24 gün**	**< 6 saat**

3.2 Bulut Bilişimin Özellikleri

Bulut bilişimin tanımından yola çıkarak, günümüzde kullanımı giderek yaygınlaşan bu yaklaşımın beş temel özellilliği sağlayıcı ve kullanıcılara temin etmesi gerekmektedir [16].

3.2.1 İstenildiğinde hizmet

Bir kullanıcı, sunucu zamanı ve ağ depolama gibi bilişim yeteneklerini ihtiyaç duyduğu anda, hiçbir servis sağlayıcı ve insan müdahalesi olmadan, tek taraflı olarak karşılayabilmelidir.

3.2.2 Geniş ağ erişimi

Farklı türde cihazlar tarafından (mobil telefonlar, tablet bilgisayarlar, dizüstü bilgisayarlar, iş istasyonları vb.) ağ üzerinde ve standartlaştırılmış mekanizmalarla erişilebilen yeteneklere sahip olmalıdır.

3.2.3 Kaynak havuzu

Sağlayıcıya ait depolama alanı, işlem gücü, bellek ve ağ bant genişliği gibi bilişim kaynakları, farklı fiziksel ve sanal kaynaklar üzerinde dinamik ve kullanıcı isteğine bağlı olarak tekrar tahsis edilebilen, bunun yanında birçok kullanıcıya aynı anda çoklu kullanım hizmeti verebilen bir havuzda bulunmalıdır. Kullanıcıların, kullanıma sunulan kaynakların konumları üzerinde kontrole sahip olmaları ve yerlerini tam olarak bilmeleri gerekmemektedir. Üst seviye bir bilgilendirmeyle yaklaşık olarak hangi ülkede veya veri merkezinde olduğu konusunda fikir sahibi olmaları Servis Seviyesi Anlaşmasına (Service Level Agreement – SLA) bağlı olarak sağlıyıcı tarafından sunulabilme imkânına sahip olabilmektedir.

3.2.4 Hızlı elastikiyet

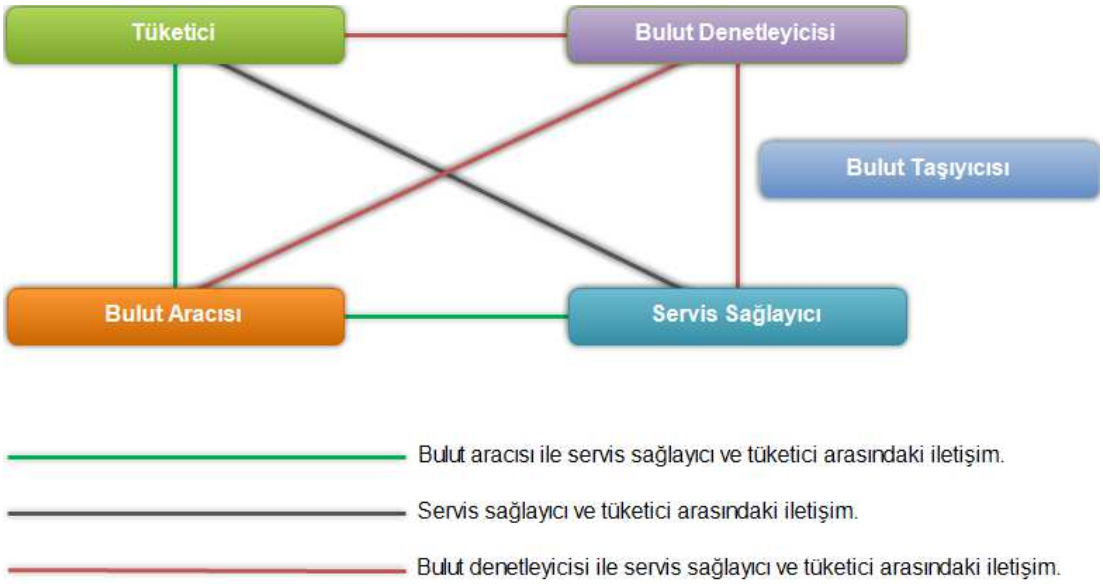
İmkân ve kabiliyetler elastik bir şekilde karşılanabilmeli, kullanım dışı bırakılabilmeli, bazı durumlarda otomatik ve hızlı bir şekilde içe veya dışa ölçeklenebilmelidir. Bu imkân ve kabiliyetler kullanıcılara genellikle sınırsız olarak lanse edilmekte ve ihtiyaçları olması durumunda, ihtiyaç miktarlarına göre tahsisleri sağlanmaktadır.

3.2.5 Düzenli servis

Bulut bilişim sistemlerinde hızlı bir şekilde verilen servisin çeşidine göre kaynak kullanımı gözlemlenebilmeli, kontrol edilebilmeli, raporlanabilmeli, şeffaf bir şekilde bunu sağlayıcı ile kullanıcıya eş zamanlı olarak sunabilmeli ve kaynakların kullanım miktarı belirlenebilmelidir.

3.3 Bulut Bilişim Mimarisi Alt Yapısı

National Institue of Standards and Technology (NIST) tarafından tanımlanan bulut bilişim mimarisi, bulut bilişim tanımı içerisinde beş temel aktör tanımlamaktadır [21]. Bunlar; tüketici, servis sağlayıcı, bulut denetleyicisi, bulut aracısı ve bulut taşıyıcısıdır. Her aktör bulut bilişim içerisindeki bir işlem veya görevi gerçekleştiren bir varlıktır. Şekil 3.2’de aktörler arasındaki etkileşim gösterilmektedir.



Şekil 3.2 : Bulut bilişimde aktörler arasındaki etkileşim.

3.3.1 Tüketici (Cloud consumer)

Tüketici, bulut bilişim hizmetinin desteklediği son kullanıcı olup, servis sağlayıcı tarafından sağlanan bulut servisini kullanan bir şahıs veya organizasyonel birim olabilmektedir. Servis sağlayıcının sunduğu hizmetler arasından ihtiyaçlarına binaen tercih ettiği uygun servis veya servisleri kullanmaktadır. Kullandığı bu servisler, servis sağlayıcı ile yapmış olduğu sözleşme dâhilinde ücretlendirilmektedir. Servis ihtiyaçlarına bağlı olarak, kullanım senaryolarının farklılıkları Çizelge 3.2'de gösterilmiştir [21].

Çizelge 3.2 : Tüketici ve Servis Sağlayıcı hareketleri.

Servis Türü	Tüketici Hareketleri	Sağlayıcı Hareketleri
IaaS	IT altyapı işlemleri için servis geliştirmekte/kurmakta, yönetmekte ve denetlemektedir.	IaaS kullanıcıları için fiziksel işlem, depolama, ağ ile hosting ortamını ve bulut altyapısını karşılamaktadır.
PaaS	Bulut ortamında bulunan uygulamaları geliştirmekte, test etmekte, dağıtımını gerçekleştirmekte ve yönetmektedir.	Platform kullanıcıları için bulut altyapısı, ara katman yazılımları, geliştirme ortamı, dağıtım ve yönetim araçlarını karşılamakta ve yönetmektedir.
SaaS	İşletme işlemlerini gerçekleştirmek için uygulama ve servisleri kullanmaktadır.	Bulut altyapısında uygulama yazılımlarını kurmakta, yönetmekte, bakımını yapmakta ve gerekli desteği vermektedir.

3.3.2 Servis sağlayıcı (Cloud Provider)

Servis sağlayıcı, şahıs, organizasyon veya tüketicilerin kullandığı servisleri geliştirmekten sorumlu bir varlık olabilmektedir. İhtiyaç duyulan yazılım/platform/altyapı servislerini yöneterek, servislerin sürekliliği için gerekli olan teknik altyapıyı oluşturmakta, güvenlik ve gizlilik başta olmak üzere belirlenen servis seviyelerini karşılamaktadır. Sağlayıcılar Çizelge 3.2'de belirtildiği gibi, farklı modeller için farklı görevler üstlenebilmektedir.

3.3.3 Servis denetleyicisi (Cloud auditor)

Servis denetleyicisi, bağımsız bir şekilde bulut servislerine, bilgi sistemi işlemleriyle performans ve güvenlik uygulamalarına bağlanabilen üçüncü parti kişi veya organizasyonlar olup, servis sağlayıcısını güvenlik kontrolleri, gizlilik uygulamaları ve performans gibi ölçütler kapsamında denetlemektedir.

3.3.4 Servis aracısı (Cloud broker)

Bulut bilişimin yönetimi, doğası gereği tüketiciler açısından çok karmaşıktır. Tüketici bulut servislerini, doğrudan servis sağlayıcısıyla iletişime geçerek değil, bir servis aracısının yardımıyla kullanabilmektedir. Servis aracısı, bulut servislerinin tüketici ile servis sağlayıcı arasındaki ilişkilerini düzenleyen, performans ve kullanılabilirliği yöneten bir varlıktır.

3.3.5 Servis taşıyıcı (Cloud carrier)

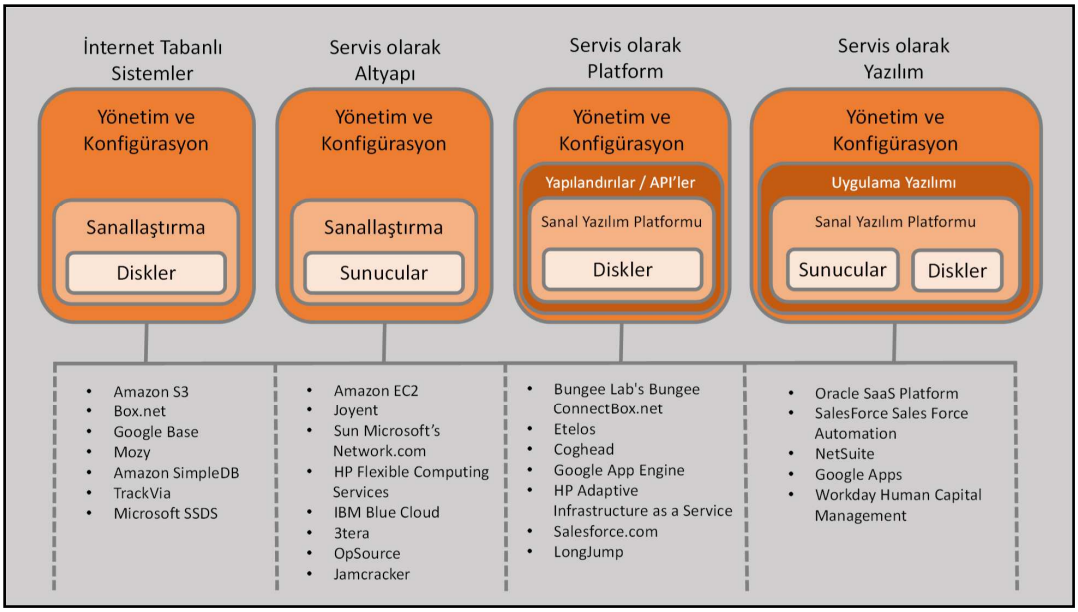
Servis taşıyıcı, bulut servislerinin servis sağlayıcılar ile tüketiciler arasındaki bağlantıları ve aktarımlarını gerçekleştirmekte olup, tüketicilerin servislere ağ, haberleşme altyapıları ve diğer cihazlar üzerinden erişimlerini sağlamaktadır. Tüketiciler bulut servislerine masaüstü veya dizüstü bilgisayarlar ile mobil cihazlar gibi ağ erişim aygıtları üzerinden erişebilmektedir. Bulut servislerinin dağıtımı normal olarak ağ ve haberleşme altyapısı veya yüksek kapasiteli depolama imkânlarının kullanıldığı fiziksel iletişim ortamlarına sahip iletişim ajanları aracılığıyla gerçekleştirilebilmektedir.

Servis sağlayıcının, servis taşıyıcısıyla servislerin istikrarlı bir şekilde iletimine yönelik olarak SLA hazırlaması gerekmekte, servis taşıyıcıları da servis sağlayıcısına

ayrılmış bir altyapı ile servis sağlayıcısı ve tüketici arasında şifreli bağlantıları gerçekleyebilmelidir.

3.4 Bulut Bilişim Servis Modelleri

İnternet tabanlı olarak sunulan depolama, aracı yazılım ve veri tabanı uygulamaları Şekil 3.3'te örneklendirilmiştir [22]. Servis olarak Altyapı hizmeti, internet üzerinden bir bilgisayarın tamamıyla kullanımı; Servis olarak Platform hizmeti, bir uygulamanın geliştirilmesi ve bunun diğer kullanıcılarla birlikte eş zamanlı olarak erişilmesi; Servis olarak Yazılım hizmeti, kurumsal kaynak yönetimi yazılımları gibi anahtar teslim uygulamaları bünyesinde barındırmaktadır.



Şekil 3.3 : Bulut bilişim hizmeti sunan bazı sağlayıcılar ve sundukları hizmetler.

En üst seviyede uygulama katmanında bulunan ve istendiği anda erişilebilen uygulamalar Servis olarak Yazılım modeli, ara katman yazılımları ve OS gibi uygulamalar Servis olarak Platform modeli, bulut uygulamaları için esnek bir çalışma ortamı sunan ve genelde donanımsal hizmetlerden oluşan Servis olarak Altyapı modeli kapsamında değerlendirilmektedir [16, 18, 21]. İnternet üzerinden birbirleriyle bağlantılı dağıtık veri merkezleri üzerinden hizmet verilmesi bulut bilişim yaklaşımının temel üç hizmet modelini oluşturmaktadır.

3.4.1 Servis olarak altyapı (Infrastructure as a service - IaaS)

Bu yetenek sayesinde kullanıcılar sağlayıcının imkân ve kabiliyetleri kapsamında işlem gücü, depolama alanı, ağ ve diğer temel bilişim kaynaklarına erişebilmekte, bu kaynaklar üzerinde OS ve uygulamalar dâhil olmak üzere her türlü yazılımı dilediğince kullanabilmektedir. Kullanıcılar bulut altyapısını kontrol ederek yönetmek yükümlülüğüne sahip değil, sadece OS, depolama ve dağıtımı yapılmış olan yazılımlar ile kullanılacak olan ağ bileşenlerinin seçimi gibi kısıtlı yapılandırma alanları üzerinde söz sahibidir.

IaaS kapsamında, fiziksel işlem, depolama, ağ ve temel bilişim kaynakları sağlayıcı tarafından tedarik edilerek, ev sahipliği (hosting) ortamı ve IaaS tüketicileri için gerekli olan bulut altyapısı oluşturulmaktadır. Tüketiciler, uygulamaların dağıtımını yaparak ve bu uygulamaları çalıştırarak hosting ortamında diğer servislere oranla daha fazla yetkilere sahip olurken, fiziksel sunucu, ağ, depolama ve hipervizör gibi bulut altyapısı kapsamındaki varlıklar üzerinde herhangi bir işlem gerçekleştirememektedir. Tüketicilere, ihtiyaçları doğrultusunda gerekli yazılımların dağıtımlarını yapabilecekleri ve kullanabilecekleri, VM'lere, ağ depolama birimlerine, ağ altyapısında kullanılan bileşenlere ve diğer temel bilişim kaynaklarına erişim imkânı sunulmakta olup, IaaS tüketicileri genellikle sistem geliştiricileri ile sistem ve IT yöneticileri olmaktadır.

3.4.2 Servis olarak platform (Platform as a service - PaaS)

Bu yetenek sayesinde kullanıcılar bulut bilişim altyapısı üzerinde programlama dilleri, kütüphaneler, servisler ve sağlayıcı tarafından desteklenen diğer araçlar vasıtasıyla yazılım geliştirebilmekte ve bu yazılımları kullanabilmektedir. Kullanıcılar bulut bilişim altyapısında bulunan ağ, sunucu, OS ve depolama alanlarını kontrol etmek ve yönetmek yükümlülüğünde olmamakla birlikte, sadece yazılımların koştuğu ortamlar üzerinde kısıtlı yapılandırma değişiklikleri yapabilmektedir.

PaaS kapsamında, kullanılacak platform için gerekli olan bulut altyapısı sağlayıcı tarafından yönetilmekte, platform için ihtiyaç duyulan araç ve yürütme kaynakları karşılanmakta ve geliştirilmekte, test edilmekte, dağıtımı ve uygulamaların sistem yöneticiliği yapılmaktadır. Tüketiciler, uygulamalar ile hosting ortamına ait ayarları kontrol etmekte, ancak altyapı ile ağ, sunucu, OS ve depolama alanlarına erişim sağlayamamaktadır.

3.4.3 Servis olarak yazılım (Software as a service - SaaS)

Bu yetenek sayesinde kullanıcılar sağlayıcı tarafından bulut bilişim altyapısında desteklenen yazılımları kullanabilmekte, uygulamalara ince istemci ve ağ tarayıcıları gibi farklı cihaz ve ara yüzler üzerinden erişebilmektedir. Kullanıcılar bulut bilişim altyapısı, ağ, sunucu, OS, depolama alanı ve diğer tüm uygulamaları kontrol etme yükümlülüğüne sahip olmamakta, sadece hizmet aldıkları yazılım kapsamında kısıtlı bazı yapılandırma değişiklikleri gerçekleştirebilmektedir.

Servis sağlayıcılar, bulut altyapısı üzerinde yazılım uygulamalarını dağıtmakta, yapılandırmakta, bakımını ve gücelleştirmelerini yapmaktadır. Sağlayıcılar altyapının yönetim ve kontrol sorumluğunu üzerinde bulundururken, tüketici uygulamaların yönetimsel sorumluluğuna sahiptir.

3.4.4 Servis olarak herhangi bir şey (Anything as a service – XaaS)

Bulut bilişim sistemleri üzerinde alınan hizmet kapsamında oluşturulabilecek ve isimlendirilebilecek her türlü servisi tanımlamak için kullanılan bir tabirdir. Alınan servis türüne bağlı olarak; Servis olarak Depolama (Storage as a Service -SaaS), Servis olarak Haberleşme (Communication as a Service - CaaS), Servis olarak Ağ (Network as a Service - NaaS), Servis olarak Gözlem (Monitoring as a service - MaaS) ve Servis olarak Bulut Yer Değişimi (Cloud Migration as a Service - C-MaaS) bu kapsamda örnek olarak verilebilir.

3.5 Bulut Bilişim Dağıtım Modelleri

Bir organizasyon; yönetim, maliyet ve güvenlik gibi gerekçeler nedeniyle, bulut bilişim sistemini kendisi yönetebilir veya bunu üçünü parti şahıs veya organizasyonlara yaptırabilir [16]. İhtiyaç ve gereksinimlere bağlı olarak şekillendirilebilen ve birbirine geçişi esnek bir şekilde gerçeklenebilen dört çeşit bulut bilişim dağıtım modeli bulunmaktadır.

3.5.1 Özel bulut

Bir organizasyonun iç kullanımında kullanılmakta ve organizasyon tarafından teşkil edilmiş olan, kendisi, üçüncü parti şahıs ve organizasyonlar ile bunların birleşimi tarafından yönetilebilmekte ve işletilebilmektedir. Temel olarak bulut bilişim altyapısı herkese değil, sadece organizasyon bünyesine açıktır.

3.5.2 Topluluk bulutu

Belirli bir topluluğu oluşturan organizasyon ve kişiler tarafından ihtiyaç duyulan görevler, güvenlik gereksinimleri ve politikalar kapsamında oluşturulmakta, bir veya daha fazla organizasyon veya kişi tarafından teşkil edilip, yönetilip, işletilebileceği gibi, tüm bunlar üçüncü bir kişi, şirket veya bunların birleşimi tarafından gerçekleştirilebilmektedir.

3.5.3 Genel bulut

Bir şahıs veya organizasyon tarafından işletilen ve üçüncü parti kullanıcılara bulut servisleri sunan bulut bilişim çözümüdür. Bulut altyapısı herkesin kullanımına açık olup, genel anlamda iletişim altyapısı olarak internet kullanılmaktadır. Bir işletme, akademik kurum ve devlet tarafından teşkil edilip, yönetilip, işletilebileceği gibi, tüm bu faaliyetler bunların birleşimi sonucu oluşan bir görev dağılımıyla da gerçekleştirilebilmektedir.

3.5.4 Melez bulut

Bulut bilişim altyapısı diğer çözüm türlerinden en az ikisinin bir arada kullanılmasıyla oluşturulan yapılardır. İhtiyaç duyulan görevler, güvenlik gereksinimleri ile politikalar ve teknolojik imkânlar dâhilinde yük dağılımı gibi veri ve uygulama taşınabilirliğini gerçekleştirebilecek şekilde teşkil edilmektedir.

3.6 Bulut Bilişimin Avantajları

Kendi IT sistemlerini işleten kurumlar, ihtiyaçları olan donanım ve yazılım altyapılarını kurarak, zaman ve maliyet açısından verimli olmayan bir şekilde bunları işletecek olan personeli de çalıştırmak durumundadır. Bulut bilişim, zaman ve maliyet açısından etkin olmayan bu durumu ortadan kaldırmaktadır. İhtiyaca binaen ortaya çıkabilecek aşırı işlem gücü veya depolama kapasitesi, kaynakların verimli olarak kullanılamamasına neden olabilmektedir. Birçok kurumsal veri merkezi sahip oldukları kaynak kapasiteni yüzde 50'nin altında kullanmaktadır. **Hazır bulunurluk;** bulut bilişimi savunan kişiler tarafından, yüksek donanım gücünü barındıran altyapıya sahip veri merkezleri üzerinde koşan sistemlerin hazırda bulunurluklarının daha yüksek olacağı genel olarak kabul görmektedir [22]. **Uygulama uyumu ve desteği;** iyi tasarlanmış bulut bilişim sistemleri üzerinde bulunan karmaşık yapılara sahip güçlü

uygulamalarla bütünleştirilecek (entegre edilecek) birçok servis daha kararlı olarak çalışacaktır. **Esneklik;** çoğu bulut bilişim sağlayıcısı, kullanıcıların çalıştırdıkları servislere ilişkin olarak anlaşmalara ihtiyaç duymamaktadır. Böylelikle yeni servis ve ürünlerin test edilebileceği ilave kaynaklara ihtiyaç duyulduğunda, bulut bilişim etkin bir çözüm olarak öne çıkmaktadır. İnternet Veri Kurumu (Internet Data Corporation – IDC) tarafından Ağustos 2008 ayında 244 kişinin katılımıyla gerçekleştirilen ve sonuçları Şekil 3.4'de gösterilen ankette, kullanıcılar açısından bulut bilişimin faydaları ağırlıklı olarak kullanılan servis kadar ücretlendirilmesi ile hızlı ve kolay dağıtım yapılması olarak ön plana çıkmaktadır. [23].

Şekil 3.4 : Kullanıcılar açısından bulut bilişimin genel faydaları.

3.6.1 Düşük donanım maliyeti

Tarayıcılar üzerinden çalışan uygulamaları kullanmak için yüksek performanslı donanımlara ihtiyaç yoktur [24]. Bu tarz uygulamaları çalıştırmak için akıllı telefonlar dahi yeterlidir. Bunun nedeni uygulamaların bulut bilişim sistemleri üzerinde çalışıyor olmasıdır. Asgari kaynak tüketimiyle günümüz teknolojisinde kullanılan en güçlü kişisel bilgisayarların performanslarına ulaşılabilmektedir. Bu nedenle bulut bilişim kullanıcılarının düşük kapasiteli sabit disk, bellek veya işlemciye sahip olmaları herhangi bir performans kaybına neden olmamaktadır. Bundan dolayı düşük donanımlara sahip bilişim sistemlerinin kullanılması sonucunda maliyetler de düşmektedir.

3.6.2 Gelişmiş performans

Geleneksel IT sistemleriyle kıyaslandığında, bulut bilişim sistemleri daha az kaynağa ihtiyaç duymakta olup, kullanılmakta olan daha az sistem kaynağı ile ihtiyaç duyulan hizmetler karşılanabilmektedir.

3.6.3 Düşük yazılım maliyeti

Bulut bilişim sistemlerinde her kullanıcı bilgisayarı için ayrı lisanslama ücretine sahip yazılımların kullanımına gerek kalmamakta ve kullanıcıların ihtiyaç duyduğu yazılımlara uygulama servisleri aracılığıyla erişim sağlanmaktadır. Kurumsal kullanıcıların büyüklüğü arttıkça, yazılımların dağıtımı ve yönetimi ilave maliyet yükleri olarak ortaya çıkmaktadır. Ancak bulut bilişim kabiliyetleriyle kullanılacak olan yazılımlara bir servis aracılığıyla erişimin sağlanmasından dolayı, geleneksel IT sistemlerindeki maliyetler ortadan kalkmaktadır. Satın alınması gereken yazılımlar bir servis olarak kiralanabilmekte, bu doğrultuda maliyetler düşürülmektedir.

3.6.4 Anında güncelleme

Kullanılan uygulamaların yeni sürümlerinin dağıtımları gerçekleştiğinde veya bu yazılımlara ait güncellemeler yayınlandığında, en güncel sürüm bulut bilişim üzerinden eş zamanlı olarak kullanılmaya başlanabilmektedir. Kullanıcılar bunlar için ilave ücret ödemek zorunda kalmamakta ve bu işlemleri gerçekleştirmek amacıyla sistem yöneticisi veya teknisyene ihtiyaç duymamaktadır.

3.6.5 Yüksek depolama kapasitesi

Kişisel bilgisayarlarda kullanılmakta olan 1,5 terabayt kapsaiteli bir sabit disk ile sunucu bilgisayarlarda kullanılan ağ yeteneğine sahip 1 petabayt (1 petabayt = 1024 terabayt) kapasiteli sabit disklerin depolama kapasiteleri kıyas kabul edemeyecek boyutlardadır.

3.6.6 Veri güvenliği

Kişisel bilgisayarların donanımları kolaylıkla arızalanabilmekte, en kötü senaryoda sabit diskler kullanılamaz ve içerdiği verilere ulaşılamaz duruma gelebilmektedir. Bulut bilişim sistemleriyle kullanıcıların verileri bulut içerisinde depolanmakta, yedekleme ve felaket kurtarma tedbirleri alınmakta, sonuç olarak ta herhangi bir veri kaybına imkân verilmemektedir.

3.6.7 İşletim sistemleri arasında uyum

Farklı çekirdek yapılarına sahip işletim sistemleri arasında ağ bağlantısı ve veri alışverişinde bulunmak nispeten kolay değildir. Windows işletim sistemine sahip bir bilgisayar ile MacOS veya Linux dağıtımına sahip bir bilgisayar arasında veri iletiminde yaşanması muhtemel sorunların üstesinden, işletim sistemlerinden bağımsız olarak verilerin sunucularda bulunması sonucunda gelinebilmektedir.

3.6.8 Dosya formatı uyumu

Geleneksel IT sistemlerinde bir dokümanın hazırlanan bilgisayar dışında açılması durumunda nasıl bir tabloyla karşı karşıya kalınacağı belirsizdir. OS, dokümanın hazırlandığı yazılımın veya sürümünün farklılığı ortadan kalkmaktadır. Web tabanlı bir doküman işlemcisinde hazırlanan bir doküman, erişilen her cihazda aynı şekilde görüntülenebilmektedir. Bunun yanında geleneksel IT sistemlerinde bir doküman işlemcisi yazılımında hazırlanan bir doküman, aynı yazılımın eski sürümlerinde görüntülenemeyebilmekte, bunun için ilave yazılım veya yamaların kurulması gerekebilmektedir.

3.6.9 Grup çalışması

Bulut bilişim sayesinde aynı belge üzerinde birden fazla kişi düzenleme yapabilmektedir. Geleneksel IT sistemlerinden farklı olarak bir doküman üzerinde aynı anda birden fazla kişinin işlem yapabilmesine imkân sağlanmaktadır. Dosyalar üzerinde herhangi bir işlem gerçekleştirmek isteyen bir kişi dokümanın en güncel haline çevrim içi olarak erişebilmektedir.

3.7 Bulut Bilişimin Dezavantajları

3.7.1 Sabit ağ bağlantısı gereksinimi

Kullanılan cihazının ağ bağlantısı bulunmuyorsa, web tabanlı bir uygulama veya bir servise erişim sağlanması mümkün olmamakta, bu şekilde ihtiyaç duyulan servisten yararlanılamadığı için mevcut gereksinimler aksayabilmektedir.

3.7.2 Düşük hızlarda düzgün hizmet alınamaması

Tarayıcı tabanlı uygulamaların yüksek bant genişliğine göre tasarlanması nedeniyle çevirmeli bağlantı veya Genel Paket Radyo Servisi (General Packet Radio Service –

GPRS) gibi düşük bant genişliklerine sahip bağlantılarda bulut bilişim servis ve uygulamaları performanslı çalışamayabilmektedir. Bunun yanında geniş bant ağ bağlantısının asimetrik indirme ve yükleme oranları nedeniyle yedekleme, güncelleme ve yükleme işlemleri geleneksel IT sistemlerine oranla daha yavaş gerçekleşebilmektedir.

3.7.3 Uygulamanın yavaş çalışması

Geniş bant ağ bağlantısına sahip olunması durumunda dahi uygulama veya hizmet alınan servisi destekleyen sunucunun aşırı yoğun olması nedeniyle kullanılan servis yavaş çalışabilmektedir.

3.7.4 Güvenlik açıkları

Mevcut tüm bilgilerin bulutta bulunması erişilebilirlik açısından büyük bir avantaj sağlasa da kötü niyetli servis sağlayıcılar veya sistem yöneticileri tarafından bu bilgilere erişildiğinde verilerin kötü amaçlarla kullanılması veya hiç kullanılamaması söz konusu olabilmektedir.

3.7.5 Sistem güncellemeleri

Bulut bilişim sistemleri üzerinde kullanılan bir yazılım, bulut altyapısı güncellendiğinde sorunlarla karşılaşabilmektedir. Servis sağlayıcının tecrübeli çalışanlara sahip olarak sistem ve yazılım güncellemelerini bir avantajdan dezavantaja dönüştürmemesi gerekmektedir.

3.7.6 Deneyimsiz bulut operatörü

Servis sağlayıcı, bulut bilişim sistemi altyapısı, platformu veya servislerinde gerekli bakımları gerçekleştirmez ise, oluşabilecek arızalarda servis hizmetinin aksamasından daha önemli olan verilerin bir daha erişilemeyecek şekilde kaybolabilmesidir.

3.7.7 Kullanılan yazılımların özellikleri

Tarayıcı tabanlı uygulamalar ne kadar platform bağımsız olarak çalışabilseler de geleneksel IT sistemlerinde kullanılan yazılımlarla aynı özelliklere sahip olamamaktadır. Bulut bilişim üzerinden servis tabanlı yararlanılacak hizmetlerde kullanılan yazılımların ihtiyaçları tam olarak karşıladığından emin olunması gerekmektedir.

3.8 Bulut Bilişimde Güvenlik

IDC tarafından Ağustos 2008 ayında 244 kişinin katılımıyla gerçekleştirilen ve sonuçları Şekil 3.5'te bulunan ankette, kullanıcıların bulut bilişim sistemlerine geçişte yaşadığı en büyük çekincenin güvenlik konusunda olduğu sonucu ortaya çıkmıştır. Aynı anket bir yıl sonra Ağustos 2009'da 263 kişinin katılımıyla gerçekleştirildiğinde güvenliğin kullanıcılar tarafındana öneminin arttığı görülmüştür.

Şekil 3.5 : Kullanıcılar tarafından bulut bilişime geçişin çekinceleri.

3.8.1 Hizmet alınan firmaların güvenliği

Kullanıcıların güvenlik açısından dikkate alacakları kısım, bulut servisi hizmeti alacakları servis sağlayıcıların sahip olduğu tecrübeler olmaktadır. Hizmet sunan sağlayıcının bulut sistemleriyle çalışma geçmişi ve mevcut güvenlik gereksinimlerine cevap verme durumlarıyla, sağlayıcının güvenlik gereksinimlerini karşılayamaması sonucunda oluşacak zaafiyetler kapsamında kullanıcıya karşı yapacağı bilgilendirme de önemli bir yere sahiptir. Sağlayıcının sahip olduğu yetenekler kapsamında; uyguladığı yöntem ve standartlar (Control Objectives for Information and Related Technology (COBIT), The Information Technology Infrastructure Library (ITIL), ISO 9001, 27001) ve bünyesinde bulunan çalışanların yeterlilik seviyeleriyle sahip oldukları sertifikasyonlar (Cisco Certified Network Associate (CCNA), Cisco Certified Network Professional (CCNP), Cisco Certified Internet Engineer (CCIE), Certified Information Systems Security Professional (CISSP), Certified Ethical Hacker (CEH), Certified Information Systems Auditor (CISA), Global Information Assurance Certification (GIAC) vb.) hizmet kalitesi açısından önemli göstergelerdir.

3.8.2 Erişim ve kimlik denetimi

Servis sağlayıcının, bulut servislerine erişim esnasında mutlak güveni sağlayacak bir bağlantı türü kullanması ve erişimin sadece kullanıcı tarafından gerçekleştirildiğinden emin olmak amacıyla bir takım gereklilikleri yerine getirmesi gerekmektedir. Kimlik Doğrulama (Authentication) kapsamında kullanıcı tarafından bilinen bir şifre, kullanıcı tarafından sahip olunan anahtar, kullanıcıya ait tekil değerler (biyometrik kontroller) ve kullanıcıya ait Küresel Konumlandırma Sistemi (Global Positioning System – GPS)) veya İnternet Protokolü (Internet Protocol – IP) yer bilgileri kontrolleri, Yetkilendirme (Authorization) kapsamında kullanıcının hangi varlıklara erişim sağlayacağının yönetimi, Hesap verilebilirlik (Accountability) kapsamında bulut bilişim sisteminde gerçekleşen tüm olayların kayıt altına alınması gerekmektedir.

3.8.3 Erişilebilirlik

Servis sağlayıcı, sunduğu hizmetleri bir SLA ile garanti etmeli, servislerin desteklendiği sunucuların erişilebilirlik durumları gibi kendi standartlarını kullanıcıyla paylaşmalıdır. Felaket önleme senaryolarının sunulan servisler içerisine dâhil edilip edilmediği veya servis sağlayıcısının mali durumu erişilebilirliği doğrudan etkileyen faktörler arasındadır.

3.8.4 Fiziksel güvenlik

Servis sağlayıcıya ait altyapının bulunduğu veri merkezi içerisinde uyguladığı güvenlik standartları, sahip olduğu güvenlik sertifikaları ile aldığı güvenlik önlemleri, yanmazlık, kesintisiz güç kaynakları, doğal afetlere karşı aldığı tedbirler (deprem, sel, yangın vb.) ve bunlar üzerindeki denetim ve yönetim durumları yüksek seviyede olmalıdır.

3.8.5 Yasal ve operasyonel güvenlik

Sağlayıcının sunduğu hizmetleri karşılarken sahip olduğu lisanslar ile mevcut hukuksal mevzuatlar dâhilindeki yasal yükümlülükleri, tüketiciyle yapılan anlaşmalar içerisine dâhil edilmelidir. Sağlayıcının operasyonel hizmetlerinde çalışan kişiler ile yaptığı gizlilik anlaşmaları (Non-Disclosure Agreement – NDA) bulunmalı ve çalışanlarının özgeçmişlerinde hukuksal olarak bir sakınca bulunmamalıdır.

3.8.6 Veri ve altyapı güvenliği

Servis sağlayıcının kullanıcı verilerini yedekleme ve kalıcı silme kabiliyetleri özellikle gizliliğin kritik olduğu kullanıcılar açısından önemlidir. Bu kabiliyetler gerekirse servis sağlayıcı tarafından ayrı bir servis olarak sunulabilmelidir. Kullanıcıların fikri bilgilerini ya da kendileri için önem arz eden sunucularda saklanan verilerin şifrelenmesi, gerekirse güvenli silmede olduğu gibi bunun bir servis olarak sunulması gerekmektedir. Bunların yanında servis sağlayıcı altyapısını proaktif bir yaklaşımla denetlemeli ve izlemeli; güvenlik duvarı, saldırı tespit ve önleme sistemleri (Intrusion Detection and Prevention System - IDPS), zararlı yazılım tespiti ve anti virüs yazılımları gibi kabiliyetleri kullanmalıdır. Gerektiğinde kullanıcılara da bunları ayrı birer servis olarak sunmalı, sunulan servislere bu sistemlere ait kayıt ve raporlamalar da eklenmelidir.

3.9 Bulut Bilişimde Hukuki Problemler

Geleneksel IT sistemlerinin günümüzde hukuki olarak birçok probleme ve cevaplanayamayan soruya sahip olduğu düşünüldüğünde, bulut bilişimin hukuksal düzlemde beraberinde yeni problem ve sorular getirdiği yorumu rahatlıkla yapılabilmektedir [25]. Teknolojik anlamda bulut bilişimi anlayabilmek için taraf pozisyonunda olan kullanıcı (tüketici) ve hizmet veren (servis sağlayıcı, bulut aracısı vb.) arasındaki ilişkilerin iyi tanımlanması ve anlaşılması gerekmektedir.

Bulut bilişim iki veya daha fazla tarafa sahip olabilmektedir. Örneğin; geleneksel IT sistemlerini bulut içerisine taşımak isteyen bir kullanıcı ilk olarak bu hizmeti sunan servis sağlayıcı veya bulut aracısı ile anlaşmaktadır. Ancak servis sağlayıcı bu hizmeti kendisine ait IT alyapısını kullanarak değil, kiraladığı ve dünyanın herhangi bir yerinde bulunan sunucuları kullanarak verebilmektedir. Bu şekilde kullanıcı A servis sağlayıcısından aldığını zannettiği hizmeti aslında farklı bir B firmasından alıyor olabilmektedir. Bulut bilişimin beraberinde getirdiği problem ve sorunlar örnekteki durumda başlamaktadır. Bulut bilişim teknolojisinin getirdiği esneklik ve evrensel paylaşımcılık, paylaşılan verilerin nerede saklanacağı sorusunu ortaya çıkarmaktadır. Sınırları ortadan kaldırarak gerçekleşen bir veri akışı söz konusu olmakta olup, herhangi bir hukuksal ayrılığın oluşması durumunda bunun nerede gerçekleştiğinin belirlenememesi ihtimali çok yüksektir.

Bu kapsamda özellikle Avrupa Birliği ülkelerine hizmet veren firmalar açısından büyük sorunların oluşması yüksek ihtimal dâhilindedir. Avrupa Birliği ülkeleri dışındaki kurulumların gerçekleştirecekleri veya Avrupa Birliği ülkeleri dışında bulunan ülkelerde kiraladıkları sunucular üzerinden gerçekleştirecekleri hizmetlerde, sunucuların bulunduğu ülkelerin Avrupa Birliği kanunlarında belirtilen veri güvenlik seviyelerini karşılamaları gerekmektedir [26], [27]. Günümüzde Amerika Birleşik Devletleri'nde uygulanmakta olan standartların bunu karşılamıyor olması söz konusu durumun ciddiyetini ortaya koymaktadır.

Avrupa Ağ ve Bilgi Güvenliği Ajansı'nın (European Network and Information Security Agency – ENISA) yapmış olduğu araştırmalar sonucunda bulut bilişim teknolojisinin en fazla verilerin korunması, gizlilik, fikri mülkiyet, mesleki sorumluluk ve dış kaynak kullanımı konularında sorun yaratacağını değerlendirmektedir [28]. Servis sağlayıcıların bilgileri sakladığı sunucuların kendi kotrollerinde olmaması ya da sanal ortam üzerinde kontrol ediyor olması ve bunun sonucu olarak fiziksel sunucuların üçüncü şahısların kontrolünde bulunması nedeniyle gizlilik ve bilginin korunmasında standartların uygulanması konusunda güçlükler meydana gelmektedir. Bunun yanında, sunucuları fiziksel olarak kontrolleri altında bulunduran kişilerin dünyanın farklı yerlerinde bulunmaları, uygulanan mevzuat ve standartların farklılıklar göstermesi sorunun daha da büyümesine ve karmaşık bir hal almasına neden olmaktadır. Davalar dâhilinde bulut bilişimin kanıt olarak ele alınmasında da problemler bulunmaktadır. Elektronik ortamda ele geçirilen kanıtların destek olabilecek nitelikteki ilave kanıtlarla geçerlilik kazanması nedeniyle, bulut bilişim üzerinden elde edilecek kanıtların hangi boyutta geçerlilik kazanabileceği soru işaretidir.

3.10 Bulut Bilişim Uygulama Örnekleri

Bulut bilişim ile birlikte bir çok bireysel ve kurumsal kullanıcının hayatına giren ve aktif olarak kullanılan uygulamalardan bazıları Çizelge 3.3'te gösterilmiştir [24].

3.11 Bulut Bilişim Teknolojileri

Bulut bilişim teknolojisinin fiziksel altyapısını kurmak için teknolojik imkânlar dâhilinde birçok yöntem vardır. Bunlardan bazıları fiziksel olarak ayrı sunucuların

(dedicated servers) kullanıldığı geleneksel IT sistemleri yaklaşımı ile yapı itibariyle birbirinden farklılıkları bulunan grid mimari, blade mimari ve sanallaştırma teknolojileridir.

Çizelge 3.3 : Bulut Bilişim Uygulamaları.

Bulut Bilişim Uygulamaları	Uygulamanın Kullanım Amacı
Google Dokümanlar	Kelime, tablo ve sunum işlemcisinin bulut ortamında kullanılmasına imkânı vermektedir.
iCloud	Apple firması tarafından üretilen cihazlar arasında depolama ve senronizasyon imkânı sunmaktadır.
Office 365	Microsoft firmasının ofis uygulamalarının bulut altyapısı üzerinde çalışan sürümüdür.
Dropbox	Çevrimiçi yedekleme imkânını verileri bulut ortamında saklayarak gerçekleştirmektedir.
Evernote	Yazılı, sesli ve görsel olarak not almanın yanında birçok platformdan erişim imkânı sunmaktadır.
Zoho	Kelime, tablo ve sunum işlemcisi, CRM, veritabanı ve raporlama gibi çeşitli uygulamaları sunmaktadır.
NetSuit	ERP, muhasebe, sipariş yönetimi, stok, CRM ve elektronik ticaret uygulamalarını etkileşimli olarak kullanabilme imkânı sunmaktadır.
Salesforce	Web tabanlı CRM'in diğer uygulamalarla uyumlu olarak çalışabilmesine imkân tanımaktadır.
Diğer Uygulamalar	Amazon Elastic Cloud Computing, Amazon Simple Storage Service, GoGrid, Skytap, Sun Grid, Right Scale, Amazon Elastic Cloud Computing, Google App Engine, Python Django, Ruby on Rails, Heroku, Mosso, Windows Azure, Force.com, Microsoft Live Mesh, OAuth, OpenID, Amazon Simple Queue Service, Google Maps, Yahoo Maps, Bing Maps, Yandex Maps, Amazon Flexible Payments Service, Google Checkout, Pay Pal, Alexa, Google Custom Search, Yahoo Boss, Amazon Mechanical Turk, Duolingo

3.11.1 Grid mimari

Grid bilişim 90'lı yılların başlarında karmaşık hesaplamalar ve çok fazla veriye sahip bilimsel uygulamalarda kullanılmak üzere, yüksek performanslı bilgisayarların yüksek

hızlarda veri aktarımı imkânıyla birbirlerine bağlanması sonucu ortaya çıkmıştır [29]. Grid bilişim tutarlı, yaygın ve kolay erişim olanaklarıyla yüksek teknoloji imkânlarına sahip donanım ve yazılım mimarisi olarak tanımlanmaktadır.

Grid bilişim geleneksel kümeleme yapılarında yaşanan yüksek performanslı ağ, düzensiz kaynak paylaşımı ve düşük düğüm güvenilirliği problemlerinin üstesinden gelen büyük ölçekli kaynak paylaşımına ve yükek performansa sahip bir yaklaşım olup, dağıtık sistemlerin planlama stratejilerini etkilemektedir [30, 31]. Bir arada bulunan ve merkezi olarak kontrol edilen gridler herhangi bir arızanın meydana gelmesi durumunda sistemin genelinin çalışmamasına neden olmaktadır.

3.11.2 Blade mimari

Blade sunucular, bir ağ sunucusu veya bir kart gibi tek bir sistem bileşeninden ziyade bunların tamamını bir arada bulunduran bir yapıya sahiptir [32]. Her bir blade, sunucu üzerinde işlemci, bellek, sabit disk, ağ kartı ve port birimlerini bulundurmakta, tek bir blade sunucu üzerinde 16 veya daha fazla blade bulunabilmektedir. Blade sunucular yer tasarrufunun yanında daha az bakım ve idameye ihtiyaç duymakta, daha az enerji tüketmekte, daha az ısınmakta ve daha kolay yükseltilebilmektedir. Blade sunucular genellikle yeşil bilişim (green computing) olarak adlandırılan çevre dostu yaklaşım kapsamında tercih edilmektedir. Donanım endüstrisi blade sunucular ile ilgili olarak herhangi bir standarda sahip olmadığı için bir blade sunucu üzerinde sadece üretici firmasına ait blade'ler kullanılabilmektedir.

Blade mimarinin en büyük avantajlarından biri, her bir blade için ayrı bir güç kaynağına ihtiyaç duyulmamasıdır [33]. Bazı yapılarda birçok blade sunucusu kabin içerisine yerleştirilen tek bir güç kaynağından beslenebilmektedir. Bunun yanında kablolamanın ortadan kalkması ve en az seviyede yapılması nedeniyle bakım idame ve yönetim son derece kolaylaşmaktadır. Kabin içerisine yerleştirilen sunucularda her bir sunucu çevresel birimlere kendine ait kablolama ile erişirken, blade mimaride tek bir kablo birçok işlevi yerine getirebilmektedir.

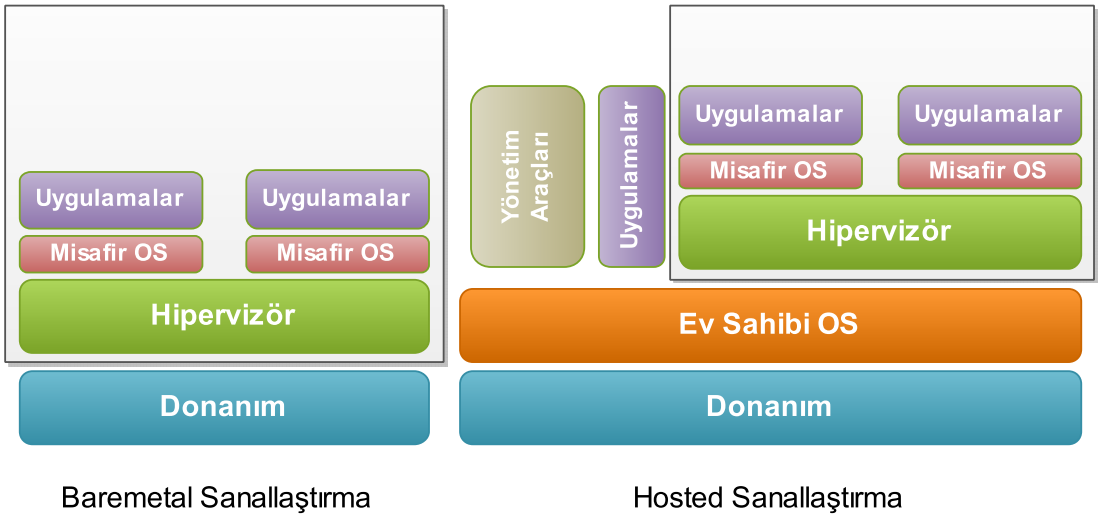
3.11.3 Sanallaştırma mimarisi

Bilişim kaynaklarının soyutlanması basitçe sanallaştırma olarak adlandırılmaktadır [34]. Fiziksel olmayan kaynaklar, donanım veya yazılım olabilmekte olup, IT sistemlerinin her seviyesinde uygulanabilmekte ve sanallaştırılan bu ortama Sanal

Makine (Virtual Machine – VM) denilmektedir. İhtiyaçlar doğrultusunda IaaS, PaaS veya SaaS bünyesinde kullanılabilmekte, ancak genellikle bulut bilişim uygulamalarında donanım üzerinde farklı sanal makineler üzerinde çoklu işletim sistemi koşturulmasına imkân veren OS sanallaştırması kullanılmaktadır.

Yarı Sanallaştırma: Yarı sanallaştırma işletim sistemine gerçek arayüzlerin yerine sanal arayüzler aracılığıyla donanıma erişim imkânı veren bir arakatman sunmakta olup, bu ara katmana hipervizör denilmektedir.

Tam Sanallaştırma: Sanallaştırılan donanım üzerinde birden fazla OS koşturulmasını sağlayan bir sanallaştırma şeklidir. İki çeşit tam sanallaştırma çeşidi bulunmaktadır. Hosted sanallaştırmada hipervizör katmanının altında çalışan bir OS bulunmaktadır. Baremetal (native) sanallaştırmada hipervizör katmanının altında bir OS çalışmamakta olup, hipervizör doğrudan donanıma erişmektedir. Şekil 3.6'da tam sanallaştırma çeşitleri farklı hipervizör mimarileriyle gösterilmiştir.



Şekil 3.6 : Tam sanallaştırma çeşitleri.

Hosted tipi tam sanallaştırılmada kullanılan bazı yönetimsel uygulamalar hipervizör katmanının altında çalışan OS'nin bünyesinde bulunmaktadır. Bu uygulamalardan bazıları sanal disk sürücüsü, sanal ağ yönetim uygulaması ve Sanal Makine Denetleyicisi (Virtual Machine Monitor – VMM)'dir. VMM, fiziksel kaynakların kontrol edilebilmesini sağlayan, aynı anda birden fazla OS'yi kontrol eden ve fiziksel donanım üzerinde çalışmasını sağlayan bir ara katman yazılımıdır. Tüm bu uygulamalar sanal ortamın yönetiminde kullanılmaktadır. Hosted sanal ortam, hipervizör katmanında çalışan OS üzerinde bulunan tarayıcılar veya antivirüs yazılımları gibi çeşitli son kullanıcı uygulamalarını bünyesinde bulundurabilmekte ve

masaüstü OS'lere ev sahipliği yapabilmektedir. Sunucular genellikle baremetal sanal ortamında bulunmaktadır. Her bir tam sanallaştırma çeşidinde, misafir OS'ler CPU, bellek, ağ arayüz kartı ile her türlü port ve çevre birimini de içeren sanal donanıma sahiptir. Bazı hipervizörler, doğrudan bellek erişimi (Direct Memory Access – DMA) desteğine de sahip olup, DMA'nın sayesinde VM'lerin performansları önemli ölçülerde artmaktadır.

Sistem sanallaştırması, bilişim kaynaklarının bir servis olarak hizmet vermesini sağlayan teknolojilerin başında gelmekte olması nedeniyle bir veri merkezinin ağ altyapısı üzerinde VM'ler oluşturulabilmesine imkân tanımaktadır [19]. Bunun yanında bulut sağlayıcıları, birden fazla kullanıcıya sunduğu hizmeti birden fazla sunucu üzerinden sunmak yerine, tek bir sunucu üzerinde oluşturulacak sanal sunucular üzerinden sunmaktadır [22]. Böylelikle, bulut sağlayıcıları kaynaklarını verimli ve etkin bir biçimde servis olarak sundukları hizmetlere ve kullanıcılara göre şekillendirebilmektedir. Kolay ölçeklenebilir ve esnek olması nedeniyle, kaynaklar kullanıcılara geçici özel projeler veya iş yüklerinin zirveye ulaştığı anlarda kolaylıkla tahsis edilebilmektedir. Yüksek performanslı donanımlar, birden fazla VM'ye sahip sanal organizasyonlar oluşturulmasına ve sanal ağlar gibi yönetimi kolaylaştıran ve maliyeti azaltan seçeneklerin kullanılmasına imkân vermektedir [20]. Bu ve buna benzer nedenlerden dolayı bulut bilişim çerçevesinde sanallaştırma teknolojileri daha fazla kullanılmaktadır. Fiziksel sunucular üzerinde yapılandırılan sanal ortamların yerine, blade sunucular üzerinde oluşturulan sanal ortamlar sayesinde daha ölçeklenebilir, daha az maliyetli ve bakım idamesi göreceli olarak daha kolay bir ortam yaratılabilmektedir.

3.12 Sanallaştırma Platformları

3.12.1 Microsoft Hyper-V

Microsoft firması tarafından geliştirilen Hyper-V Server, Windows sunucular üzerinde bir rol olarak kurulmaktadır [35, 36, 37, 38]. İşletim sistemin bir rol olarak kurulması nedeniyle OS üzerinde bir sistem servisi olarak çalışmaktadır. Sanal ortam yönetimi OS üzerinde bulunan araçlar ve uygulamalar vasıtasıyla gerçekleştirilmekte olup, kümeleme ve canlı göç gibi imkânlara sahiptir. Donanımsal olarak 32 çekirdeğe kadar CPU ve 1 terabayta kadar bellek desteği bulunmaktadır. Windows tabanlı bir sanallaştırma ürünü olması nedeniyle Linux veya Unix tabanlı sanallaştırma

çözümlerinde olduğu gibi komut satırı yerine görsel arayüzler aracılığıyla gerekli işlemler yapılabilmektedir. Bu özelliği nedeniyle Linux ve Unix tecrübesi olmayan kullanıcılar tarafından tercih edilmektedir.

3.12.2 VMWare

VMWare Server bireysel kullanım ve testler için ücretsiz olarak kullanılabilmekte olup kurumsal amaçlarla sınırlı bellek ve hissedilir seviyedeki düşük disk performansları nedeniyle tercih edilmemektedir [35, 39]. VMWare ESX/ESXi/ VSphere kurumsal sanallaştırma çözümlerinde kullanılmakta ve bu anlamda özellikle Linux ve Unix tabanlı diğer sanallaştırma çözümlerine rakip olmaktadır.

3.12.3 Sun Virtual Box

Sun firması tarafından geliştirilmiş olan Virtual Box, açık kaynak kodlu olması nedeniyle özellikle son kullanıcılar tarafından ücretsiz platformlar arasında en fazla tercih edilen sanallaştırma çözümüdür [35, 40]. Birçok sanallaştırma çözümünden farklı olarak ayarlanabilir görüntü belleği, uzak cihaz bağlantısı ve uzak masaüstü bağlantısını desteklemesi diğer tercih nedenleri arasındadır. Küçük ağlar ve bireysel kullanımlarda son derece başarılı olmasına rağmen, donanım üzerinde doğrudan çalışamaması nedeniyle özellikle yüksek performans ihtiyacı olan kurumsal çözümlerde tercih edilmemektedir.

3.12.4 Citrix Xen

Xen, Cambridge Üniversitesi'nde bir araştırma projesi kapsamında, daha sonra Xen Source isimli şirketi kuracak olan Ian Pratt liderliğinde geliştirilmiştir [41, 42]. Genel kullanıma ilk kez 2003 yılında sunulmuştur. Citrix Systems'in Ekim 2007'de Xen Source'u satın almasıyla bundan sonraki dönemde Xen Server ismini almıştır. 21 Ekim 2009 tarihinde Xen Cloud Platform (XCP) adı altında kullanıma sunulmuş ve tamamen açık kaynak kodlu hale gelmiştir. Citrix tarafından sahip olunmasına karşın hâlihazırda gönüllü bir topluluk tarafından ücretsiz bir yazılım olarak genel kamu lisansı GPL altında geliştirilmektedir. Xen Server herhangi bir işletim sistemine ihtiyaç duymadan doğrudan donanım üzerinde çalışmakta, böylelikle daha verimli ve ölçeklenebilir bir sistem olarak diğer sanallaştırma çözümleri arasında öne çıkmaktadır. Xen Server donanımı soyutlayarak üzerinde çalışan VM'lere paylaştırmaktadır. Xen sistemlerinde

hipervizör en altta bulunan ve en fazla yetkiye sahip katmandır. Bu katman fiziksel CPU'ları bir veya daha fazla işletim sistemine atayabilmektedir.

Xen terminolojisinde Xen Server üzerindeki ilk işletim sistemine Kontrol Etki Alanı (Control Domain – Dom0) denmektedir. Dom0 hipervizörün boot aşamasında otomatik olarak çalışmaya başlamakta ve fiziksel donanıma doğrudan erişerek bu kapsamda yönetim yetkisine sahip olmaktadır. Dom0, Xen Server üzerinde çalışan ve yönetim araçlarını üzerinde bulunduran güvenli VM'dir.

4. SALDIRI TESPİT SİSTEMLERİ

Saldırı tespiti, bilgi sistemlerinde alınan tüm önlemlere karşın gerçekleşen saldırıları, gerçek zamanlı veya daha sonrasında tespit etmek maksadıyla, zararlı davranışlarda bulunabilecek şüpheli olayları tespit eden, bilgi sistemi veya haberleşme ağı bünyesindeki alt sistemleri kötü yönde etkileyebilecek, her türlü paket ve verinin analiz edilmesiyle gerçekleştirilen basamakların birleşimidir [43, 44]. Saldırı tespit sistemleri ise, saldırı tespit işlemlerini gerçekleştiren ve tespit edilen bilgileri yedekleyen, bu doğrultuda alarm oluşturan veya daha önceden belirlenen bir takım prosedürleri yerine getiren sistemlerdir. Gözlemlenen bileşen bilişim kaynaklarını da içeren her çeşit donanım veya yazılım olabilmektedir. Yakalanan her şüpheli olay saldırı girişimi anlamına gelmemektedir. Bazı beklenmeyen olaylar nadiren gerçekleşebilmekte olup, bunların saldırı olup olmadığına karar vermek önemli bir noktadır.

Bir Saldırı Tespit Sistemi (Intrusion Detection System – IDS), korunan sistem üzerinden veri toplayan ve sistem yöneticilerini kayıt altına alma, e-posta veya sistemi tespit edilen saldırıya karşı koruma altına alma gibi çeşitli yollar aracılığıyla uyaran yazılım, donanım veya bunların hibrit bir bileşkesi olabilmektedir.

4.1 Korunan Ortama Göre Saldırı Tespit Sistemlerinin Sınıflandırılması

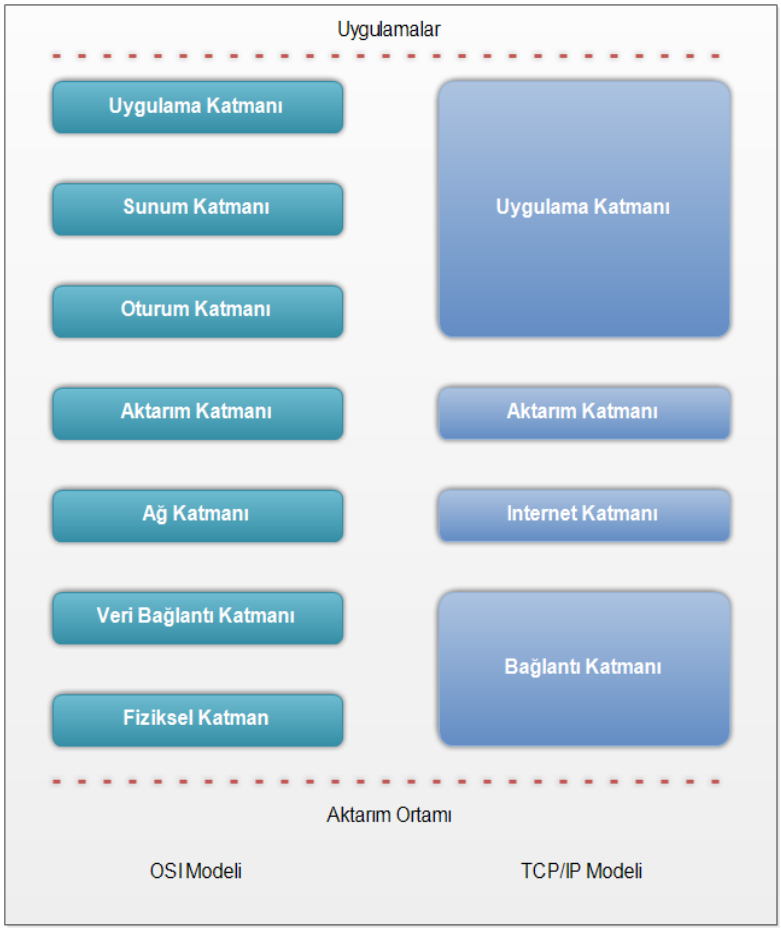
Korunan ortamın kapsamına göre çeşitli IDS yaklaşımları (Sunucu (Host) tabanlı, Ağ tabanlı, Kablosuz Ağ tabanlı, VMM tabanlı, Bulut tabanlı vb.) bulunmaktadır. Ancak temelde bunlar iki sınıf altında toplanmaktadır; Ağ Tabanlı Saldırı Tespit Sistemleri ve Sunucu Tabanlı Saldırı Tespit Sistemleri.

4.1.1 Ağ tabanlı saldırı tespit sistemleri

Ağ Tabanlı IDS'ler (Network-based IDS – NIDS) belli bileşenleri önceden tanımlanan ve belirlenen ağ trafiği bölümleri doğrultusunda gözlemlemekte, incelemekte ve analiz etmektedir. Dikkat edilen noktaya göre farklı türlerdeki durumları tespit edebilmekte ve genellikle bir ağ içerisinde uç noktalarda, özellikle sınır kısımlarındaki önemli

görevleri bulunan yönlendirici, güvenlik duvarı gibi cihazlar ile güvenli bir yerel alan ağı ile dış dünya arasındaki ağın (Demilitarized Zone – DMZ) veya sunucunun hemen ardında teşkil edilmektedir.

Bir NIDS ağ üzerinde yerleştirilerek trafik analizi yapmakta ve istenmeyen veya zararlı olan olaylar ile aktiviteleri aramaktadır. Ağ trafiği çeşitli katmanlar üzerinde inşa edilmiş olup, her katman diğer bir katmandan aldığı veriyi başka bir katmana iletmektedir. Açık Sistemler Arabağlaşımı (Open Systems Interconnection – OSI) modeli ve Aktarım Kontrol Protokolü/İnternet Protokolü (Transmission Control Protocol/Internet Protocol – TCP/IP) modeli bu katmanların ne şekilde düzenlendiğini ortaya koymaktadır. OSI ve TCP/IP referans modellerine ait katmansal yapı Şekil 4.1'de gösterilmiştir.



Şekil 4.1 : OSI ve TCP/IP referans modelleri.

TCP/IP modeliyle, en altta bulunan bağlantı (link) katmanı kontrol voltları ve donanımların fiziksel adreslerinin kontrolü gibi verinin kablo üzerinde ne şekilde iletildiğini açıklamaktadır. İnternet katmanı adres yönlendirmelerini (routing) kontrol

etmekte ve IP yığınını kapsamaktadır. İletişim katmanı veri akışını ve veri bütünlüğünü kontrol etmekte olup, Aktarım Kontrol Protokolü (Transmission Kontrol Protocol – TCP) ve Kullanıcı Veri İletisi Protokolü'nü (User Datagram Protocol – UDP) içermektedir. Son kısımda en karmaşık olan ancak programlar tarafından kullanılması nedeniyle en fazla aşina olunan uygulama katmanı bulunmaktadır. Uygulama katmanı trafiği Üstün Metin Aktarım Protokolü (Hypertext Transfer Protocol – HTTP), FTP ve Basit Posta Aktarım Protokolü (Simple Mail Transfer Protocol – SMTP) gibi protokolleri içermektedir. Çoğu NIDS her katmanda meydana gelen istenmeyen trafiği tespit edebilmesine rağmen, bu trafiğin büyük kısmı uygulama katmanında yer almaktadır. IP katmanındaki istenmeyen trafiğin tespiti için güvenlik duvarları kullanılmakta ve arzu edilmeyen trafiğin tespiti, tek bir varlığın üzerine yüklenmemeyerek iş bölümü yapılmaktadır.

NIDS'ler araç ve yazılım olmak üzere iki bileşenden meydana gelmektedir. Araç işletim sistemi, uygulama ve ağ kartından oluşmaktadır. Yazılım ise, IDS yazılımı ve bazen de kullanıcı tarafından kullanılan OS'den oluşmaktadır. Sadece yazılım olarak NIDS araç tabanlı sistemler ile karşılaştırıldığında genellikle daha az maliyetli olmakla birlikte, donanımla uyumlu çalışabilmeleri için ayrıca yapılandırılmaları gerekmektedir. NIDS genellikle bir cihazın parçası olmamakta, ancak birçok ayrı donanımı bünyesinde bulunduran ayrı bir fiziksel cihaz olabilmektedir.

Algılayıcılar; ağ trafiğini izleyen ve bu trafiğin zararlı olup olmadığına karar veren algılayıcı veya ajanlardır. Çoklu algılayıcılar genellikle ağ bünyesinde belirlenen noktalara yerleştirilmektedir. Ağ bağlantıları güvenlik duvarları, anahtarlama cihazları, yönlendiriciler veya ağı bölen diğer noktalara yerleştirilmektedir.

Veri Tabanı Sunucusu; NIDS'lerin depolama bileşenleridir. Algılayıcılardan gelen olaylar ve bu olaylarla ilişkili veriler bu sunucularda kayıt altına alınmaktadır. Veri tabanları yüksek depolama alanlarına sahip olmak ve performanslı çalışmak zorundadır.

Yönetim Sunucusu; analiz birimi olarak, tüm algılayıcılardan sonuçların gönderilebileceği merkezi bir noktada konumlandırılmaktadır. Yönetim sunucuları algılayıcılarla genellikle güvenlik gerekçeleri ile asıl ağdan bağımsız olarak yönetimsel bir ağ üzerinden bağlantı kurmakta ve algılayıcı üzerinden gelen raporlar doğrultusunda kararlar vermektedir. Farklı algılayıcılardan gelen veriler ile ağın farklı

bölümlerinde meydana gelen trafikler doğrultusunda ilişkilendirme işlemleri gerçekleştirilebilmektedir.

Konsol; NIDS'lerin kullanıcı ara yüzleridir. NIDS'lerin yönetici tarafından kontrol edilmesini, yapılandırılmasını ve mevcut durumun gözlenebilmesini sağlayan kısımlardır. Konsol yerel olarak çalışan bir yazılım olabileceği gibi güvenli bir web portalı da olabilmektedir.

Bileşenler arasındaki trafik güvenli bir şekilde gerçekleşmeli, hiçbir bileşen bu trafiği değiştirememeli ve görüntüleyememelidir. Engellenen trafik bir saldırganın saldırı trafiğini değiştirmesine neden olabilmektedir.

NIDS'lerde algılayıcılar ağ üzerinde farklı şekillerde konumlandırılabilmektedir.

Inline Konumlandırma: Ağı ikiye bölen yönlendirici veya güvenlik duvarı gibi iki cihaz arasına yerleştirilmektedir. Bu, iki cihaz arasında gerçekleşen trafiğin algılayıcı üzerinden geçmesi anlamında olup, algılayıcının trafiği analiz edebilmesini garanti etmektedir. Bu şekilde yerleştirilen bir algılayıcı zararlı olarak nitelendirilen trafiğin üzerinden geçmesini engellemek amacıyla kullanılabilmektedir. Bu tür algılayıcılar güvenlik duvarının arkasına ve iç ağın girişine yerleştirilerek, analiz edilecek trafiğin daha az olması sağlanmaktadır.

Pasif Konumlandırma: Pasif bir algılayıcı, ağ üzerinden geçen trafiği kendi üzerine kopyalayarak analiz yapmaktadır. Kopyalanan bu trafik birden fazla kaynağa sahip olabilmektedir.

Spanning Port Konumlandırma: Bir anahtarlama cihazı üzerinden geçen trafiğin tamamının tek bir porta kopyalanmasıdır. Düşük miktardaki ağ trafiğinde bütün trafiği gözlemlemek için kolay bir yöntem olmakla birlikte, ağ trafiği arttıkça anahtarlama cihazı tüm trafiği kopyalayamayabilmektedir.

Network Tap Konumlandırma: Bu tür algılayıcılar üzerinden geçen tüm trafiği fiziksel katmanda ve genellikle fiber optik kablolarda hat üzerine yerleştirilerek sinyal seviyesini düşürmeden kopyalamaktadır. Eğer doğrudan aktarım ortamına yerleştirilirlerse tüm bağlantının kesilmesine neden olabilmektedir.

NIDS'ler tarafından algılanabilen bazı olaylar sırasıyla örneklendirilmiştir. Tek başına keşif maksadıyla gerçekleştirilen olaylar tehlikeli olmamakla birlikte, tehlikeli saldırılara öncülük edebilmektedir. Bu tür olaylar ağ katmanında port taraması

şeklinde gerçekleşmektedir. Sistemler üzerinde çalışan servisler, izin verilen bağlantıların gerçekleştirilmesi amacıyla bazı portları açmaktadır. Bir port taraması esnasında saldırgan, her port üzerinden bağlantı kurmaya ve hangi servislerin çalıştığını tespit etmeye çalışmaktadır. Bu sayede bir web sunucusu üzerinde çalışan OS ve sürümü hakkında bilgi sahibi olunabilmektedir. NIDS'ler ağ, iletişim ve uygulama katmanındaki saldırıları da tespit edebilmektedir. Bu tür saldırılar servis dışı bırakma (Denial of Service – DoS) saldırıları ve bilgi hırsızlığında kullanılan zararlı kodları içerebilmektedir. NIDS'ler daha az tehlikeli olan ama istenmeyen trafikleri tespit etmek amacıyla da kullanılabilmekte, böylece arka kapılar gibi istenmeyen servisler ve politika ihlallerinin önüne geçilebilmektedir.

4.1.2 Sunucu tabanlı saldırı tespit sistemleri

Sunucu Tabanlı IDS'ler (Host-based IDS – HIDS) sistem çağrısı, işlem veya iplikçik (thread), varlık ve yapılandırma erişimi ya da değişikliği gibi şüpheli bir durumun oluşmasına karşın sunucu bir sistemin özelliklerini gözlemleyerek analiz etmektedir. Özellikle önemli sunucu sistemler üzerinde gizli ve kritik bilgileri korumak amacıyla kullanılmaktadır. HIDS'ler tek bir sunucu üzerine kurularak ağ davranışlarını algılamak için yapılandırıldıklarında NIDS görevi görebilmektedir.

HIDS'ler, belirli bir makinede meydana gelebilecek saldırıları önlemek üzere sunuculara ya da çalışma istasyonlarına yerleştirilmiş algılayıcılardan oluşmaktadır. Bir HIDS sadece ağ trafiğini değil daha fazlasını görebilmekte ve yerel ayarlara, bir OS'ye ait özel ayarlara ve log kayıtlarına dayalı olarak kararlar verebilmektedir. Diğer IDS'lerde olduğu gibi, HIDS'lerde de çeşitli tiplerde görevlendirilmiş varlıklar mevcuttur.

Algılayıcı veya ajan; sunucu, iş istasyonu ve uygulama hizmeti gibi bir sunucunun üzerine veya yakınına yerleştirilmektedir. Olay verisi kayıt altına alma hizmetine, olayları kaydetmek ve diğer muhtemel alakalı olaylarla bağlantı kurmak üzere gönderilmektedir. HIDS ajanları sayısız sunucu tipine yerleşerek, buralarda çalışabilmektedir. HIDS algılayıcıları sunucuları, kullanıcı bilgisayarlarını ve uygulama sunucularını izleyebilmektedir.

Sunucu; genellikle, kullanıcıların web, e-posta veya FTP sunucularına veri göndermek veya almak üzere bağlandıkları, kullanıcılara belirtilen hizmetleri vermek üzere adanmış olarak çalışan bir bilgisayardır.

Kullanıcı makinesi; bir kullanıcının diğer makinelere bağlanmasını sağlayan masaüstü veya dizüstü bilgisayarı gibi bir iş istasyonudur.

Uygulama hizmeti; web hizmeti veya veritabanı uygulaması gibi sunucu üzerinde çalışan bir yazılımdır.

Her bir kullanıcı makinesi farklı OS çalıştırmakta veya hizmet yürütmekte olduğu için, makinelere etki edecek saldırıların tipleri de söz konusu makinelere özgü olarak gerçekleşmektedir. HIDS algılayıcısı sadece ağ trafiğini değil, kullanıcı makinelerini de izlediği için ajan, kullanıcı makinelerinde yazılımın bir parçası olarak yer almaktadır. Mantıksal olarak ajan, varlık yani kullanıcı makinesi ve dış ağ arasında, geleneksel bir NIDS'e benzer şekilde yerleştirilmektedir. Ancak HIDS ajanı, bir ağ cihazı olmak yerine, trafiğin hizmet alabilmek için aşması gereken bir yazılım katmanı olarak görev yapmaktadır.

Bir HIDS, geleneksel NIDS'lerin algılayıcılarında olduğu gibi, ağ temelli saldırılara karşı sistemi izlemekte ve aynı zamanda kullanıcıya özel olayları da tespit edebilmektedir. Kullanıcıya özel olaylar; zararlı bir kodun çalıştırılması ve bellek taşması gibi kod analizlerini, bütünlük ve erişimi içerecek şekilde dosya sisteminin izlenmesini, kullanıcı kayıtları gözden geçirilirken meydana gelen kayıt analizlerini ve son olarak ağ ayarı yapılandırmalarındaki değişiklikleri kapsamaktadır.

4.1.3 Ağ davranışı anormallik tespiti

Ağ Davranışı Anormallik Tespiti (Network Behaviour Anomally Detection - NBAD), ağ trafiğinin şekilsel veya istatiksel durumundan, paketler ile tek tek ilgilenmeden, trafiğin kötü niyetli yani zararlı olup olmadığına karar vermeyi sağlayan bir saldırı tespit yöntemidir. NBAD algılayıcıları, ağ üzerinde anahtarlama cihazları, DMZ'ler ve trafiğin farklı bölümlere ayrıldığı yerler gibi kilit noktalara yerleştirilmektedir.

Algılayıcılar, görev bölgeleri içerisinde hangi tipte ve büyüklükte trafiğin aktığını bilerek, bu trafiğin incelemesiyle DoS saldırılarını, ağ boyunca tarama yaparak solucanları, beklenmedik uygulama servislerini ve kural ihlallarini tespit etmektedir.

Geleneksel kablolu NIDS'ler ve NBAD sistemleri, algılayıcılar ve yönetim konsolları gibi sahip oldukları aynı yapı içeriklerinin bazılarını paylaşmakta olup, geleneksel NIDS'lerin aksine NBAD sistemleri genellikle veritabanı sunucularına sahip değildir.

NBAD sistemleri, trafiğin temel çizgisinden ayrıldığı durumlarda, karar vermeye yönelik olarak çalışmaktadır. Bu durum DoS saldırılarının ve solucanların tespit edilmesinde genellikle faydalı olmaktadır. Diğer IDS'lerde olduğu gibi, NBAD sistemleri de kötü niyetli trafiğin engellenmesini, trafiğin akışını durdurarak önlemede faydalı bir şekilde görev alabilmektedir.

4.1.4 Kablosuz saldırı tespit sistemleri

Kablosuz teknolojilerin gelmiş olduğu nokta itibarıyla çok popüler olması ve kablosuz haberleşmenin doğası gereği ağlar arasındaki sınırların belirsizleşmesi nedeniyle saldırı tespit ve önleme yaklaşımları önemli bir çalışma alanı haline gelmiştir. Kablosuz IDS'ler, NIDS'ler ile benzer yapılardadır. Bunun nedeni, kablosuz ağların kablolu ağlar ile benzer saldırılara maruz kalmalarıdır. Kablosuz ağları geleneksel kablolu ağlardan ayıran fonsiyonalite ve zayıflıklarından dolayı bir kablosuz IDS, meydana gelebilecek kablosuz özgün saldırıları da normal ağ saldırılarında olduğu gibi izleyebilmektedir.

Kablosuz IDS algılayıcıları tüm kablosuz trafiği herhangi bir yönlendirme yapmadan izleyebilen bağımsız cihazlar olup, kablolu ağlara bağlandıklarında olduğu gibi, kablosuz ağlarda da erişim noktaları olarak görev yapabilmektedir. Bir kablosuz LAN algılayıcısının konumu, algılayıcının fiziksel konumunun kapsama alanına doğrudan etki etmektedir. Bir algılayıcı sadece kablosuz LAN'daki cihazları değil, aynı zamanda kablosuz ağa bağlanabilme ihtimali ve yeteneği olan cihazları da izleyebilmektedir. Kablosuz bir IDS'nin bünyesinde, geleneksel bir NIDS'de olduğu gibi algılayıcılar, yönetim sunucusu, kayıt veritabanı ve konsol bulunmaktadır.

Kablosuz algılayıcılar, kablolu ağlarda görüntülenebilen ve kablosuz ağlara özel meydana gelen olaylar gibi birçok tipte olayı izleyebilmekte ve takip edebilmektedir. Yetkilendirilmemiş LAN ve kablosuz cihazlar, düşük güvenlikli kablosuz cihazlar, alışılmadık veri kullanım yolları, kablosuz tarayıcılar, DoS saldırıları ve araya girme saldırıları gibi anormallikleri tespit edebilmektedir.

4.2 Saldırı Tespit Yöntemlerine Göre Saldırı Tespit Sistemleri

4.2.1 İmza (kural) tabanlı saldırı tespit sistemleri

İmza tabanlı IDS'ler bilinen saldırı çeşitlerine karşın önceden tanımlanan kurallar ile saldırı girişimlerini tespit etmekte olup, yeni saldırı tiplerine karşı başarılı değillerdir. Bu tespit mekanizmaları, olayları önceden tanımlanan kurallar ile karşılaştırdığı için uygulama ve yönetim açısından uygulanması en kolay yöntemdir. Daha önceden saldırı gerçekleştiren bir kaynağı kayıtlarında bulunduramamaktadır. Bu nedenle, imza tabanlı saldırı tespiti daha önce saldırı gerçekleşmiş bir kaynaktan gerçekleşebilecek karmaşık bir saldırıya karşı risk oluşturmaktadır.

Kullanıcı adı "root" olan bir telnet bağlantı girişiminin bir organizasyonun güvenlik politikası ihlali olabilmesi, "free pictures" konulu ve "freepics.exe" isimli bir dosya ekine sahip e-posta'nın bilinen bir zararlı yazılımın özelliklerinden oluşunu ya da "645" değerine sahip bir OS kaydı girişinin sunucunun kayıt özelliğinin devre dışı bırakıldığını belirtmesi kullanılan imzalara birer örnek olarak gösterilebilir.

İmza tabanlı tespitler, bilinen saldırı türlerine karşı başarılı olmalarına karşın, yeni tür saldırılara karşı etkili olamamaktadır. Örneğin; bir önceki örnekten yola çıkarak "freepics.exe" isimli bir dosya üzerinde gerçekleştirilecek bir modifikasyon sonrasında, bu dosyaya ait imza ile dosya arasında yapılacak bir karşılaştırmada eşleşme meydana gelmeyecektir.

İmza tabanlı tespitler, paket ve kayıt girdileri gibi mevcut durumları "string" birer ifade şeklinde oluşturulan imzalar ile karşılaştırmaları nedeniyle, uygulanabilirlik açısından en basit saldırı tespit yöntemidir. Bu yöntem, karmaşık haberleşmeleri takip edememeleri ve anlayamamaları nedeniyle, birçok ağ ve uygulama protokolü hakkında bilgi sahibi değildir. Örneğin; web sayfaları tarafından üretilen ve durum kodu "403" olan sunucunun reddetme isteği imza tabanlı tespit yöntemleriyle anlaşılamamaktadır. Bunun yanında mevcut bir isteği değerlendirirken, daha önce yapılan istekler hakkında bilgi sahibi de değildir. Bu kısıtlamalar nedeniyle imza tabanlı saldırı tespit yöntemleri, birden fazla durum bilgisini birleştirerek gerçekleştirilen karmaşık saldırıları tespit edememektedir.

4.2.2 Anormallik tabanlı saldırı tespit sistemleri

Anormallik tabanlı IDS'ler bir sistemin davranışlarını gözlemleyerek bu davranışları analiz etmektedir. Anormallik tabanlı tespit teknikleri, imza tabanlı tekniklerle birleştirilerek kullanıldığı takdirde saldırı tespitindeki başarı oranları artmaktadır. Genellikle yeni geliştirilen ve çoklu işlem veya ağ erişimi gerçekleştirerek bilişim kaynaklarını tüketen zararlı bir yazılım gibi bilinmeyen saldırı tiplerine karşı tercih edilmekte, bu nedenle de sistemin alışıldık davranışlarından farklı olarak ayırt edici tespitlerde bulunabilmektedir.

Anormallik tabanlı saldırı tespiti, normal olaylara ait tanımlamalar ile mevcut duruma göre dikkat çeken sapmaları karşılaştırmaktadır. Bir kullanıcı, sunucu, ağ bağlantıları veya uygulamalara ait profiller, belirli bir zaman içerisindeki belirli bir aktivitenin özelliklerini gözlemleyerek oluşturulmaktadır. Örneğin; bir ağa ait profil bir iş günü çalışma saatleri içerisinde ağ bant genişliğinin %13'ünün web hareketleri tarafından oluştuğunu gösterebilmektedir. Bir IDS mevcut bir aktivite özelliklerinde bulunan sınır değerler ile ilgili profile ait normallikleri karşılaştıran istatiki bir yöntem kullanmaktadır. Bir kullanıcı tarafından gönderilen e-posta sayısı, bir sunucu üzerinde gerçekleşen başarısız erişim girişimleri ve bir sunucu üzerinde belirli bir zaman aralığı içerisindeki işlemci kullanımı, normalliklerde meydana gelen sapmalara örnek olarak gösterilebilir.

Anormallik tabanlı saldırı tespit yöntemlerinin en büyük faydası daha önceden bilinmeyen tehditlere karşı oldukça başarılı olmalarıdır. Yeni geliştirilen bir zararlı yazılım bulaşan bir bilgisayar olduğunu ele alırsak, bu zararlı yazılımın bilgisayar kaynaklarını tüketmeye başladığında, yüksek miktarda e-posta gönderimi veya normalin çok üzerindeki miktarlarda ağ bağlantısı açılması rahatlıkla tespit edilebilecektir.

4.2.3 Durum bilgisi analizi tabanlı saldırı tespit sistemleri

Durum bilgisi analizi tabanlı IDS'ler bilinen protokollerin özelliklerini kullanmaktadırlar. Anormallik tabanlı yöntemlerden farklı olarak protokollerin yapılarını tanımlamakta, bu nedenle imza tabanlı IDS'lere benzemektedirler. Genel olarak kullanılan protokollerin önceden belirlenen tanımlamaları doğrultusunda oluşan ve rutin aktivitelerinde meydana gelen sapmaları temel alan bir yöntemdir. Protokolü geliştirenler tarafından evrensel olarak tanımlanan özellikler doğrultusunda, bu

protokollerin genel davranış ve hareketlerinden oluşan profilleri kullanmaktadır. “Durum” kelimesi, protokolün tasarımından kaynaklı olarak mevcut durumunun ne olması gerektiğini anlayabildiği manasına gelmektedir.

Bir kullanıcı Dosya Aktarım Protokolü (File Transfer Protocol – FTP) oturumu başlattığında, bu oturum ilk olarak doğrulama (authentication) gerçekleşmeden başlayacaktır. Bu durumda, doğrulanmamış (authentication gerçekleşmemiş) kullanıcı yardım bilgilerini görüntüleme gibi sadece temel birkaç komutu çalıştırabilecektir. Durumu anlayabilmenin önemli noktalarından biri istek (request) ve cevapların (response) birbirleriyle eşleştirilmesi, böylelikle de bir FTP doğrulama talebi geldiğinde, IDS’in gelen istek içerisindeki durum (status) kodunu bulabilmesidir. Bir defa kullanıcı başarıyla doğrulandığında, oturum doğrulanmış durumda olacak ve kullanıcı yetkisi dâhilindeki komutlar çalıştırılabilecektir. Doğrulanmamış iken bu komutların çoğu çalıştırıldığında şüpheli olarak değerlendirilecek, ancak doğrulanmış bir oturumda bu komutlar çalıştırıldığında normal bir aktivite olarak değerlendirilecektir.

4.3 Yanlış Pozitif ve Negatif Tespitler

IDS’ler sistem yöneticilerine her zaman kesin doğru bilgiler verememektedir. Sistem içerisindeki bazı alışılmadık hareketler saldırı denemesi olarak algılanabilmekte olup, aslında sistem tarafından gerçekleştirilen gerekli birer hareket olabilmektedir. Yanlış pozitif veya negatif tespitler nedeniyle IDS’lerin çok iyi bir şekilde yapılandırılmış olması gerekmektedir.

Ağ trafiğinin çok karmaşık olması nedeniyle bir IDS’nin kusursuz ve hatasız bir şekilde çalışması imkânsızdır. Hatalı sonuçlar kapsamında IDS tespitleri iki guruba ayrılmaktadır; yanlış pozif ve yanlış negatif tespitler. Yanlış pozitifler, IDS’nin saldırı olmamasına rağmen olay ve trafiğin yapısından kaynaklı olarak mevcut durumları saldırı olarak tespit etmesidir. Yanlış negatifler ise aslında saldırı girişimi olan olay ve trafiğin IDS tarafından tespit edilememesidir. Her iki durumda da sistem güvenlik yöneticilerinin sistem üzerinde kalibrasyon ve gerekli konfigürasyon değişikliklerini yapmaları gerekmektedir. Yanlış pozitif tespitlerin yüksek oranlarda da olsa gerçekleşmesi genellikle sistemin güvenliği için bir sorun teşkil etmemekte ve kabul edilebilir olarak ele alınmakta, sadece sistem güvenlik yöneticilerinin gereksiz yere meşgul olmalarına sebep olmaktadır. Bunula birlikte, tespit edilemeyen yanlış

negatifler, sistem güvenlik yöneticilerine yük getirmemekle birlikte sistemin güvenliği için büyük tehlike oluşturmaktadır.

4.4 Sistem Bileşenleri

4.4.1 Algılayıcılar (Sensörler)

Bir sistem veya ağ üzerinde bilgi toplamak amacıyla dağıtılmaktadır. Birçok kaynak üzerinden ağ paketleri de dâhil olmak üzere çeşitli bilgileri toplamakta, bunları kayıt altına almakta ve sistem çağrılarının izini sürmektedir. Girdiler toplandıktan sonra, düzenlenmekte ve bir veya daha fazla analizciye yönlendirilmektedir.

4.4.2 Analizciler

Algılayıcılardan yönlendirilen verileri toplamakta ve bir saldırı girişimi olup olmadığını saptama işlemlerini gerçekleştirmektedir. Analizcilerin oluşturduğu çıktılar saldırı raporunda bulunan kanıtları desteklemektedir. Ayrıca sistem güvenlik yöneticilerine rehberlik ederek çıktı ve rapor oluşturmaktadır.

4.4.3 Kullanıcı arayüzü

Son kullanıcılar ile sistem arasındaki bağlantıyı sağlayan kısımlardır. Arayüz üzerinden kullanıcılar sistemi kontrol ederek yapılandırabilmekte ve sistem raporları üretmektedir.

4.4.4 Bal çanağı

Dağıtımı gerçekleştirilmiş bir IDS içerisinde “bal çanağı (honeypot)” olarak adlandırılan ve saldırganlar için tuzak niteliğinde olan yapılar teşkil edilebilmektedir. Bal çanakları sistem üzerine gerçekleşen saldırılar için bir erken uyarı mekanizması olmakta, kritik sistemlerin önlerinde teşkil edilerek bir yerde yem görevi görmekte, böylelikle de saldırı kaynaklarının analizleri kapsamında veri toplanmasına olanak sağlamaktadır. Birçok IDS üreticisi araştırma amacıyla sistem içerisinde bal çanağı yerleştirmekte ve bunlardan elde edilen verileri yeni saldırı imzaları geliştirirken kullanmaktadır.

4.5 Bulut Bilişimde Karşılaşılan Saldırılar

Bulut bilişim sistemleri üzerine gerçekleştirilen saldırılar, geleneksel IT sistemlerine yönelik saldırıların tamamını kapsamakla birlikte, ağırlıklı olarak bulut üzerine gerçekleştirilen bazı saldırı türleri bulunmaktadır [45].

4.5.1 Bulut içerisinden gelen saldırılar

Yetkili bulut kullanıcıları, sistem yöneticileri veya bulut operatörleri tarafından yetkileri dışında gerçekleştirilen davranışlardır. Bu kullanıcılar, bulut kullanıcılarına ait bilgilere yetkilerini aşan bir şekilde erişerek, yeri geldiğinde bu bilgileri amaçları doğrultusunda değiştirmekte ve bulut kullanıcılarının zarar göreceği şekilde kullanmaktadır.

4.5.2 Taşma (Flooding) saldırıları

Bu saldırı çeşidinde, saldırganlar kurbanlara ağ üzerinden zombi adı verilen ele geçirilmiş bilişim kaynakları aracılığıyla çok yüksek miktarlarda paket göndermektedir. Paketler TCP, UDP, İnternet Kontrol Mesaj Protokolü (Internet Control Message Protocol – ICMP) veya bunların bileşkesi olabilmektedir. Bu saldırılar genellikle yetkisiz ağ bağlantıları üzerinden gerçekleştirilmektedir.

Bulut bilişimin doğası gereği, VM'ler üzerine gelen tüm ağ bağlantıları internet üzerinden gerçekleşmektedir. Bu nedenle, bulut kullanıcılarının zombi kaynaklar tarafından DoS veya Dağıtık Servis Dışı Bırakma (Distributed Denial of Service – DDoS) saldırılarına maruz kalmaları kaçınılmaz bir sonuçtur.

Taşma saldırıları yetkili kullanıcıların servis hazır bulunurluğunu etkilemektedir. Tek bir servis hizmeti veren bir sunucuya gerçekleştirilecek bir saldırı ile saldırgan sunulan bu servise erişimi büyük ölçüde engelleyebilmektedir. Bu tarz saldırılar DoS saldırısı olarak adlandırılmaktadır. Eğer sunucu kaynakları, taşma saldırıları sonucunda işlem yapmaya zorlanır ve sunucu üzerindeki diğer servislerin çalışmasını engeller ise bu tür saldırılara dolaylı DoS saldırısı ismi verilmektedir.

Taşma saldırılarının yüksek miktarlarda donanım kullanımına sebep olması ve bulut bilişimin de kullandığın kadar öde modeliyle ücretlendirilmesi nedeniyle, bu saldırılar sonucunda bulut kullanıcıların maliyetleri gereksiz yere artmaktadır.

4.5.3 Yetkili kullanıcı saldırıları

Yetkili bir kulanıcının hesap ve şifre bilgilerini ele geçiren bir saldırgan, sistem üzerinde sınırsız erişim yetkisine sahip olmaktadır. Arabellek taşmaları (buffer overflows), yönetici yetkisiyle çalışan işlemlere ait konsol bağlantıları oluşturmak amacıyla kullanılmaktadır. Bu tür saldırılar, bir uygulamanın statik olarak tanımlanan ara belleğine kapasitesinden daha fazla bilgi yazılmak istenmesiyle gerçekleştirilmekte ve ara belleğe sığmayan bilgiler saldırganlar tarafından ele geçirilmektedir. Yetkili bir kullanıcının hesap ve şifre bilgisini ele geçiren bir saldırgan sunucu ve dolayısıyla sunucu üzerinde çalışan tüm VM'lere erişim hakkına sahip olmaktadır.

4.5.4 Port taraması

Bir sistem üzerindeki açık, kapalı ve filtrelenen portların tespit edilmesini sağlayan bir saldırıdır. Port taraması sayesinde saldırganlar açık olan portlar yardımıyla, bir sistem üzerinde çalışan servisler ve ağ bağlantılarına ait IP adresi, ortam erişim kontrolü (Media Access Control – MAC) adresi, yönlendirici, ağ geçidi ve güvenlik duvarı kuralları gibi ayrıntılı bilgiler elde edilebilmektedir. TCP taraması, UDP taraması, SYN/FIN/ACK taraması ve Window taraması en çok kullanılan port tarama saldırılarındandır. Port taraması kendi başına kullanılan bir saldırı olmamakla birlikte, bulut bilişim kapsamında bir saldırgan port taraması sayesinde servis olarak sunulan uygulamalara ait açık portları tespit ederek, elde edilen bilgiler doğrultusunda asıl saldırıyı gerçekleştirmektedir.

4.5.5 VM veya hipervizöre gerçekleştirilen saldırılar

Hipervizörün ele geçirilmesi sonrasında, sanal ortamda çalışmakta olan tüm VM'ler üzerinde kontrol yetkisi elde edilmektedir. Sıfırıncı gün saldırıları, saldırganların VM'ler üzerine saldırarak hipervizörü veya sanal ortamda çalışan diğer VM'leri ele geçirmek amacıyla kullandıkları yöntemlerden biridir. Sistem ve yazılım geliştiriciler tarafından mevcut bir açıklığa ait yama veya güncelleme uygulanmadan önce, bu açıklığın kullanılmasıyla gerçekleştirilmektedir.

4.5.6 Arka kapı (Back door channel) saldırıları

Saldırganların bulut içerisinde bulunan bir düğümü ele geçirerek, bu düğümü DDoS saldırılarında kullanmak üzere zombi bir kaynak haline getirmelerini sağlayan pasif

bir saldırı türüdür. Sistem üzerinde bulunan truva atları ve benzer yapılar sayesinde sistem ele geçirilmekte, ele geçirilen sistemin zombi olarak kullanılmasının yanında, kullanıcıya ait bilgilere de erişilebilmektedir.

5. TEST SENARYOLARI

Bulut bilişim sistemlerinin maruz kaldığı saldırılar incelendiğinde, büyük kısmının ağ üzerinden gerçekleştirildiği görülmektedir. Sistemin içerisinden, yani sanal altyapıda kullanılan hipervizör katmanından, bulut sistem yöneticilerinin veya operatörlerinin yetkisiz erişimlerine karşı bulut kullanıcıları tarafından alınabilecek evrensel bir güvenlik mekanizması hâlihazırda bulunmamaktadır. Bununla birlikte sistem üzerindeki istenmeyen ve zararlı aktivitelere karşı HIDS'lerin, ağ üzerinden gerçekleştirilecek saldırıları tespit etmek üzere de NIDS'lerin kullanılması gerekmektedir. Önceki bölümlerde açıklanan IDS'lerin konumlandırmaları incelendiğinde, bulut bilişim sistemlerinde yazılım tabanlı olarak çalışan NIDS'lerin ağ üzerinde üç farklı şekilde konumlandırılabileceği açıkça görülmektedir [43, 44]. Bunlar sırasıyla;

1. Kullanıcılara ait herbir VM bünyesinde kendisine ağ üzerinden gelecek saldırıları tespit etmek üzere NIDS'i konumlandırmak,

2. Ağ üzerindeki tüm trafiği dinlemek üzere sistem üzerinde ayrı bir VM bünyesinde NIDS'i konumlandırmak,

3. Güvenlik duvarı ya da yönlendirici olarak kullanılmakta olan varsayılan ağ geçidi görevindeki ayrı bir VM üzerinde NIDS'i konumlandırmaktır.

Birinci senaryoda donanım kullanımının NIDS'lerin üzerinde çalıştığı OS'lerin bulunduğu VM sayısıyla doğru orantılı olarak artması, ikinci senaryoda ağ üzerindeki trafik yükünün iki katına çıkması ve son olarak ta üçüncü senaryoda ağ yükü değişmeden donanım kullanımının ikinci senaryo ile yakın oranlarda olması beklenmiştir. Her üç senaryoda da varsayılan ağ geçidi olarak kullanılacak olan VM üzerindeki ağ trafiği yükü, OSI modelindeki ağ katmanında gerçekleşeceği ve IDS'lerin genel yapsı itibariyle çoğunlukla üst katmanlarda meydana gelen olayları incelemelerinden dolayı, NIDS'lerin uygulama bazında CPU ve bellek kullanımları, bulut kullanıcılarının maliyetlerini belirleyen etkenler olarak öne çıkmaktadır. Bu senaryolardan yola çıkarak saldırı tespit işlemlerinin kullanıcılara ait müstakil

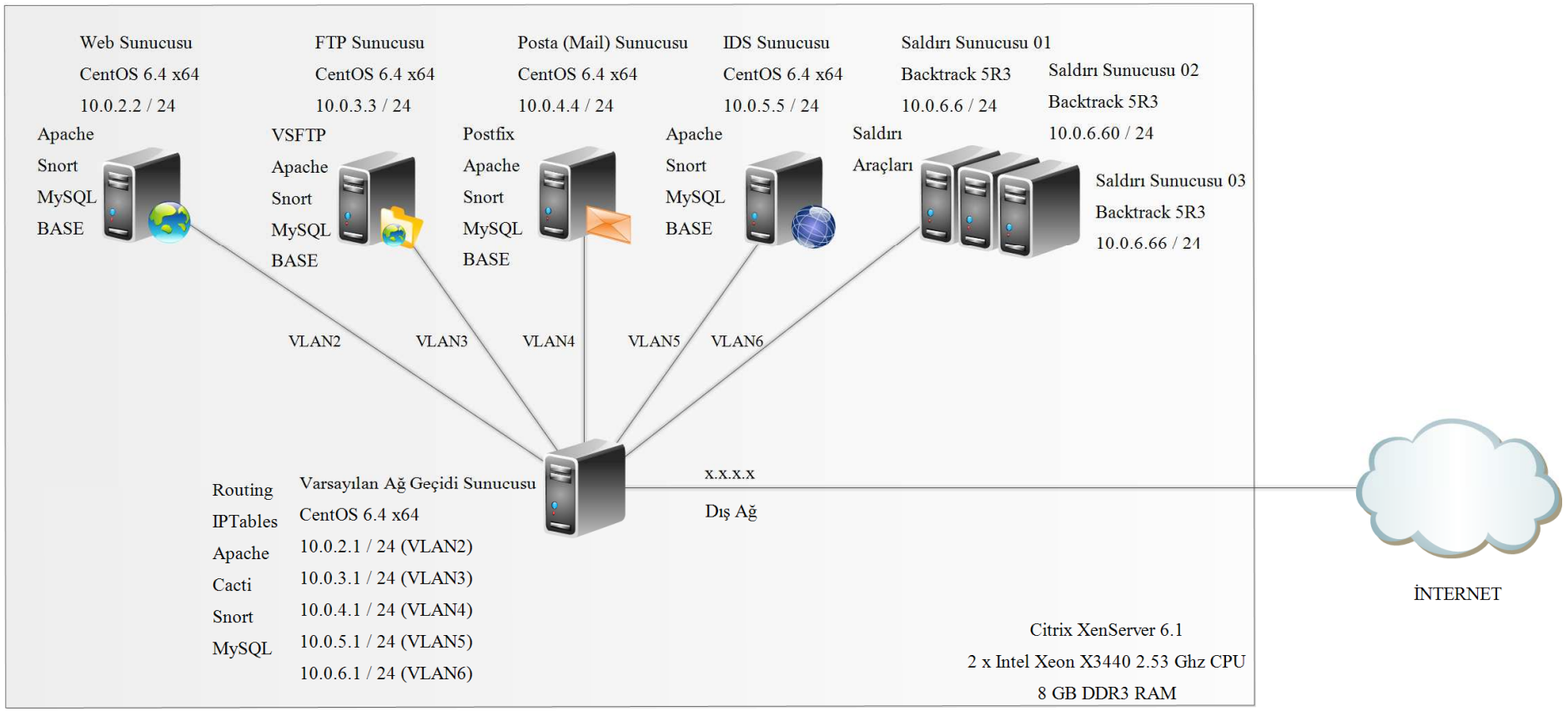
VM'lere yüklenmeyerek, en iyi donanım kullanımını sağlayacak ayrı bir VM üzerinde gerçekleştirilecek olması sonucunda, saldırı tespit işlemini gerçekleştirecek olan VM kendisine yüklenecek olan NIDS görevini diğer VM'lerin yerine vekâleten gerçekleştirecektir. Söz konusu durum yapı itibariyle ele alınacak olursa, saldırı tespit işlemlerini gerçekleştirecek olan VM vekil bir sunucu şeklinde çalışacak olup, senaryolar dâhilinde ortaya konacak en makul sisteme Bulut Bilişimde Vekil Ağ Saldırı Tespit Sistemi (Proxy Network Intrusion Detection System for Cloud Computing) adının verilmesi uygun bulunmuştur. Kurulumu gerçekleştirilen test ortamına ait topoloji Şekil 5.1'de gösterilmektedir.

Ortaya konan senaryoları gerçekleştirmek amacıyla bulut bilişimin doğası gereği maliyet etkin bir çözüm olması ve sistem üzerine istenildiği şekilde müdahale edilebilmesi maksadıyla, sanal ortamı oluşturacak platform da dâhil olmak üzere kullanılan tüm bileşenler açık kaynak kodlu (AKK) olarak seçilmiştir.

Senaryoların gerçekleştirilmesi maksadıyla oluşturulacak topoloji hyper-threading özelliği bulunan bir adet Intel Xeon X3440 2,53 Ghz CPU ve 8 GB DDR3 rastgele erişimli bellek (Random Access Memory – RAM) kapasiteli bir sunucu üzerinde inşa edilmiş olup, sanallaştırma platformu olarak Citrix XenServer 6.1.0 kullanılmıştır.

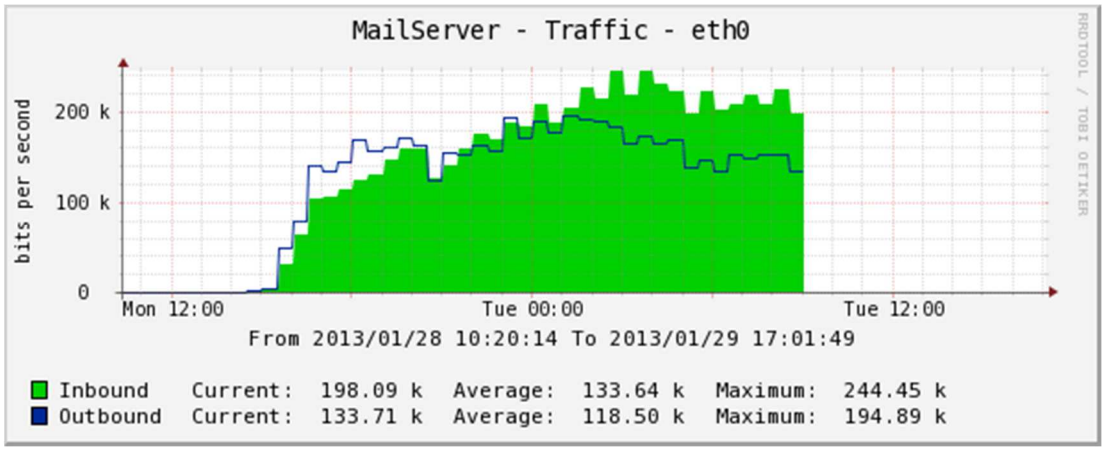
Sanal platform üzerinde iç ve dış ağ arasında bir ağ geçidi ve ağ geçidinin arkasında güvenlik gerekçesiyle birbirinden farklı sanal yerel alan ağları (Virtual Local Area Network – VLAN) üzerinde sekiz adet VM yapılandırılmıştır. Sanal platform üzerinde kullanıcıya ait VM'leri temsilen kurulumları yapılan üç adet VM'ye sırasıyla web sunucusu, FTP sunucusu ve e-posta sunucusu görevleri yüklenmiş, bir VM IDS sunucusu olarak yapılandırılmış ve saldırı benzetimlerini gerçekleştirmek amacıyla da üç adet VM'nin kurulumu gerçekleştirilmiştir. VM'lerin farklı VLAN'lar üzerinde oluşturulmasının nedeni, ele geçirilmiş olan bir VM'nin ağ üzerinden diğer VM'lere doğrudan erişebilmesini engelleyerek, ağ üzerindeki tüm trafiğin varsayılan ağ geçidi olarak yapılandırılan vekil konumundaki sunucu üzerinden gerçekleşmesini sağlamaktır.

E-posta sunucusu yapılandırılırken sehven yapılan açık röle (open relay) konfigürasyonundan dolayı e-posta sunucusu diğer açık röle yapılandırmasına sahip e-posta sunucularının gereksiz posta (spam) saldırılarında maruz kalmış ve yaklaşık dokuz saat içerisinde internet ortamına binlerce gereksiz e-posta göndermiştir.



Şekil 5.1 : Vekil NIDS test senaryolarının gerçekleştirildiği topoloji.

Bu esnada dış ağ kartına ait IP adresi spam kara listelerine (black list) düşmüştür. Bu zaman dilimi içerisinde e-posta sunucusuna ait kayıt dosyası yaklaşık olarak 2,5 GB boyutuna ulaşmış ve varsayılan ağ geçidi durumundaki sunucu, sunucunun fiziksel olarak bulunduğu veri merkezi tarafından kapatılmıştır. Söz konusu gereksiz e-posta saldırısı sırasında sunucu üzerinde meydana gelen ağ trafiğine ait grafik Şekil 5.1'de gösterilmektedir.



Şekil 5.2 : E-posta sunusunun açık röle yapılandırması ağ trafiği.

Test ortamı kurulumlarını gerçekleştirirken, saldırı benzetimlerinin gerçekleştirileceği sunucular dış ağ üzerinde konumlandırılmak üzere planlanmıştır. Ancak, e-posta sunucusunda sehven gerçekleşen yapılandırma hatasına benzer hataların oluşması durumunda, fiziksel sunucunun bulunduğu veri merkezi ve test ortamının başına gelebilecek olumsuzlukların önüne geçmek amacıyla, saldırı benzetimlerinin yapılacağı sunucuların da ayrı VLAN üzerinde yapılandırılılması planlanmıştır.

5.1 Test Ortamında Kullanılan Bileşenler

Test ortamında bulunan web sunucusu, FTP sunucusu, e-posta sunucusu, IDS sunucusu ve varsayılan ağ geçidi görevindeki sunucu üzerinde 64 bit mimariye sahip Community Enterprise Operating System (CentOS) 6.4; saldırı sunucularının üzerinde ise 64 bit mimariye sahip Backtrack 5R3 OS kullanılmıştır. Literatür taraması esnasında, özellikle IDS'ler ile ilgili çalışmaların AKK ya da en azından GPL sahibi uygulamalar kullanılarak yapıldığı görülmüş olup, bu çalışma dâhilinde de kurulan test ortamındaki OS'ler de dâhil tüm uygulamaların AKK ya da en azından GPL altında dağıtımlarının yapılıyor olmasına dikkat edilmiştir.

5.1.1 Citrix Xen Server

Sanallaştırma platformu olarak AKK ve kararlı olmasının yanında bulut altyapısını desteklemesi açısından Citrix firması tarafından dağıtımı GPL altında gerçekleştirilen Xen Server tercih edilmiştir. Bulut altyapısında kullanılan ve kaynak havuzu oluşturulmasına imkân veren Citrix Xen Server Enterprise 6.1.0 [42] sanallaştırma platformu, H2B Internet Technologies [46] PlusClouds IaaS Servis Sağlayıcı [47] tarafından sağlanmıştır.

5.1.2 CentOS

CentOS, Community Enterprise Operating System kelimelerinin kısaltmasıdır. RedHat Enterprise Linux (RHEL) [48] OS'ye ait kaynak kodlardan derlenen CentOS'un dağıtımı GPL altında gerçekleştirilmektedir [49]. Yaklaşık iki sene aralıklarla yeni sürümü, altı aylık aralıklarla da yeni versiyonunun dağıtımı yapılmaktadır. OS olarak CentOS'un tercih edilmesinin nedeni, sahip olunan tecribe ve destek açısından yeterli kaynağa sahip olunmasıdır. Test ortamı bünyesindeki sunucular üzerinde en güncel sürüm olan 6.3 sürümünün kurulumları gerçekleştirilmiş olup, 09 Mart 2013 tarihinde 6.4 sürümünün dağıtımının gerçekleşmesi sonrasında, test ortamının kurulumları yeniden yapılmış ve kullanılan OS'ler CentOS 6.4'e yükseltilmiştir.

5.1.3 IPTables

Birçok linux tabanlı OS çekirdeğinde ağ katmanındaki işlemleri gerçekleştiren netfilter modülünün uygulama katmanındaki yönetimini sağlayan yazılımdır [50]. Ağ adres ve port dönüşümleri (Network Address and Port Translation – NAPT) netfilter üzerinde, IPTables'a girilen kurallar doğrultusunda gerçekleştirilmektedir. IPTables, doğru ve kararlı kurallar girilmesi sonucunda güvenlik duvarı olarak çalışabilmektedir. Test ortamındaki en önemli kısımlardan biri olan yönlendirme ve NAPT işlemleri için IPTables kullanılmıştır. Varsayılan olarak tüm trafiği engelleyecek ve sadece kullanılacak olan ağ trafiğine izin verecek şekilde yapılandırılmıştır. Bu kurallar temel olarak, varsayılan ağ geçidi konumundaki sunucunun arkasında (yani iç ağ) bulunan sunuculara SSH konsol bağlantısı ve HTTP web erişimleri gibi temel birkaç dış bağlantı ile sunucuların kendi aralarındaki yönlendirmelerden oluşmaktadır.

5.1.4 Apache web sunucusu

Apache Software Foundation bünyesindeki gönüllü bir topluluk tarafından geliştirilen ve şu an web sunucusu olarak kullanılan en popüler yazılımlardan biridir [51]. İnternet araştırmaları üzerine çalışan Netcraft şirketi tarafından Mayıs 2013 ayında yaklaşık olarak 170 milyon internet sitesi üzerinde yapılan bir araştırmaya göre, değerlendirilen internet sitelerinin %50'si Apache Web Sunucusu kullanmaktadır [52].

5.1.5 Very secure FTP daemon (VSFTPD)

Çok güvenli FTP arka plan programı olarak adlandırılan VSFTPD, hemen hemen tüm linux tabanlı OS'lerin yazılım depolarında bulunmakta olup, kurulumu göreceli olarak basit olması sebebiyle tercih edilmiştir [53].

5.1.6 Postfix

Kamusal lisans altında dağıtımı yapılan ve linux tabanlı sistemlerde en fazla tercih edilen Posta Aktarım Ajanlarından (Mail Transport Agent – MTA) birisidir [54].

5.1.7 MySQL veritabanı

İlişkisel veri tabanı yönetim sistemleri arasında en fazla tercih edilen, çoklu kullanıcı desteği bulunan, hızlı ve tutarlı bir seçenek olarak ön plana çıkmasının yanında, sunucu olarak hizmet vermektedir [55].

5.1.8 Cacti

Bir ağ üzerinde bulunan aktif cihazlara (sunucu, yönlendirici, güvenlik duvarı, yönetilebilir anahtarlama cihazı vb.) ait durum ve kullanım bilgilerine ait ortalamaları Basit Ağ Yönetim Protokolü (Simple Network Management Protocol – SNMP) üzerinden alarak, bu bilgileri Round Robin Veritabanı formatında zaman serisi olarak saklamakta ve web arayüzü sayesinde oluşturulan şablonlarla donanım kullanım grafiklerin izlenebilmesine imkân tanımaktadır [56].

5.1.9 Snort

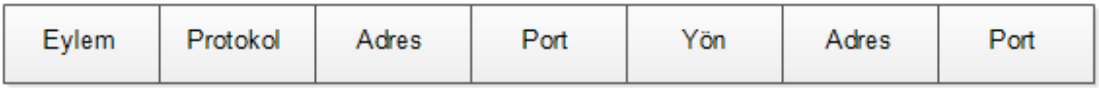
İmza, anormallik ve durum bilgisi tabanlı çalışabilen, AKK ve GPL altında dağıtımı yapılan bir IDPS'tir [57, 58]. Oluşturulan test ortamının en önemli bileşeni pozisyonundadır. Ticari bir ürün olmamasının avantajıyla birlikte, bu alanda kullanılan diğer IDS ve IDPS'ler ile karşılaştırıldığında, göreceli bir şekilde kontrol ve yönetim

olarak kullanıcı dostudur. Bu özelliklerinden dolayı test ortamında kullanılacak IDS olarak Snort belirlenmiştir. Oluşturulan test ortamının rutin bir ağ trafiğine sahip olmaması nedeniyle, ağın normalitesinin tespit edilerek anormallik durumlarının analizlerinde kullanılacak profillerin oluşturulmasının gerçekçi sonuçlar elde edilmesine engel olacağı değerlendirilmiştir. Bu nedenle test ortamında bulunan Snort yapılandırmalarının tamamı birbiriyle bire bir aynı olan kural tabanlı tespit yapılmasına yönelik hazırlanmıştır. Snort kuralları, uygulama katmanında tanımlanmaktadır. Snort'un saldırı tespit kurallarının güncellenmesi amacıyla **Pulledpork** [57] isimli üçüncü parti bir yazılım kullanılmış ve 19 Nisan 2013 tarihinde Snort'u geliştiren Sourcefire tarafından yayınlanan güncel saldırı tespit kuralları sisteme yüklenmiş, standart konfigürasyon yapısını bozmamak adına Pulledpork devre dışı bırakılmış ve bu tarihten sonra kural güncellemesi yapması engellenmiştir.

Snort kural yapısı incelendiğinde, Kural Başlığı (Rule Header) ve Kural Seçenekleri (Rule Options) kısımlarından oluştuğu görülmektedir [58]. Şekil 5.3'te Snort kurallarının temel yapısı gösterilmiştir.

Şekil 5.3 : Snort kurallarının temel yapısı.

Snort kural başlığı, ağ paketi ile bir kuralın eşleşmesi durumunda gerçekleştirilecek eylem bilgisini içermektedir. Kural seçenekleri ise genellikle bir uyarı mesajı ve paketin hangi kısmıyla ilgili uyarı üretileceğini tanımlamaktadır. Bir kural içerisinde bir veya daha fazla saldırı tanımlaması yapılabilmekte olup, doğru tespitler yaparak yanlış negatif veya yanlış pozitif tespitler oluşmaması amacıyla bir kural içerisinde tek bir tanımlama yapılması genel bir yaklaşım olarak kullanılmaktadır. Şekil 5.4'te Snort kural başlığının genel yapısı gösterilmiştir.



Şekil 5.4 : Snort kural başlığının genel yapısı.

Eylem; kuralın veri paketini analizi sonrasında gerçekleştireceği durumu tanımlamaktadır. Genellikle uyarı oluşturma, kayıt altına alma veya diğer bir kuralı devreye sokma şeklinde kullanılmaktadır.

1. *Pass;* Snort'un paketi yok sayarak hızının artırılması ve daha performanslı çalışmasının istendiği durumlarda bazı paketlerin analize tabi tutulmaması maksadıyla kullanılmaktadır.

2. *Log;* paketi kayıt altına almak amacıyla kullanılmaktadır.

3. *Alert;* eşleşen durum sonrasında, pakete ait bir uyarı oluşturmaktadır. Uyarı mekanizması bir dosya içerisine veya konsola mesaj yazılması şeklinde gerçekleşmektedir.

4. *Activate;* eşleşilen durumun ardından başka bir kuralı çalıştırmaktadır.

5. *Dynamic;* Activate tarafından çağrılan kuralların başında bulunan tanımlayıcıdır.

6. *Kullanıcı tanımlı eylemler;* kullanıcılar tarafından farklı amaçlarla oluşturulan kuralların başında tanımlayıcı olarak kullanılmaktadır.

Adres; bir Snort kuralının içerisinde iki adet adres bölümü bulunmaktadır. Adres bölümleri paketin hangi kaynaktan geldiğini ve hangi hedefe gönderildiği bilgilerini içermektedir. Adres bölümleri tek bir IP adresinden oluşabildiği gibi, belli bir adres grubunu da ifade edebilmektedir. Adres grubundan sonra kullanılan "/" işaretinden sonra gelen sayı alt ağ maskesinde bulunan bit sayısını ifade etmektedir.

Port; paketlerin kaynak ve hedef port numaralarını belirtmektedir. Örneğin; 23 numaralı port numarasının uygulanacağı paketler bir Telnet sunucusuna aittir. Tüm paketlere uygulanması istenen bir kural içerisinde port kısmı "*any*" şeklinde ifade edilmektedir. Belirtilen port numaraları TCP ve UDP paketler için anlam ifade etmekte olup, IP veya ICMP türü paketlere uygulanacak kurallarda kullanılmasının bir anlamı bulunmamaktadır.

Yön; bir kural içerisindeki adres ve port numaralarının uygulanacağı doğrultuyu ifade etmektedir. "->", "<-" ve "<>" işaretleriyle gösterilmektedir.

- "->" işareti, kuralın sol tarafında bulunan adres ve port numaralarının paketin kaynak kısmına, sağ tarafında bulunan adres ve port numaralarının paketin hedef kısmına ait olduğunu ifade etmektedir.

- "<-" işareti, kuralın sağ tarafında bulunan adres ve port numaralarının paketin kaynak kısmına, sol tarafında bulunan adres ve port numaralarının paketin hedef kısmına ait olduğunu ifade etmektedir.

- “<>” işareti, kuralın tanımlanan adres ve port numaraları arasında iki yönlü olarak uygulanacağını ifade etmektedir.

Kural Seçenekleri; kural başlığından sonra parantezler içerisinde tanımlanmaktadır. Kural seçenekleri içerisinde bir veya daha fazla seçenek aynı anda tanımlanabilmekte olup, çoklu seçenek tanımlaması sırasında seçenekler arasında *“AND”* mantıksal ifadesi kullanılmaktadır. Kural seçenekleri içerisindeki ifadeler, başlık kısmındaki kriterlerin tamamının sağlanması durumunda başlatılmaktadır. Tüm seçenekler bir anahtar kelime ve bu anahtar kelimeye ait bir argümanla ifade edilmekte olup, seçenekler ile argümanlar noktalı virgül *(;)* ile ayrılmaktadır. Kural seçenekleri içerisinde çok sayıda anahtar kelime kullanılmakta olup, yaygın olarak kullanılan bazı anahtar kelimeler aşağıda açıklanmıştır.

- *ACK;* TCP başlık kısmı 32 bit uzunluğunda bir Alındı Sayısı alanına ahiptir. Bu alan bir sonraki beklenen TCP paketinin sıra numarasını ifade etmektedir. Bu seçenek sadece TCP paketlerindeki ACK bayrağının bulunması durumunda önemli olmaktadır.

- *classtype;* Snort kuralları sınıflara ve öncelik sıralarına göre gruplara ayrılabilmektedir. Bu sınıf ve öncelikler *classification.conf* dosyası içerisinde tanımlanmaktadır. Örnek bir sınıflandırma tanımı *config classification: sınıf adı, tanımlama, öncelik sırası* şeklinde yapılmaktadır.

- *content;* Snort’un önemli faydalarından biri de, paket içerisinde bulunan bir veriyi bulabilmesidir. Veri Bilgi Değişimi için Amerikan Kod Standardı (American Standard Code for Information Interchange – ASCII) veya onaltılık sayı düzeninin ikilik düzendeki karşılıkları olabilmektedir. Virüsler ve saldırganlara ait paketler içerisinde bulunan veriler doğrultusunda imza ve içerik anahtar kelimeleri oluşturulmaktadır.

- *content-list;* bir dosya ismiyle birlikte kullanılmakta olup, belirtilen dosya içerisinde bulunan karakter katarı (string) halindeki ifadeleri paket içerisinde aramaktadır. Dosya içerisindeki her bir string ifade ayrı bir satırda tanımlanmaktadır.

- *dsize;* paket içerisinde tanımlanan uzunluktaki bölümleri aramaktadır. Birçok saldırı önbellek taşma açıklıklarını kullanmaktadır. Bu anahtar kelimenin kullanılmasıyla bir paketin belirtilen boyuttan büyük, küçük ya da aynı boyutta olduğu tespit edilebilmektedir.

- *msg;* kural eşleşmeleri sonrasında oluşturulan kayıt ve uyarılara string bir ifade eklemek amacıyla kullanılmaktadır. Eklenecek olan string ifade iki adet çift tırnak işaretinin arasında belirtilmektedir. Bu anahtar kelime neredeyse tüm kural tanımlamaları içerisinde kullanılmaktadır.

Aşağıda örnek Snort kuralları ve açıklamaları bulunmaktadır.

- Örnek 1;

  alert tcp any any -> 192.168.1.0/24 111 (msg: "111 numaralı porta erişim";)

  Herhangi bir IP adresi ve port kaynaklı TCP paketlerinin 192.168.1.0 (alt ağ maskesi 24 bit uzunluğunda yani 255.255.255.0 olan) ağı üzerinde port numarası 111 olan herhangi bir hedefe sahip paket ile eşleşme olduğunda *"111 numaralı porta erişim"* içeriğinde uyarı üretilmektedir.

- Örnek 2;

  alert tcp any any -> 192.168.1.0/24 !6000:6010 (msg: "port ihlali";)

  Herhangi bir IP adresi ve port kaynaklı TCP paketlerinin 192.168.1.0 (alt ağ maskesi 24 bit uzunluğunda yani 255.255.255.0 olan) ağı üzerinde port numarası 6000 ve 6010 aralığında olmayan herhangi bir hedefe sahip paket ile eşleşme olduğunda *"port ihlali"* içeriğinde uyarı üretilmektedir.

- Örnek 3;

  alert icmp ![192.168.2.0/24,192.168.8.0/24] any -> any any (msg: "Ping TTL=100"; ttl: 100;)

  192.168.2.0 (alt ağ maskesi 24 bit uzunluğunda yani 255.255.255.0 olan) ve 192.168.8.0 (alt ağ maskesi 24 bit uzunluğunda yani 255.255.255.0 olan) ağ aralığı dışından gelen ve sonlandırılmadan önceki hop sayısı 100 (time to live – TTL) olan ICMP paketleriyle eşleşme olduğunda *"Ping TTL=100"* içeriğinde uyarı üretilmektedir.

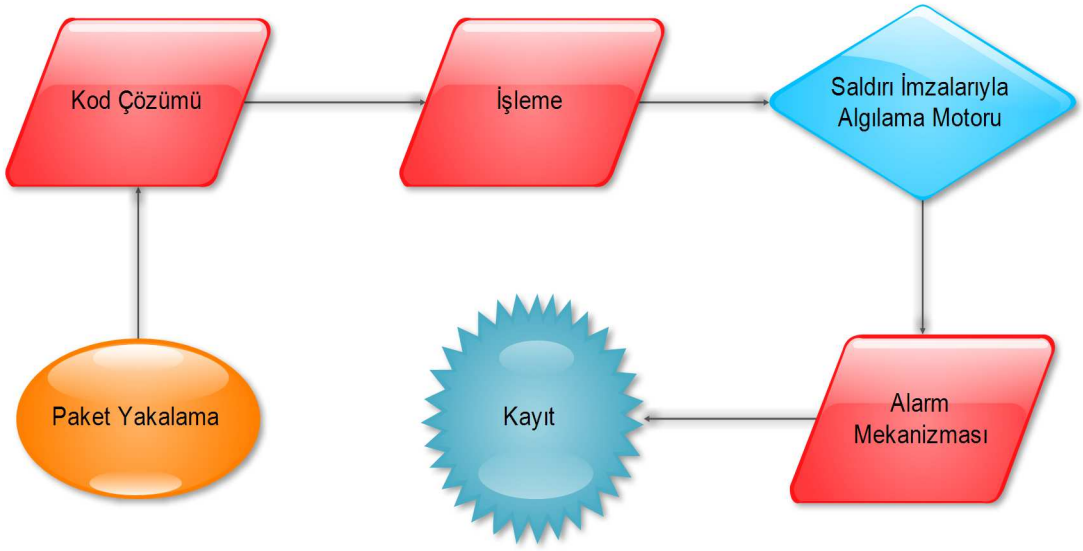
- Örnek 4;

  alert tcp 192.168.1.0/24 23 -> any any (content: "confidential"; msg: "Detected confidential";)

  192.168.1.0 (alt ağ maskesi 24 bit uzunluğunda yani 255.255.255.0 olan) ağ kaynaklı herhangi bir IP adresi ve 23 port numaralı kaynağa sahip herhangi bir hedefe

gönderilen TCP paketleriyle eşleşme olduğunda *"Detected confidential"* içeriğinde uyarı üretilmektedir.

Snort farklı katmanlarda tespitlerde bulunabilmek için yakalanan paketleri parçalara bölmektedir. Tespit motoru paket bilgilerini saldırı imza veri tabanı aracılığıyla analiz etmekte ve saldırı izi tespit edilirse uyarı üretmektedir. Üretilen uyarılar **Barnyard** [57] vb. üçüncü parti yazılımlarla tanımlanan veritabanına (MySQL vb.) aktarılmakta (parsing) ve **Basit Analiz ve Güvenlik Motoru (Basic Analysis and Security Engine – BASE)** [59] vb. uygulamalarla web tabanlı olarak görüntülenebilmesi sağlanarak sistem güvenlik yöneticileri tarafından daha kolay izleme ve analiz yapılmasına imkân verilmektedir. Snort'un temel bileşenleri Şekil 5.5'te gösterilmiştir.



Şekil 5.5 : Temel Snort bileşenleri.

5.1.10 Backtrack

Sızma ve güvenlik uzmanları tarafından yapılan testlerde kullanılan bir OS'dir [60]. Backtrack GPL altında dağıtımı yapılan, Debian [61] çekirdeğine sahip olan, ağ tarama ve yakalama, web sitesi istismarı, kablosuz ağ güvenlik açıkları ve tersine mühendislik gibi birçok sızma ve güvenlik aracını bünyesinde bulundurmaktadır. Söz konusu özelliklerinden dolayı saldırı simülasyonlarını yapmak üzere tercih edilmiştir. 13 Mart 2013 tarihi itibariyle Backtrack projesi sonlandırılmış olup, Kali Linux adı altında yeniden yapılandırılan dağıtım Backtrack OS'nin yerini almıştır [62].

Mevcut test senaryolarının gerçekleştirilmesi amacıyla Bactrack OS'nin 64 bit mimariye sahip en güncel sürümü olan Bactrack 5R3'ün kurulumu gerçekleştirilmiştir.

Backtrack 5R3 bünyesinde **hping** [63] ve **nping** [64] gibi paket üreteçlerini (packet generator) bulundurmasına karşın söz konusu araçların belirli bir zaman dilimi içerisinde ya da belirli bir sayıda paket trafiği oluşturmaması nedeniyle, parametre olarak girilen miktarda paket üretebilen **IP Stack Integrity Checker (ISIC)** isimli aracın kurulumu gerçekleştirilmiş ve saldırı simülasyonları ISIC aracılığıyla yapılmıştır. Saldırı simülasyonları gerçekleştirilirken ISIC kurulumunun özellikle Backtrack OS'nin üzerinde gerçekleştirilerek farklı bir OS'nin tercih edilmemesinin nedeni, ağ paketlerinin oluşturulmasını sağlayan **libnet** [65] kütüphanesinin varsayılan olarak Backtrack üzerinde bulunmasıdır. Örnek ISIC komutları ve açıklamaları aşağıda bulunmaktadır.

- Aşağıdaki ISIC komutu ile kaynak adresi *192.168.76.123* ve hedef adresi *10.72.83.19* olan 100,000 adet ICMP paketi oluşturulmaktadır.

icmpsic -p 100000 -s 192.168.76.123 -d 10.72.83.19

- Aşağıdaki ISIC komutu ile kaynak adresi *172.22.223.12* ve hedef adresi *192.168.87.11* olan 1,000,000 adet TCP paketi oluşturulmaktadır.

tcpsic -p 1000000 -s 172.22.223.12 -d 192.168.87.11

- Aşağıdaki ISIC komutu ile kaynak adresi *10.15.23.43* ve hedef adresi *10.10.0.17* olan 10,000,000 adet UDP paketi oluşturulmaktadır.

udpsic -p 10000000 -s 10.15.23.43 -d 10.10.0.17

5.2 Test Senaryolarının Uygulanması

Test ortamının kurulumu Citrix Xen Server Enterprise 6.1.0 üzerinde gerçekleştirilmiştir. Fiziksel sunucu ve sanallaştırma platformu H2B Internet Technologies PlusClouds IaaS Servis Sağlayıcısı tarafından sağlanmıştır. Servis sağlayıcı tarafından kurulumu yapılan Xen Server Enterprise 6.1.0 üzerinde gerçekleştirilen işlemler Xen Center yazılımı ve Güvenli Kabuk (Secure Shell – SSH) konsol bağlantısı aracılığıyla internet üzerinden uzak bağlantıyla gerçekleştirilmiştir. PlusClouds tarafından oluşturulan kalıp kütüphanesi sayesinde CentOS ve Backtrack OS kurulumları gerçekleştirilmiştir.

Test ortamında kurulumları yapılan ve IDS tarafından gerçekleştirilen tespitlerin kullanıcı dostu bir şekilde izlenmesi ve incelenmesini sağlayan Barnyard ile web

arayüzüne sahip BASE yazılımları, IDS'in performansını etkilememesi amacyla saldırı simülasyonları esnasında devre dışı bırakılmış ve yalnızca Snort yazılımına ait işlemlerin sistemler üzerindeki donanım kullanımlarına ait sonuçlar karşılaştırılmıştır.

Test senaryolarının gerçeklenmesi öncesinde ön görülen donanım kullanımları Çizelge 5.1'de gösterilmiştir.

Çizelge 5.1 : Saldırı simülasyonları esnasında beklenen donanım kullanımları.

Senaryo	Donanım Kullanımı		
	CPU	Bellek	Ağ Trafiği
Senaryo 01	< Senaryo 2, > Senaryo 3	< Senaryo 2, > Senaryo 3	x
Senaryo 02	Muhtemelen en verimsiz	Muhtemelen en verimsiz	~2x
Senaryo 03	Muhtemelen en verimli	Muhtemelen en verimli	x

5.2.1 IDS'nin kullanıcı VM'leri üzerinde kurulması (Senaryo 01)

Kullanıcılara ait herbir VM üzerinde kendisine ağ üzerinden gelecek saldırıları tespit etmek amacıyla NIDS'in konumlandırılması senaryosunda, test ortamına ait topoloji üzerinde aktif olarak çalışan VM'lere ait yapılandırma ayarları ve VM'ler üzerinde çalışan servisler ile uygulamalar aşağıda açıklanmıştır.

1. Varsayılan ağ geçidi konumundaki VM
    - *VM üzerinde çalışan OS;* CentOS 6.4 x64
    - *Ağ konfigürasyonu*

      eth0; 213.128.88.10 (dış bacak - internet)

      *eth2;*10.10.2.1 / 24 (VLAN2)

      eth3; 10.10.3.1 / 24 (VLAN3)

      eth4; 10.10.4.1 / 24 (VLAN4)

      eth6; 10.10.6.1 / 24 (VLAN6)

- *IPTables;* iç ağ içerisindeki VM'lere SSH üzerinden konsol bağlantısı kurulabilmesi için NAPT ve VM'ler arasında yönlendirme yapmak üzere konfigüre edilmiştir.

- *Apache Web Sunucusu;* web tabanlı bir uygulama olan Cacti üzerinden VM'lere ait donanım kullanım durumlarını izlemek amacıyla konfigüre edilmiştir.

- *MySQL Veri Tabanı Sunucusu;* web tabanlı bir uygulama olan Cacti'nin VM'lere ait donanım kullanım durumlarını izleyebilmesi amacıyla konfigüre edilmiştir.

- *Cacti;* VM'lere ait donanım kullanım durumlarını web tabanlı olarak izlemek amacıyla konfigüre edilmiştir.

2. Web Sunucusu

- *VM üzerinde çalışan OS;* CentOS 6.4 x64

- *Ağ konfigürasyonu*

  eth0; 10.10.2.2 / 24 (VLAN2)

- *Apache Web Sunucusu;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

- *Snort;* saldırı simülasyonları yapıldığı sırada oluşturulan paketlerin analiz edilmesi amacıyla konfigüre edilmiş olup, sistemi yorarak daha gerçekçi sonuçlar elde etmek amacıyla, güncellenen saldırı imzalarının tamamı (yaklaşık 12,000 adet) Snort üzerinde aktif edilmiştir.

3. FTP Sunucusu

- *VM üzerinde çalışan OS;* CentOS 6.4 x64

- *Ağ konfigürasyonu*

  eth0; 10.10.3.3 / 24 (VLAN3)

- *VSFTPD;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

- *Snort;* saldırı simülasyonları yapıldığı sırada oluşturulan paketlerin analiz edilmesi amacıyla konfigüre edilmiş olup, sistemi yorarak daha gerçekçi sonuçlar elde etmek amacıyla, güncellenen saldırı imzalarının tamamı (yaklaşık 12,000 adet) Snort üzerinde aktif edilmiştir.

4. Mail Sunucusu

- *VM üzerinde çalışan OS;* CentOS 6.4 x64
- *Ağ konfigürasyonu*

  eth0; 10.10.4.4 / 24 (VLAN4)

- *Postfix;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

- *Apache Web Sunucusu;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

- *MySQL Veri Tabanı Sunucusu;* Postfix tarafından e-posta servisinin çalıştırılması amacıyla konfigüre edilmiştir.

- *Snort;* saldırı simülasyonları yapıldığı sırada oluşturulan paketlerin analiz edilmesi amacıyla konfigüre edilmiş olup, sistemi yorarak daha gerçekçi sonuçlar elde etmek amacıyla, güncellenen saldırı imzalarının tamamı (yaklaşık 12,000 adet) Snort üzerinde aktif edilmiştir.

5. Saldırı Sunucuları

- *VM'ler üzerinde çalışan OS;* Backtrack 5R3 x64
- *Ağ konfigürasyonu*

  Saldırı Sunucusu 01 eth0; 10.10.6.6 / 24 (VLAN6)

  Saldırı Sunucusu 02 eth0; 10.10.6.60 / 24 (VLAN6)

  Saldırı Sunucusu 03 eth0; 10.10.6.66 / 24 (VLAN6)

- ISIC; saldırı simülasyonları sırasında TCP, UDP ve ICMP paketlerinin üretilmesi amacıyla kullanılmıştır.

Saldırı simülasyonları sırasında kullanıcı VM'si pozisyonunda bulunan Web Sunucusu, FTP Sunucusu ve Mail Sunucusu'na aynı anda sırasıyla ICMP Flooding, TCP Flooding ve UDP Flooding saldırıları gerçekleştirilmiştir. Saldırı simülasyonları esnasında VM'lere ait CPU ve bellek kullanım oranları Altıncı Bölüm'de anlatılmış olup, senaryonun uygulandığı ağ topolojisi Şekil 5.6'da gösterilmiştir.

5.2.2 IDS'nin ayrı bir VM üzerinde kurulması (Senaryo 02)

Varsayılan ağ geçidi konumundaki VM'nin üzerinden geçen tüm ağ trafiğini dinlemek amacıyla NIDS'in konumlandırılması senaryosunda, test ortamı üzerinde aktif olan VM'lere ait yapılandırma ayarları ve VM'ler üzerinde çalışan servisler ile uygulamalar aşağıda açıklanmıştır. Varsayılan ağ geçidi konumundaki VM üzerinden geçen tüm trafiğin kopyalanarak IDS Sunucusuna gönderilmesi amacıyla IPTables üzerinde gerekli konfigürasyonlar yapılmak istendiğinde CentOS 6.4'ün çekirdeğinde bulunan netfilter üzerinde güvenlik gerekçesiyle yapılan kısıtlamalardan dolayı tüm trafik tek bir hedefe yönlendirilememiş ve ağ trafiğinin kopyalama işleminin gerçekleştirildiği kabul edilmiştir. Üretilen tüm paketlerde IDS Sunucusu hedef olarak gösterilmiştir.

1. Varsayılan ağ geçidi konumundaki VM

    - *VM üzerinde çalışan OS;* CentOS 6.4 x64
    - *Ağ konfigürasyonu*

      eth0; 213.128.88.10 (dış bacak - internet)

      *eth2;*10.10.2.1 / 24 (VLAN2)

      eth3; 10.10.3.1 / 24 (VLAN3)

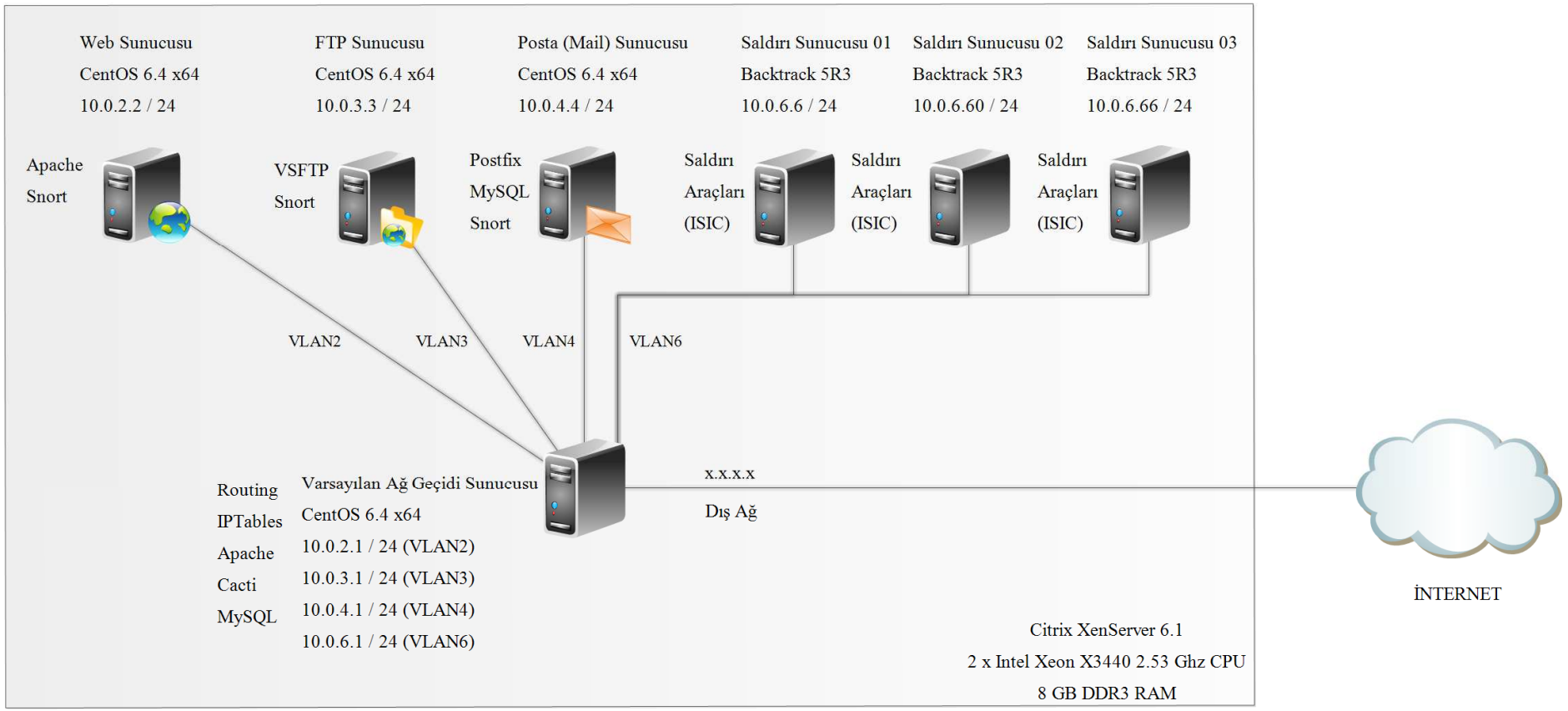
      eth4; 10.10.4.1 / 24 (VLAN4)

      eth5; 10.10.5.1 / 24 (VLAN5)

      eth6; 10.10.6.1 / 24 (VLAN6)

    - *IPTables;* iç ağ içerisindeki VM'lere SSH üzerinden konsol bağlantısı kurulabilmesi için NAPT ve VM'ler arasında yönlendirme yapmak üzere konfigüre edilmiştir.

    - *Apache Web Sunucusu;* web tabanlı bir uygulama olan Cacti üzerinden VM'lere ait donanım kullanım durumlarını izlemek amacıyla konfigüre edilmiştir.



Şekil 5.6 : IDS'nin kullanıcı VM'leri üzerinde kurulması (Senaryo 01).

- *MySQL Veri Tabanı Sunucusu;* web tabanlı bir uygulama olan Cacti'nin VM'lere ait donanım kullanım durumlarını izleyebilmesi amacıyla konfigüre edilmiştir.

- *Cacti;* VM'lere ait donanım kullanım durumlarını web tabanlı olarak izlemek amacıyla konfigüre edilmiştir.

2. Web Sunucusu (NIDS'in IDS Sunucusu üzerinde çalışmasından dolayı Web Sunucusu üzerinde çalışan servis ve uygulamalar önemsizdir)

- *VM üzerinde çalışan OS;* CentOS 6.4 x64

- *Ağ konfigürasyonu*

  eth0; 10.10.2.2 / 24 (VLAN2)

- *Apache Web Sunucusu;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

3. FTP Sunucusu (NIDS'in IDS Sunucusu üzerinde çalışmasından dolayı FTP Sunucusu üzerinde çalışan servis ve uygulamalar önemsizdir)

- *VM üzerinde çalışan OS;* CentOS 6.4 x64

- *Ağ konfigürasyonu*

  eth0; 10.10.3.3 / 24 (VLAN3)

- *VSFTPD;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

4. Mail Sunucusu (NIDS'in IDS Sunucusu üzerinde çalışmasından dolayı Mail Sunucusu üzerinde çalışan servis ve uygulamalar önemsizdir)

- *VM üzerinde çalışan OS;* CentOS 6.4 x64

- *Ağ konfigürasyonu*

  eth0; 10.10.4.4 / 24 (VLAN4)

- *Postfix;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

- *Apache Web Sunucusu;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

- *MySQL Veri Tabanı Sunucusu;* Postfix tarafından e-posta servisinin çalıştırılması amacıyla konfigüre edilmiştir.

5. IDS Sunucusu

   - *VM üzerinde çalışan OS;* CentOS 6.4 x64

   - *Ağ konfigürasyonu*

     eth0; 10.10.5.5 / 24 (VLAN5)

   - *Snort;* saldırı simülasyonları yapıldığı sırada oluşturulan paketlerin analiz edilmesi amacıyla konfigüre edilmiş olup, sistemi yorarak daha gerçekçi sonuçlar elde etmek amacıyla, güncellenen saldırı imzalarının tamamı (yaklaşık 12,000 adet) Snort üzerinde aktif edilmiştir.

6. Saldırı Sunucuları

   - *VM'ler üzerinde çalışan OS;* Backtrack 5R3 x64

   - *Ağ konfigürasyonu*

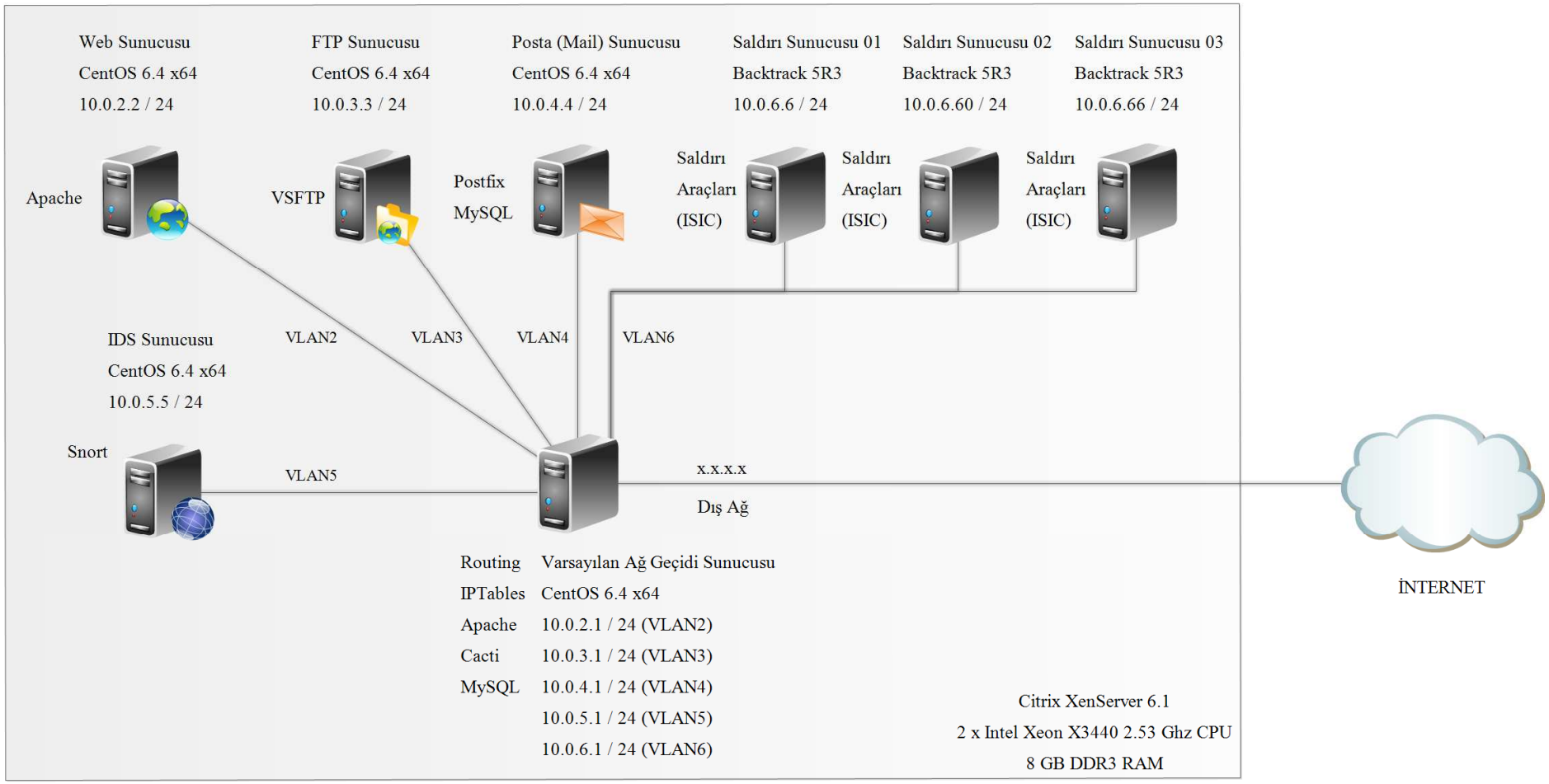
     Saldırı Sunucusu 01 eth0; 10.10.6.6 / 24 (VLAN6)

     Saldırı Sunucusu 02 eth0; 10.10.6.60 / 24 (VLAN6)

     Saldırı Sunucusu 03 eth0; 10.10.6.60 / 24 (VLAN6)

   - *ISIC;* saldırı simülasyonları sırasında TCP, UDP ve ICMP paketleri üretilmesi amacıyla kullanılmıştır.

Varsayılan ağ geçidi konumundaki VM üzerinden geçen tüm trafik, yukarıda belirtilen güvenlik gerekçesi nedeniyle netfilter üzerinde yapılan kısıtlamadan dolayı tek bir hedefe yönlendirilememiş ve ağ trafiğinin kopyalama işleminin ve saldırı simülasyonları sırasında kullanıcı VM'si pozisyonunda bulunan Web Sunucusu, FTP Sunucusu ve Mail Sunucusu'na aynı anda sırasıyla ICMP Flooding, TCP Flooding ve UDP Flooding saldırılarının gerçekleştirildiği varsayılmıştır. Saldırı simülasyonları esnasında IDS Sunucusuna ait CPU ve bellek kullanım oranları Altıncı Bölüm'de anlatılmış olup, senaryonun uygulandığı ağ topolojisi Şekil 5.7'de gösterilmiştir.



Şekil 5.7 : IDS'nin ayrı bir VM üzerinde kurulması (Senaryo 02).

5.2.3 IDS'nin varsayılan ağ geçidinde kurulması (Senaryo 03)

Varsayılan ağ geçidi konumundaki VM'in üzerinde NIDS'in konumlandırılması senaryosunda, test ortamı üzerinde aktif olan VM'lere ait yapılandırma ayarları ve VM'ler üzerinde çalışan servisler ile uygulamalar aşağıda açıklanmıştır. Varsayılan ağ geçidi konumundaki VM üzerinden geçen tüm trafik bu VM üzerinde konfigüre edilmiş olan NIDS tarafından analiz edilmektedir.

1. Varsayılan ağ geçidi konumundaki VM

- *VM üzerinde çalışan OS;* CentOS 6.4 x64
- *Ağ konfigürasyonu*

  eth0; 213.128.88.10 (dış bacak - internet)

  *eth2;*10.10.2.1 / 24 (VLAN2)

  eth3; 10.10.3.1 / 24 (VLAN3)

  eth4; 10.10.4.1 / 24 (VLAN4)

  eth6; 10.10.6.1 / 24 (VLAN6)

- *IPTables;* iç ağ içerisindeki VM'lere SSH üzerinden konsol bağlantısı kurulabilmesi için NAPT ve VM'ler arasında yönlendirme yapmak üzere konfigüre edilmiştir.

- *Apache Web Sunucusu;* web tabanlı bir uygulama olan Cacti üzerinden VM'lere ait donanım kullanım durumlarını izlemek amacıyla konfigüre edilmiştir.

- *MySQL Veri Tabanı Sunucusu;* web tabanlı bir uygulama olan Cacti'nin VM'lere ait donanım kullanım durumlarını izleyebilmesi amacıyla konfigüre edilmiştir.

- *Cacti;* VM'lere ait donanım kullanım durumlarını web tabanlı olarak izlemek amacıyla konfigüre edilmiştir.

2. Web Sunucusu (NIDS'in IDS Sunucusu üzerinde çalışmasından dolayı Web Sunucusu üzerinde çalışan servis ve uygulamalar önemsizdir)

- *VM üzerinde çalışan OS;* CentOS 6.4 x64

- *Ağ konfigürasyonu*

  eth0; 10.10.2.2 / 24 (VLAN2)

- *Apache Web Sunucusu;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

3. FTP Sunucusu (NIDS'in IDS Sunucusu üzerinde çalışmasından dolayı FTP Sunucusu üzerinde çalışan servis ve uygulamalar önemsizdir)

- *VM üzerinde çalışan OS;* CentOS 6.4 x64
- *Ağ konfigürasyonu*

  eth0; 10.10.3.3 / 24 (VLAN3)

- *VSFTPD;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.

4. Mail Sunucusu (NIDS'in IDS Sunucusu üzerinde çalışmasından dolayı Mail Sunucusu üzerinde çalışan servis ve uygulamalar önemsizdir)

- *VM üzerinde çalışan OS;* CentOS 6.4 x64
- *Ağ konfigürasyonu*

  eth0; 10.10.4.4 / 24 (VLAN4)

- *Postfix;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.
- *Apache Web Sunucusu;* saldırı simülasyonları yapıldığı sırada Snort tarafından gerçekleştirilecek işlemlere ilave olarak sistemi yorması amacıyla konfigüre edilmiştir.
- *MySQL Veri Tabanı Sunucusu;* Postfix tarafından e-posta servisinin çalıştırılması amacıyla konfigüre edilmiştir.

5. Saldırı Sunucuları

- *VM'ler üzerinde çalışan OS;* Backtrack 5R3 x64

- *Ağ konfigürasyonu*

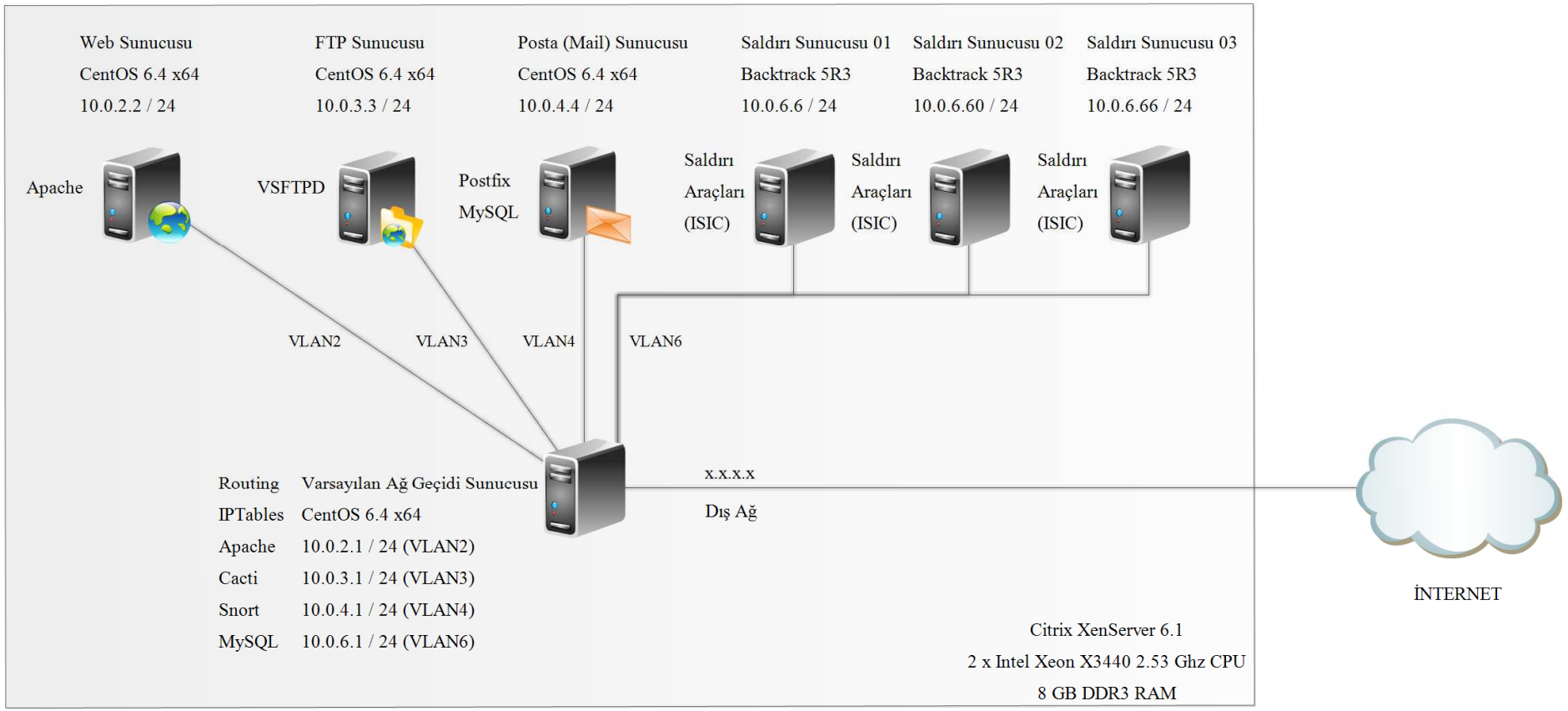
  Saldırı Sunucusu 01 eth0; 10.10.6.6 / 24 (VLAN6)

  Saldırı Sunucusu 02 eth0; 10.10.6.60 / 24 (VLAN6)

  Saldırı Sunucusu 03 eth0; 10.10.6.60 / 24 (VLAN6)

- *ISIC;* saldırı simülasyonları sırasında TCP, UDP ve ICMP paketleri üretilmesi amacıyla kullanılmıştır.

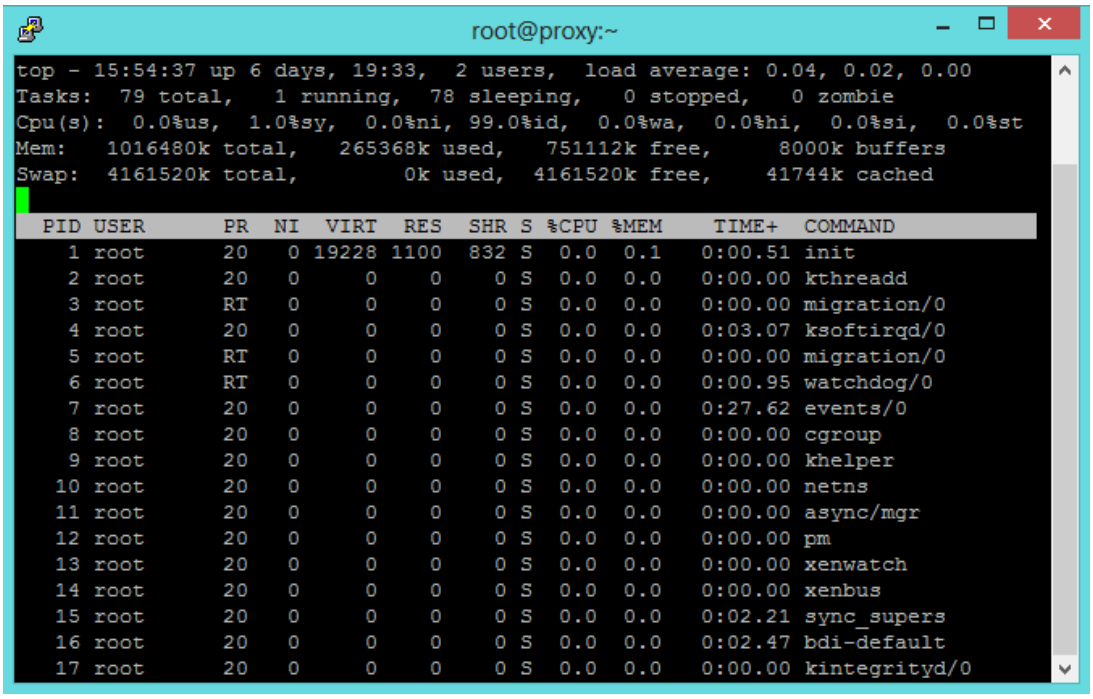
Saldırı simülasyonları sırasında kullanıcı VM'i pozisyonunda bulunan Web Sunucusu, FTP Sunucusu ve Mail Sunucusuna aynı anda sırasıyla ICMP Flooding, TCP Flooding ve UDP Flooding saldırıları gerçekleştirilmiştir. Saldırı simülasyonları esnasında VM'lere ait CPU ve bellek kullanım oranları Altıncı Bölüm'de anlatılmış olup, senaryonun uygulandığı ağ topolojisi Şekil 5.8'de gösterilmiştir.



Şekil 5.8 : IDS'nin varsayılan ağ geçidinde kurulması (Senaryo 03).

6. DENEYSEL SONUÇLAR

Saldırı simülasyonları sırasında CentOS OS'ler üzerinde varsayılan olarak kurulumu gerçekleştirilen *"top"* isimli yazılım kullanılmıştır. *top* ile anlık olarak OS üzerinde çalışan uygulama ve servislerin donanım kullanımları da dahil olmak üzere sistem ile ilgili bazı bilgilerin izlenebilmesini sağlamaktadır. Şekil 6.1'de örnek bir *top* ekran görüntüsü bulunmakta olup, ekran görüntüsünde bulunan bölümler aşağıda açıklanmıştır.



Şekil 6.1 : Varsayılan ağ geçidi konumundaki VM'ye ait *top* çıktısı.

İlk satırda bulunan *"15:54:37"* top komutunun çalıştırıldığı zamanı, *"up 6 days"* sistemin ne kadar süredir çalışmakta olduğunu, *"2 users"* kaç kullanıcının sistem üzerinde giriş yapmış olduğunu, *"load avarage: 0.04, 0.02, 0.00"* sistem üzerindeki yük durumunu (1 dakika, 5 dakika, 15 dakika) göstermektedir. Sistem üzerindeki yük durumu CPU üzerindeki kullanım miktarını oransal olarak göstermektedir.

İkinci satırda bulunan *"tasks"* kısmı, çalışmakta olan işlemlerin sayısını ve o anki durumunu göstermektedir.

Üçüncü satır CPU kullanım ayrıntılarını göstermektedir. *“0.0%us”* kullanıcılara ait işlem miktarının % 0, *“1.0%sy”* sisteme ait işlem miktarının % 1, *“99.0%id”* kullanılabilir CPU miktarını yüzde olarak; *“0.0%wa”* girdi/çıktı (intpu/output – IO) için beklenen CPU zamanını göstermektedir.

Dört ve beşinci satırlar bellek kullanımını göstermektedir. Sırasıyla *“1016480k total”* toplam bellek miktarını, *“265368k used”* kullanılan bellek miktarını, *“751115k free”* bellek üzerindeki boş miktarı, *“8000k buffers”* tampon bellek miktarını ve *“41744k cached”* önbellek olarak kullanılan miktarı göstermektedir.

Kullanılabilecek boşta olan RAM miktarı: *free + (buffer + cached)*

751115k + 8000k + 41744k = 800859k

Programlar tarafından kullanılan RAM miktarı: *used – (buffer + cached)*

265368k – (8000k + 41744k) = 215624k

top CPU’yu en fazla kullanan işlemleri azalan sırayla göstermektedir. *top* ekranında bulunan sütunlara ait bilgiler aşağıda açıklanmıştır.

PID – İşlemin kimlik (identification – ID) numarasını göstermektedir.

USER – İşlemi çalıştıran kullanıcıyı göstermektedir.

PR – İşlemin öncelik sırasını belirtmektedir. (-20 ile 19 arasında ifade edilmekte, 19 en düşük öncelikli işlemi göstermektedir.)

NI – İşlemin öncelik sırasını değiştirmek için kullanılmaktadır (negatif bir değer işlemin öncelik sırasını artırmakta, pozitif bir değer ise düşürmektedir.)

VIRT – Kullanılan sanal bellek miktarını göstermektedir.

RES – İşlem tarafından kullanılan bellek miktarını kilobayt (kb) cinsinden ifade etmektedir.

SHR – Kullanılan paylaşımlı bellek miktarını göstermektedir.

S – Görevin durumunu belirtmektedir. (sleeping – S, uninterruptible sleep – D, running – R, zombies – Z, stopped veya traced – T)

%CPU – Kullanılan CPU yüzdesini göstermektedir.

%MEM – Kullanılan bellek (RAM) yüzdesini göstermektedir.

TIME+ – CPU’nun toplam kullanım zamanını göstermektedir.

COMMAND – İşlemin adını belirtmektedir.

top yazılımında varsayılan olarak 3 saniye aralıklarla donanım kullanım miktarları gösterilmektedir. Daha gerçekçi veriler elde etmek amacıyla yapılandırma dosyası içerisindeki donanım kullanım durum güncelleme frekansı 1 saniye olarak değiştirilmiş olup, snort yazılımının donanım kullanım miktarı için aşağıdakki komut çalıştırılmıştır.

top -b -p 3757 >> S01.01.TCP.Web.txt

Yukarıdaki örnek komut ile *top* yazılımı *-b* parametresi ile batch modda (arka planda) çalıştırılmakta, *-p* parametresi ile 3757 işlem kimlik (process ID – PID) numarasına sahip işlem izlenmekte ve >> işareti ile *S01.01.TCP.Web.txt* isimli dosyaya sonuçlar yazılmaktadır.

Her senaryoda ayrı ayrı gerçekleştirilen saldırı simülasyonlarına ait CPU ve bellek kullanım miktarları *txt* formatında sistem üzerinden alındıktan sonra, zaman/donanım kullanım miktarı grafiklerine aktarılmış olup, senaryolar içerisinde bu grafikler açıklanmıştır.

6.1 IDS'nin kullanıcı VM'leri üzerinde kurulması (Senaryo 01)

IDS'nin kullanıcı VM'leri üzerinde kurulması senaryosunda, IDS olarak belirlenen Snort kullanıcı VM'leri üzerinde konfigüre edilmiştir. Test ortamında bulunan saldırı sunucuları üzerinden kullanıcı VM'lerine sırasıyla ICMP Flooding, TCP Flooding ve UDP Flooding saldırıları bir, iki ve üç milyon paket ile gerçekleştirilmiştir.

Bir milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında; Web Sunucusu üzerinde çalışan Snort CPU'nun yaklaşık %13,4'ünü, belleğin ise yaklaşık %70,9'unu; FTP Sunucusu üzerinde çalışan Snort CPU'nun yaklaşık %16,7'sini, belleğin ise yaklaşık %70,9'unu ve Mail Sunucusu üzerinde çalışan Snort CPU'nun yaklaşık %11,8'ini, belleğin ise yaklaşık %70,7'sini kullanmakta olup; bir, iki ve üç milyon paket ile gerçekleştirilen ICMP Flooding, TCP Flooding ve UDP Flooding saldırılarında Snort tarafından kullanılan CPU ve bellek miktarları Çizelge 6.1'de gösterilmiştir.

Çizelge 6.1 : Saldırı simülasyonlarında donanım kullanımı (Senaryo 01).

Sunucu	Paket Sayısı	ICMP Flooding		TCP Flooding		UDP Flooding	
		CPU (%)	RAM (%)	CPU (%)	RAM (%)	CPU (%)	RAM (%)
Web Sunucusu	1m	13,41	70,90	16,12	73,85	30,41	43,58
	2m	12,88	70,67	15,97	79,68	29,55	72,39
	3m	10,88	70,55	14,8	80,96	30,34	58,23
FTP Sunucusu	1m	16,71	70,96	23,74	75,59	31,96	36,45
	2m	16,35	70,59	24,55	74,52	35,29	70,01
	3m	15,55	70,45	25,38	67,18	32,13	51,05
Mail Sunucusu	1m	11,84	70,72	18,1	74,51	29	33,69
	2m	11,82	70,53	16,91	80,59	29,72	70,38
	3m	9,93	70,44	18,69	69,88	30,86	51,88
Toplam	**1m**	**33,85**	**176,04**	**56,03**	**215,4**	**84,57**	**104,26**
	2m	**35,95**	**187,63**	**53,18**	**214,08**	**89,93**	**202,2**
	3m	**34,42**	**201,6**	**53,85**	**194,9**	**86,65**	**148,56**

6.2 IDS'nin ayrı bir VM üzerinde kurulması (Senaryo 02)

IDS'nin sanal platform üzerindeki ayrı bir VM üzerinde kurulması senaryosunda, IDS olarak belirlenen Snort kullanıcı VM'lerinin dışında bir VM üzerinde konfigüre edilmiştir. Test ortamında bulunan saldırı sunucuları üzerinden kullanıcı VM'lerine sırasıyla ICMP Flooding, TCP Flooding ve UDP Flooding saldırılarının bir, iki ve üç milyon paket ile gerçekleştirildiği varsayılmış olup, üretilen tüm paketlerde hedef olarak IDS'nin konumlandırıldığı VM gösterilmiştir.

Bir milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında, IDS Sunucusu üzerinde çalışan Snort CPU'nun yaklaşık %24,24'ünü, belleğin ise yaklaşık %70,6'sını; iki milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında, CPU'nun yaklaşık %24,86'sını, belleğin ise yaklaşık %70,48'ini ve üç milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında; CPU'nun yaklaşık %23,02'sini, belleğin ise yaklaşık %65,13'unu kullanmakta olup; bir, iki ve üç milyon paket ile gerçekleştirilen ICMP Flooding, TCP Flooding ve UDP Flooding saldırılarında Snort tarafından kullanılan CPU ve bellek miktarları Çizelge 6.2'de gösterilmiştir.

6.3 IDS'nin varsayılan ağ geçidinde kurulması (Senaryo 03)

IDS'nin sanal platform üzerinde varsayılan ağ geçidi konumundaki VM üzerinde kurulması senaryosunda, IDS olarak belirlenen Snort varsayılan ağ geçidi olarak

görevlendirilen VM üzerinde konfigüre edilmiştir. Test ortamında bulunan saldırı sunucuları üzerinden kullanıcı VM'lerine sırasıyla ICMP Flooding, TCP Flooding ve UDP Flooding saldırıları bir, iki ve üç milyon paket ile gerçekleştirilmiştir.

Çizelge 6.2 : Saldırı simülasyonlarında donanım kullanımı (Senaryo 02).

Sunucu	Paket Sayısı	ICMP Flooding		TCP Flooding		UDP Flooding	
		CPU (%)	RAM (%)	CPU (%)	RAM (%)	CPU (%)	RAM (%)
IDS Sunucusu	1m	24,24	70,76	41,15	72,07	76,78	68,86
	2m	24,86	70,48	40,3	73,9	78,83	32,22
	3m	23,02	65,13	42,75	68,9	75,41	48,03

Bir milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında, Gateway Sunucusu üzerinde çalışan Snort CPU'nun yaklaşık %45,67'sini, belleğin ise yaklaşık %70,95'ini; iki milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında, CPU'nun yaklaşık %45,93'ünü, belleğin ise yaklaşık %70,66'sını ve üç milyon ICMP paketiyle gerçekleştirilen ICMP Flooding saldırısında, CPU'nun yaklaşık %42,2'sini, belleğin ise yaklaşık %70,56'sını kullanmakta olup; bir, iki ve üç milyon paket ile gerçekleştirilen ICMP Flooding, TCP Flooding ve UDP Flooding saldırılarında Snort tarafından kullanılan CPU ve bellek miktarları Çizelge 6.3'te gösterilmiştir.

Çizelge 6.3 : Saldırı simülasyonlarında donanım kullanımı (Senaryo 02).

Sunucu	Paket Sayısı	ICMP Flooding		TCP Flooding		UDP Flooding	
		CPU (%)	RAM (%)	CPU (%)	RAM (%)	CPU (%)	RAM (%)
Gateway Sunucusu	1m	45,67	70,95	58,18	91,02	84,7	80,63
	2m	45,93	70,66	59,69	101,24	87,54	80,81
	3m	42,2	70,56	60,63	98,67	74,4	79,97

6.4 Test Sonuçlarının Karşılaştırılması

Kararlı sonuçlar elde edilmesi amacıyla test ortamında bulunan tüm VM'lerin standart donanıma sahip olması hedeflenmiş olup, tüm VM'ler bir adet CPU ve 1024 MB RAM'e sahip olarak konfigüre edilmiştir. Ancak, saldırı simülasyonları sırasında bazı VM'lerde bellek yetersizliğinden kaynaklı olarak Snort'un belirli bir süre tespitleri gerçekleştirdikten sonra kendisini sonlandırması nedeniyle, sorun yaşanan VM'lerdeki bellek miktarları ihtiyaç doğrultusunda iki veya iki buçuk katına çıkartılmış ve elde edilen donanım kullanım yüzdeleri bu bağlamda göreceli olarak oranlanmıştır.

Snort tarafından bir milyon paket ile gerçekleştirilen saldırı simülasyonlarına ait donanım kullanım yüzdeleri Çizelge 6.4'te; iki milyon paket ile gerçekleştirilen saldırı simülasyonlarına ait donanım kullanım yüzdeleri Çizelge 6.5'te ve üç milyon paket ile gerçekleştirilen saldırı simülasyonlarına ait donanım kullanım yüzdeleri Çizelge 6.6'da gösterilmiştir. Donanım kullanım oranları incelendiği zaman Senaryo 02'de uygulanan, IDS'nin ağ üzerinde ayrı bir VM'de kurulmasının daha verimli bir donanım kullanımına sahip olduğu görülmektedir.

Çizelge 6.4 : Test sonuçlarının karşılaştırılması (1m paket).

Senaryo	ICMP Flooding		TCP Flooding		UDP Flooding	
	CPU (%)	RAM (%)	CPU (%)	RAM (%)	CPU (%)	RAM (%)
Senaryo 01	33,85	176,04	56,03	215,4	84,5	104,26
Senaryo 02	**24,24**	**70,76**	**41,15**	**72,07**	**76,78**	**68,86**
Senaryo 03	45,67	70,95	58,18	91,02	84,7	80,63

Çizelge 6.5 : Test sonuçlarının karşılaştırılması (2m paket).

Senaryo	ICMP Flooding		TCP Flooding		UDP Flooding	
	CPU (%)	RAM (%)	CPU (%)	RAM (%)	CPU (%)	RAM (%)
Senaryo 01	35,95	187,63	53,18	214,08	89,93	202,2
Senaryo 02	**24,86**	**70,48**	**40,3**	**73,93**	**78,83**	**32,22**
Senaryo 03	45,93	70,66	59,69	101,24	87,54	80,81

Çizelge 6.6 : Test sonuçlarının karşılaştırılması (3m paket).

Senaryo	ICMP Flooding		TCP Flooding		UDP Flooding	
	CPU (%)	RAM (%)	CPU (%)	RAM (%)	CPU (%)	RAM (%)
Senaryo 01	34,42	201,6	53,85	194,9	86,65	148,56
Senaryo 02	**23,02**	**65,13**	**42,75**	**68,9**	75,41	**48,03**
Senaryo 03	42,2	70,56	60,63	98,67	**74,4**	79,97

Test ortamında gerçekleştirilen saldırı simülasyonlarında CPU ve bellek kullanım oranlarının yanında oluşan ağ trafiği bulut kullanıcısının ücretlendirilmesi bağlamında etkili olmamaktadır. Bunun nedeni servis sağlayıcılarının ağ kullanım miktarlarını, internet üzerinden gerçekleşen indirme (download) ve yükleme (upload) oranlarıyla ücretlendiriyor olmasıdır. Senaryo 01 ve Senaryo 03'te iç ağ üzerinde meydana gelen ağ trafiği aynı olup, Senaryo 02'de oluşan ağ trafiği, tamamının kopyalanarak farklı bir VM üzerine yönlendirilmesi nedeniyle diğer senaryoların iki katı şeklinde gerçekleşmektedir.

Senaryo 01'in en fazla donanım kullanımına sahip olmasının nedeni, Snort'un herbir VM üzerinde ayrı bir servis olarak çalıştırılması ve Snort'un en alt seviyedeki temel bileşenlerinin (paket yakalama ve saldırı imza veri tabanı vb.) her Snort servisi için ayrı ayrı çalıştırılıyor olmasıdır. Senaryo 03'ün Senaryo 02'ye göre daha fazla CPU ve bellek kullanmasının nedeni ise varsayılan ağ geçidi konumundaki VM üzerinden geçen ağ paketlerinin analizleri yapıldıktan sonra tekrar asıl hedeflerine gönderiliyor olmasıdır. Snort yakaladığı paketlerin analizini gerçekleştirdikten sonra analizi gerçekleştirdiği paketleri tekrar sarmalamakta (encapsulation) ve ağ üzerinden yönlendirmekte olup, bu işlemleri gerçekleştirirken de ilave CPU ve bellek gücüne ihtiyaç duymaktadır.

Saldırı simülasyonları esnasında VM'ler üzerinde Snort tarafından kullanılan donanım miktarlarına ait karşılaştırma Çizelge 6.7'de gösterilmiştir. Bulut kullanıcılarının yanında servis sağlayıcılarının da kendi sistemleri üzerinde gerçekleştirecekleri IDS konumlandırmalarını, ortaya konan donanım kullanım miktarları doğrultusunda gerçekleştirmelerinin daha maliyet etkin bir yöntem olacağı görülmektedir.

Çizelge 6.7 : Saldırı simülasyonları esnasında gerçekleşen donanım kullanımları.

Senaryo	Donanım Kullanımı		
	CPU	Bellek	Ağ Trafiği
Senaryo 01	> Senaryo 2, > Senaryo 3	> Senaryo 2, > Senaryo 3	x
Senaryo 02	**< Senaryo 1, < Senaryo 3**	**< Senaryo 1, < Senaryo 3**	**~2x**
Senaryo 03	< Senaryo 1, > Senaryo 2	< Senaryo 1, > Senaryo 2	x

7. SONUÇ VE GELECEK ÇALIŞMALAR

Bulut bilişim, günümüzde geleneksel bilgi sistemi yaklaşımlarının yerini almaya başlayan ve gün geçtikçe tercih edilme oranı artan teknolojilerin başında gelmektedir. Bulut bilişimin doğası gereği, servis sağlayıcılar kullanıcılarına belirli gereksinim ve gereklilikler doğrultusunda hizmet vermektedir. Bunu temelde IaaS, PaaS ve SaaS olmak üzere sundukları servisler ile gerçekleştirmektedirler. Servis sağlayıcı ve kullanıcı arasında varılan mutabakat ile belirlenen maliyetler doğrultusunda kullanıcı, kullandığı servis miktarı kapsamında maliyetlendirilmektedir. Literatürde buna *kullandığın kadar öde* modeli ismi verilmektedir.

Bulut kullanıcılarının ihtiyaçları, geleneksel bilgi sistemlerindekinden çok büyük farklılıklar göstermemekte olup, özellikle servislerin sunulduğu altyapıda kullanılan donanım ve ağırlıklı olarak ta yazılımın, üçüncü şahısların kontrolünde bulunmasından dolayı, güvenlik ve gizlilik temel sistem gereksinimlerinde öne çıkmaktadır. Kullanıcıların güvenlik ihtiyaçlarını karşılamaları, sunulan servislerin bünyesinde işlenen veya saklanan verilerin de önemine bağlı olarak belirli mekanizmaların kullanılmasıyla mümkün olabilmektedir.

Geleneksel bilgi sistemlerinde olduğu gibi, bulut bilişimde yaşanan saldırılar, gerçeklenme kolaylığı ve yapısının karmaşıklığı nedeniyle çoğunlukla ağ üzerinden gerçekleştirilmektedir. Ağ üzerinden gerçekleştirilecek saldırıların önüne geçmek amacıyla, yönlendiriciler üzerinde koruma mekanizmasının kurulması ve güvenlik duvarı kullanılması en temel güvenlik tedbirlerindendir. Bu tedbirler genellikle OSI referans modelinde ağ katmanında gerçekleşmekte olup, ağ paketleri içerisinde bulunan veriler kullanılarak uygulama katmanında gerçekleştirilen saldırılara karşı IDS veya IDPS kullanılması gerekmektedir.

Saldırı tespit mekanizması çalışırken, güvenliğini sağladığı bilgi sistem varlığının yoğunluğuna bağlı olarak işlediği veri miktarından kaynaklı, donanım ihtiyacı ortaya çıkmaktadır. Bu sistemin ağ üzerinden gelen tehditlere karşı bir mekanizma olarak görev yapması durumunda, ağ bant genişliğine bağlı olarak analizi gerçekleştirilen paketler yüksek miktarlarda CPU ve bellek kullanımına neden olmaktadır.

Tüm bunların sonucu olarak, bulut bilişim bünyesinde tesis edilecek bir IDS mekanizması, kullanıcının *kullandığın kadar öde* modeliyle ücretlendirilmesinden dolayı maliyetlerini artırmaktadır. Ağ üzerinden gelebilecek saldırılara karşı tespit işlemlerini gerçekleştirecek bir IDS'nin ağ üzerinde farklı yerlerde konumlandırılması sonucunda ortaya çıkacak maliyetler, donanım kullanımlarının değişmesine bağlı olarak farklılıklar gösterecektir. Donanım kullanım miktarlarının değişmesine bağlı olarak ta kullanıcının maliyetleri değişecektir.

Ağ topolojisi üzerinde Ağ Tabanlı Saldırı Tespit Sistemleri (Network Based Intrusion Detection System – NIDS)'nin konumlandırılması literatürde farklılıklar göstermekle birlikte, bulut bilişim bünyesinde kullanılan gerçekleme teknolojilerinin başında gelen sanallaştırma üzerinde NIDS'in konumlandırılması üç farklı şekilde gerçekleştirilebilmektedir. Bunlar sırasıyla; güvenliği sağlanan *VM üzerinde*, ağ *üzerindeki farklı bir VM üzerinde* ve *varsayılan ağ geçidi konumundaki bir VM üzerindedir*. Çalışma kapsamında bu üç farklı konumlandırma senaryosu dâhilinde NIDS tesis edilmiştir. Sanal ortam içerisinde varsayılan ağ geçidi konumundaki VM'nin arkasındaki iç ağda bulunan üç farklı VM üzerine, gerçeklenmesi göreceli olarak daha kolay olan DoS saldırıları (ICMP Flooding, TCP Flooding ve UDP Flooding) gerçekleştirilerek tesis edilen NIDS'lerin CPU ve bellek kullanımları incelenmiştir. Saldırı senaryolarının geçekleştirilmesi sırasında gerçeğe daha yakın sonuçlar elde edilmesi amacıyla, bulut bilişim ve sanallaştırma teknolojilerinde ağırlıklı olarak kullanılan Açık Kaynak Kodlu (AKK) yazılım ve uygulamalar tercih edilmiştir.

Bu çalışma kapsamında ağ tabanlı bir IDS'nin en iyi donanım kullanım durumu, farklı konumlandırma senaryoları üzerinde incelenmiş olup; temel olarak saldırı tespit görevinin, güvenliği sağlanan varlığın dışında ve bir yerde vekil sunucu görevinde bulunan bir varlık tarafından gerçekleştirilmesi nedeniyle, *Bulut Bilişimde Vekil Ağ Saldırı Tespit Sistemi* ismi kullanılmıştır.

Saldırı benzetimleri öncesinde varsayılan ağ geçidi pozisyonundaki VM üzerinde tesis edilen NIDS'in donanım kullanım oranlarının, üç senaryo içerisinde en iyi olması beklenmiştir. Bunun birinci nedeni, VM'ler üzerinde tesis edilen NIDS'lerin her birinin saldırı tespit işlemlerini ayrı ayrı gerçekleştirecek olması ve bunun sonucunda yüksek donanım kullanım oranlarına ulaşılmasının beklenmesidir. İkinci olarak, sistemde ayrı bir VM üzerine tesis edilecek NIDS'in ağ trafiğinin tamamının kendi

üzerine kopyalanması sonucunda yüksek miktarlarda ağ yükü ve darboğazların oluşmasının yanında, varsayılan ağ geçidi pozisyonundaki VM ile aynı donanım kullanımına sahip olması beklenmiştir. Tüm ağ trafiğinin bu VM üzerine kopyalanması sonucu ağ üzerinde oluşacak trafik yükü nedeniyle (yaklaşık iki kat), varsayılan ağ geçidi üzerinde tesis edilen NIDS'in daha verimli bir donanım kullanım oranı ortaya koyması öngörülmüştür.

Saldırı benzetimleri sonrasında elde edilen donanım kullanım oranları EK'te sunulmuştur. Bu sonuçlar incelendiğinde, varsayılan ağ geçidi konumundaki VM üzerinde tesis edilen NIDS'in en iyi donanım kullanım oranlarına sahip olmadığı açıkça görülmektedir. Bunun nedeni, saldırı tespit işlemleri sırasında NIDS olarak kullanılan Snort'un çalışma mekanizması dâhilinde, ağ üzerinde yakalanan paketlerin analizlerinin yapılmak üzere açılması, içeriğinin incelenmesi ve tekrar sarmalanarak ağ üzerinde asıl gönderilmesi gereken hedefe yollanması sırasında meydana gelen donanım kullanımıdır. İkinci senaryo olan, sanal ortam bünyesinde farklı bir VM üzerinde kurulan NIDS'te ise en verimli donanım kullanımı gerçekleşmiştir. Bu senaryoda meydana gelen ağ trafiği göz ardı edilebilmektedir. Bunun nedeni, bulut bilişimde *kullandığın kadar öde* yaklaşımındaki ağ trafiği kullanım miktarının belirlenmesinde internet ortamına aktarılan veya internet ortamından alınan veri miktarının belirleyici olmasıdır. Bu durumda ise ağ trafiği sanal ortam bünyesinde gerçekleşmekte olup, sağlayıcıya ve dolayısıyla da kullanıcıya ilave bir ağ yükü getirmemekte, bunun sonucu olarak ta kullanıcının maliyeti ağ haberleşmesi nedeniyle artışa maruz kalmamaktadır.

Saldırı benzetimlerinin gerçekleştirilmesi esnasında, NIDS tarafından birbirine yakın oranlarda bellek kullanımının gerçekleştiği, CPU kullanımının ise paket sayısı arttıkça UDP Flooding saldırısında daha yüksek olduğu gözlemlenmiştir. Bunun nedeni; ICMP, TCP ve UDP arasındaki karakteristik farklılıklardır. ICMP ve UDP Flooding saldırılarında iki yönlü karşılıklı el sıkışmanın gerçekleşmemesinden kaynaklı olarak, oluşturulan paketler daha yoğun bir şekilde hedefe ulaşmakta, TCP Flooding saldırısında ise bu süreç biraz daha genişlemekte ve daha düşük bir frekansta oluşturulan paketler hedefe ulaşmaktadır. Bunun yanında ICMP paketleri içerisinde bir veri bulunmaması ve sadece kontrol bitleri ile sağlama (cheksum) değerlerinin bulunması nedeniyle, analizler esnasında daha az miktarda CPU kullanımı gerçekleşmektedir.

Bulut Bilişimde Vekil Ağ Saldırı Tespit Sistemi sonucunda elde edilen donanımsal kullanım oranları neticesinde, sadece kullanıcıların değil, servis sağlayıcıların da kendi sistemleri üzerinde saldırı tespit işlemlerini gerçekleştirmek amacıyla daha verimli bir donanım kullanımına sahip olan ayrı bir VM üzerinde tesis edilecek NIDS'i kullanarak kendi maliyetlerini azaltabilecekleri değerlendirilmektedir. Bulut bilişimin doğası gereği merkeziyetçi bir mimariye sahip olmayan, birden fazla konumda kümelenen veri merkezlerinin kullanılması nedeniyle, veri merkezleri arasında gerçekleşen ağ trafiği üzerindeki saldırı girişimleri de ayrı bir VM veya donanımsal varlık üzerinde daha düşük bir maliyetle gerçekleştirilebilecektir.

7.1 Gelecek Çalışmalar

Donanım kullanım oranlarının gözlemlemesi için oluşturulan test ortamında rutin bir ağ trafiğinin olmaması nedeniyle, anormallik tabanlı saldırı tespit yaklaşımının kullanılması durumunda gerçekten çok uzak sonuçların elde edilmesi öngörülmüştür. Bu nedenle, imza tabanlı saldırı tespit mekanizmasının kullanılması tercih edilmiştir. NIDS'ler üzerinde daha fazla yük oluşturmak ve sistemleri daha fazla yormak adına Snort'un internet sitesi üzerinden [57] güncel olarak indirilen saldırı imzalarının tamamı (yaklaşık olarak 12,000 adet) ağ paketlerinin analizlerinde kullanılmıştır. Gerçeğe daha yakın sonuçlar elde etmek adına, normalitesi gözlemlenerek oluşturulacak profiller üzerinden saldırı tespit edilebilmesine imkân verecek gerçek bir bulut ortamı veya en azından sanal ağ trafiği oluşturulmasının uygun olacağı değerlendirilmektedir. Bu sayede gerçekleştirilecek saldırı benzetimleri ile daha gerçekçi sonuçların elde edilmesi mümkün olabilecektir. Yanlış negatif ve yanlış pozitif saldırı tespitleri, sistem üzerinde güvenlik önlemleri alınmasının önünde bir engeldir. Bunun önüne geçmek amacıyla, anormallik ve imza tabanlı saldırı tespit mekanizmasının birlikte kullanılmasıyla elde edilecek sonuçlar, sistemin donanım kullanım oranlarının gözlemlenmesi ve maliyet analizlerinin yeniden yapılmasının yanında, daha güvenli bir ortam tesis edilmesine de olanak sağlayacaktır.

Donanım kullanım oranları, saldırı benzetimlerinin tek bir sefer gerçeklenmesi ile elde edilmiştir. Daha gerçekçi ve ortalama sonuçlar elde etmek, bunun yanında VM'ler üzerinde çalışan diğer servis ve uygulamaların saldırı tespiti sırasındaki donanım kullanımına etkilerini azaltmak amacıyla, saldırı benzetimlerinin belirlenecek miktarlarda tekrarlanarak gerçeklenmesinin daha uygun olacağı değerlendirilmektedir.

KAYNAKLAR

[1] **Bharadwaja S., Sun W., Niamat M. ve Shen F.** (2011), Collabra: A Xen Hypervisor based Collaborative Intrusion Detection System. Proceedings of the Eighth International Conference on Information Technology: New Generations, (Sf. 695-700), 11-13 Nisan 2011.

[2] **Yassin W., Udzir N.I., Muda Z., Abdullah A. ve Abdullah M.T.** (2012), A Cloud-Based Intrusion Detection Service Framework. Proceedings of International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), (Sf. 213-218), 26-28 Haziran 2012.

[3] **Vanathi R.ve Gunasekaran S.** (2012), Comparison of Network Intrusion Detection Systems in cloud computing environment. Proceedings of International Conference on Computer Communication and Informatics (ICCCI), (Sf. 1-6), 10-12 Ocak 2012.

[4] **Alharkan T. ve Martin P.** (2012), IDSaaS: Intrusion Detection System as a Service in Public Clouds. Proceedings of 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), (Sf. 686-687), 13-16 Mayıs 2012.

[5] **Loganayagi B. ve Sujatha S.** (2012), Enhanced Cloud Security by Combining Virtualization and Policy Monitoring Techniques. Procedia Engineering, (Cilt 30, Sf. 654-661), 2012.

[6] **Krautheim F.J.** (2009), Private virtual infrastructure for cloud computing. Proceedings of Conference on Hot Topics in Cloud Computing (HotCloud'09), (Sayı 5), 14-19 Haziran 2009.

[7] **Pfoh J., Schneider C. ve Eckert C.** (2009), A Formal Model for Virtual Machine Introspection. Proceedings of the 2nd ACM Workshop on Virtual Machine Security (VMSec '09), (Sf. 1-10), 9-13 Kasım 2009.

[8] **Kong J.** (2010), Protecting the Confidentiality of Virtual Machines Against Untrusted Host. Proceedings of International Symposium on Intelligence Information Processing and Trusted Computing (IPTC), (Sf. 364-368), 28-29 Ekim 2010.

[9] **Kong J.** (2011), AdjointVM: a New Intrusion Detection Model for Cloud Computing. Energy Procedia, (Cilt 13, Sf. 7902-7911), 2011.

[10] **Oktay U., Aydın M.A. ve Şahingöz Ö.** (2013), Circular Chain VM Protection in AdjointVM. Proceedings of The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE2013), 9-11 Mayıs 2013.

[11] **Mazzariello C., Bifulco R. ve Canonico R.** (2010), Integrating a network IDS into an open source Cloud Computing environment. Proceedings of Sixth International Conference on Information Assurance and Security (IAS 2010), 23-25 Ağustos 2010.

[12] **Weiser, M.** (1991), The Computer for the 21st Century. Scientific American Special Issue on Communications, Computers, and Networks. (Sf. 94-104), Eylül 1991.

[13] **Dick, P.K.** (1969), Ubik, Doubleday.

[14] **Amor, D.** (2002), Internet Future Strategies, How Pervasive Computing Services Will Change the World. Prentice Hall Computer Books, New Jersey, Ağustos 2000.

[15] **Schick, L. ve Malmborg L.** (2010) Bodies, embodiment and ubiquitous computing, Digital Creativity, (Cilt 21, Sayı 1, Sf. 63-69), 2010.

[16] **Mell P. ve Grance T.** (2011), The NIST Definition of Cloud Computing, NIST Special Publication 800-145 (SP800-145). National Institute of Standards and Technology, Gaithersburg, Eylül 2011.

[17] **Tupakula, U., Varadharajan, V. ve Akku, N.** (2011), Intrusion Detection Techniques for Infrastructure as a Service Cloud. Proceedings of the IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC), (Sf. 744-751), 12-14 Aralık 2011.

[18] **Lin, G., Fu, D. Zhu, J. ve Dasmalchi, G.** (2009), Cloud Computing IT as a Service. IT Professional, (Cilt 11, Sayı 2, Sf. 10-13), Mart-Nisan 2009.

[19] **Voas, J. ve Zhang, J.** (2009), Cloud Computing New Wine or Just a New Bottle. IT Professional, (Cilt 11, Sayı 2, Sf. 15-17), Mart-Nisan 2009.

[20] **Grossman, R.L.** (2009), The Case for Cloud Computing. IT Professional, (Cilt 11, Sayı 2, Sf. 23-27), Mart-Nisan 2009.

[21] **Hogan, M., Liu, F., Sokol, A. ve Tong, J.** (2011), NIST Cloud Computing Standards Roadmap, NIST Special Publication 500-291 (SP500-291), National Institute of Standards and Technology, Gaithersburg, Temmuz 2011.

[22] **Leavitt, N.** (2009), Is Cloud Computing Really Ready for Prime Time. Leavitt Communications, (Cilt 42, Sayı 1, Sf. 15-20), Ocak 2009.

[23] **Internet Data Corporation (IDC),** http://www.idc.com/ (Erişim tarihi: Nisan 2013).

[24] **Yüksel, H.** (2012), Bulut Bilişim El Kitabı, Ocak 2012.

[25] **Bilişim Hukuku,** http://www.bilisimhukuk.com/. (Erişim tarihi: Nisan 2013).

[26] **European Union (EU) The Data Protection Directive 95/46/EC,** Ekim 1995.

[27] **European Union (EU) General Data Protection Regulation (GDPR) (Taslak),** Ocak 2012.

[28] **European Network and Information Security Agency (ENISA) Cloud Computing Security Risk Assessment,** Kasım 2009.

[29] **Zissis D., ve Lekkas, D.** (2012), Addressing cloud computing security issues. Future Generation Computer Systems, (Cilt 28, Sayı 3, sf. 583-592), Mart 2012.

[30] **Lee H., Chung K, Chin S., Lee J., Lee D., Park S., ve Yu H.** (2005), A resource management and fault tolerance services in grid computing. Journal of Parallel and Distributed Computing, (Cilt 65, Sayı 11 sf. 1305-1317), Kasım 2005.

[31] **Khoo B., Veeravalli B., Hung T. ve Seec S.** (2007), A multi-dimensional scheduling scheme in a Grid computing environment. Journal of Parallel and Distributed Computing, (Cilt 67, Sayı 6, sf. 659-673), Haziran 2007.

[32] **Shelly G.B. ve Vermaat M.E.** (2011) Discovering Computers 2011. Course Technology Cengage Learning, Boston, Şubat 2010.

[33] **Lowe D.** (2012), Networking All-in-One Desk Reference for Dummies. Wiley Publishing, New Jersey, Kasım 2012.

[34] **Scarfone K., Souppaya M., ve Hoffman P.** (2011), Guide to Security for Full Virtualization Technologies, NIST Special Publication 800-125 (SP800-125). National Institute of Standards and Technology, Gaithersburg, Ocak 2011.

[35] **Hess K. ve Newman A.** (2009), Practical Virtualization Solutions. Prentice Hall, Boston, Ekim 2009.

[36] **Barrett D. ve Kipper G.** (2010), Virtualization and Forensics A Digital Forensic Investigator's Guide to Virtual Environment. Elsevier Syngress, Burlington, Haziran 2010.

[37] **Finn A., Lownds P., Luescher M. ve Flynn D.** (2012), Windows Server 2012 Hyper-V Installation and Configuration Guide. Sybex, Indiana Mart 2013.

[38] **Microsoft Hyper-V Server,** www.microsoft.com/hyper-v-server/. (Erişim tarihi: Mayıs 2013).

[39] **VmWare,** http://www.vmware.com/. (Erişim tarihi: Mayıs 2013).

[40] **Virtual Box,** https://www.virtualbox.org/. (Erişim tarihi: Mayıs 2013).

[41] **Tosatto D.** (2013), Citrix XenServer 6.0 Administration Essential Guide. Packt Publishing, Birmingham, Haziran 2012.

[42] **Citrix Xen Server,** http://www.citrix.com/. (Erişim tarihi: Mayıs 2013).

[43] **Scarfone K., ve Mell P.** (2007), Guide to Intrusion Detection and Prevention Systems (IDPS), NIST Special Publication 800-94 (SP800-94). National Institute of Standards and Technology, Gaithersburg, Şubat 2007.

[44] **Tyler, G.** (2009), Information Assurance Tools Report Intrusion Detection Systems. Information Assurance Technology Analysis Center (IATAC), Eylül 2009.

[45] **Modi C., Patel D., Borisaniya B., Patel H., Patel A. ve Rajarajan M.** (2013), A survey of intrusion detection techniques in Cloud. Journal of Network and Computer Applications, (Cilt 36, Sayı 1, Sf. 42-57), Ocak 2013.

[46] **H2B Internet Technologies,** http://www.h2b.com.tr/. (Erişim tarihi: Mayıs 2013).

[47] **PlusClouds,** https://www.plusclouds.com/. (Erişim tarihi: Mayıs 2013).

[48] **RedHat,** http://www.redhat.com/. (Erişim tarihi: Mayıs 2013).

[49] **CentOS,** http://www.centos.org/. (Erişim tarihi: Mayıs 2013).

[50] **IPTables,** http://www.netfilter.org/. (Erişim tarihi: Mayıs 2013).

[51] **Apache Web Sunucusu,** http://www.apache.org/. (Erişim tarihi: Mayıs 2013).

[52] **Netcraft,** http://www.netcraft.com. (Erişim tarihi: Mayıs 2013).

[53] **VSFTPD,** https://security.appspot.com/vsftpd.html. (Erişim tarihi: Mayıs 2013).

[54] **Postfix,** http://www.postfix.org/. (Erişim tarihi: Mayıs 2013).

[55] **MySQL Server,** http://www.mysql.com/. (Erişim tarihi: Mayıs 2013).

[56] **Cacti,** http://www.cacti.net. (Erişim tarihi: Mayıs 2013).

[57] **Snort,** http://www.snort.org. (Erişim tarihi: Mayıs 2013).

[58] **Rehman R.U.** (2003), Intrusion Detection Systems with Snort. Pearson, New Jersey, 2003

[59] **BASE,** https://www.secureideas.com. (Erişim tarihi: Mayıs 2013).

[60] **Backtrack,** http://www.backtrack-linux.org/. (Erişim tarihi: Mayıs 2013).

[61] **Debian,** http://www.debian.org/. (Erişim tarihi: Mayıs 2013).

[62] **Kali Linux,** http://www.kali.org/. (Erişim tarihi: Mayıs 2013).

[63] **Hping,** http://www.hping.org/. (Erişim tarihi: Mayıs 2013).

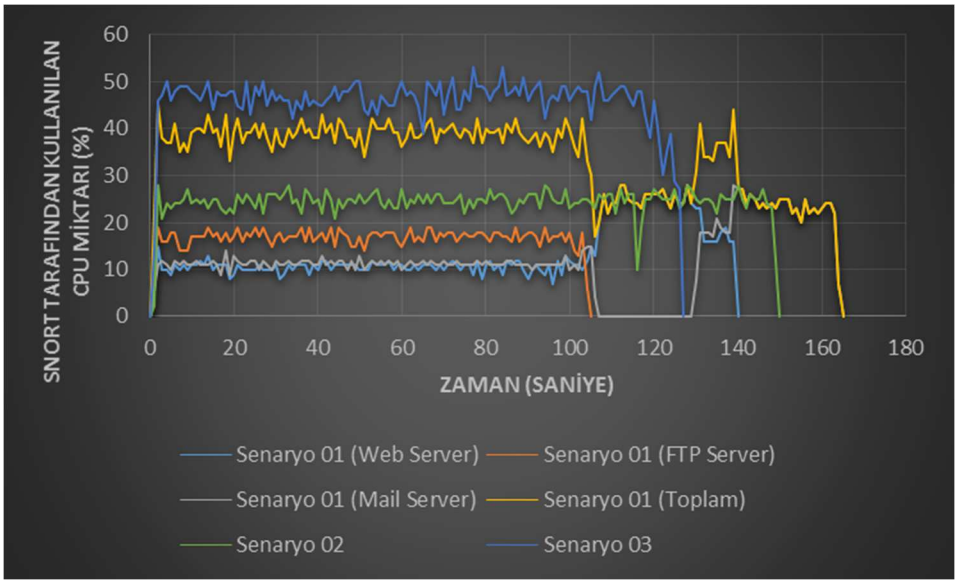
[64] **Nping,** http://nmap.org/nping/. (Erişim tarihi: Mayıs 2013).

[65] **Libnet,** http://packetfactory.openwall.net/projects/libnet/. (Erişim tarihi: Mayıs 2013).

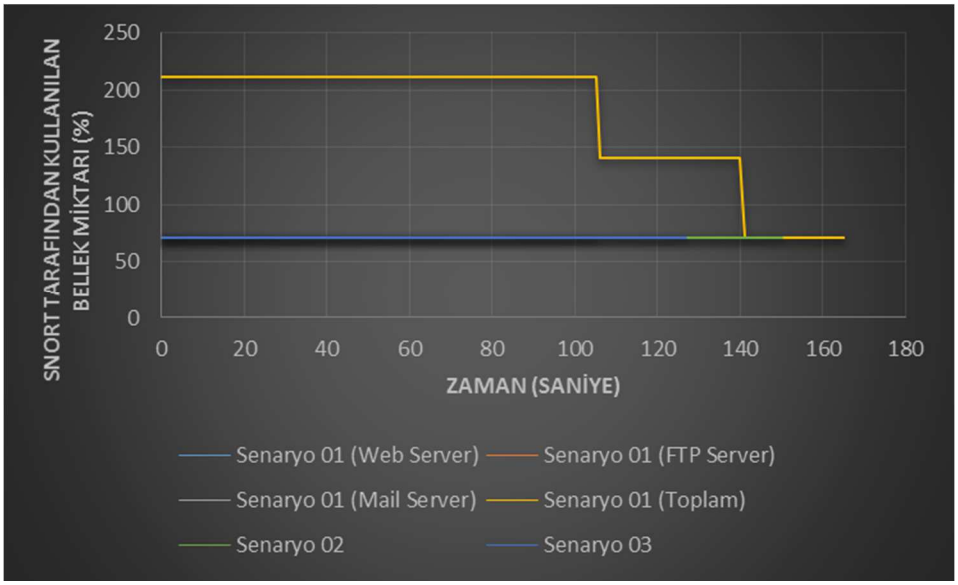
EKLER

EK A: Saldırı simülasyonlarında donanım kullanım oranları.

EK A



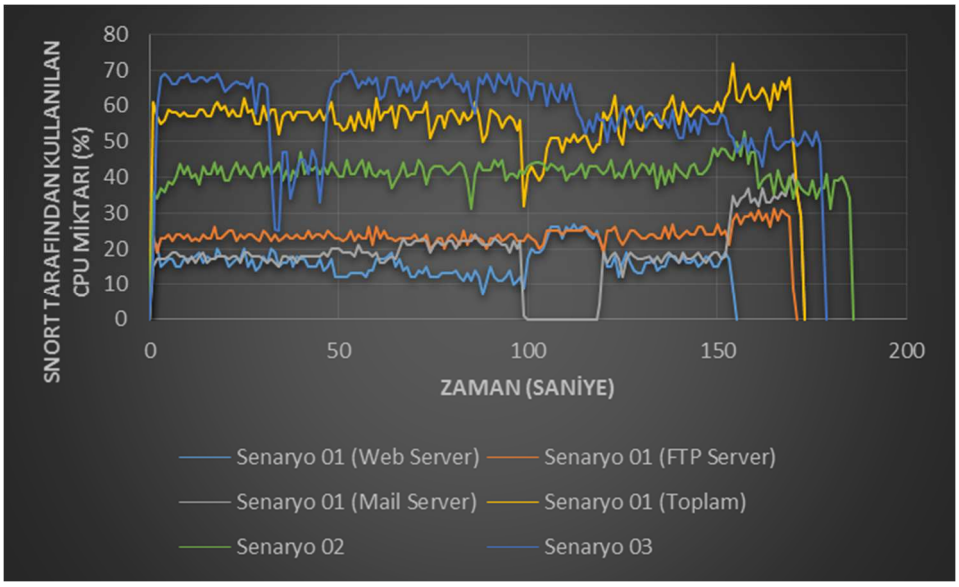
(a) ICMP Flooding saldırısında CPU kullanımı (1,000,000 paket)



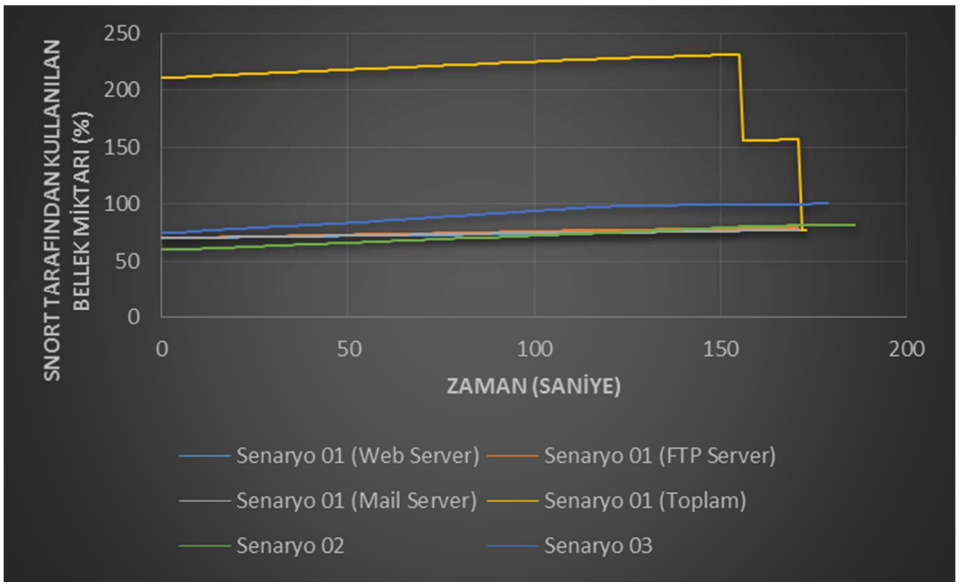
(b) ICMP Flooding saldırısında bellek kullanımı (1,000,000 paket)

Şekil A.1 : ICMP Flooding saldırısında donanım kullanımı (1m paket).

EK A



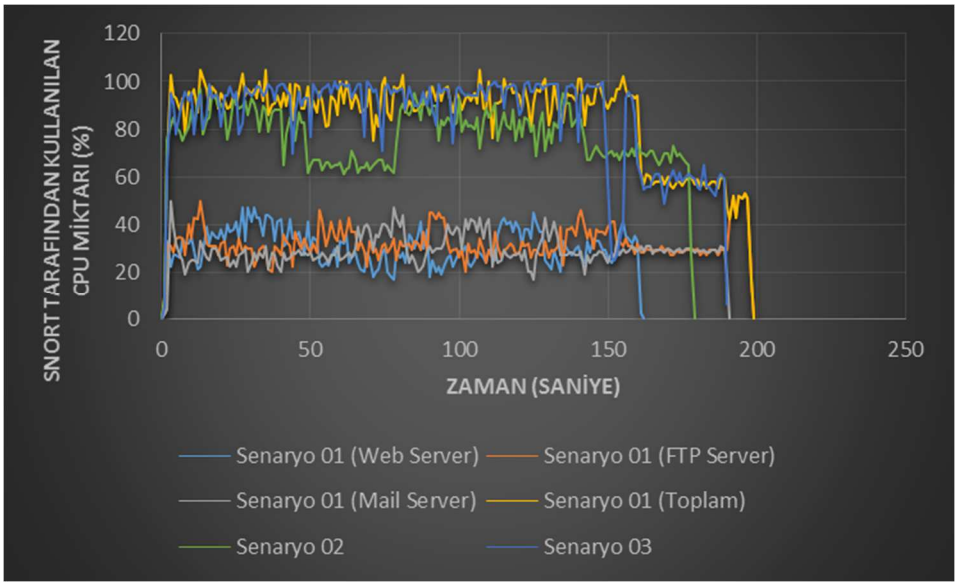
(a) TCP Flooding saldırısında CPU kullanımı (1,000,000 paket)



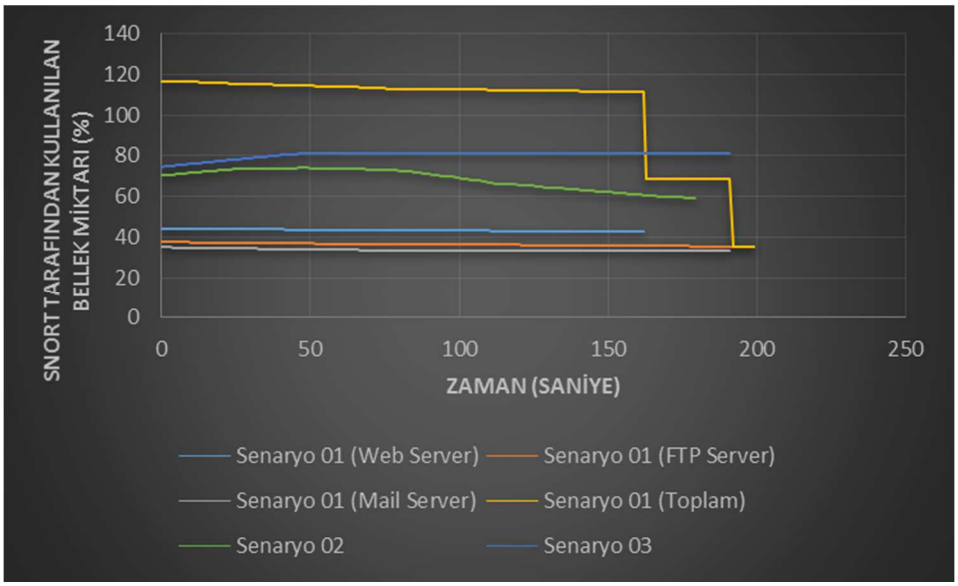
(b) TCP Flooding saldırısında bellek kullanımı (1,000,000 paket)

Şekil A.2 : TCP Flooding saldırısında donanım kullanımı (1m paket).

EK A



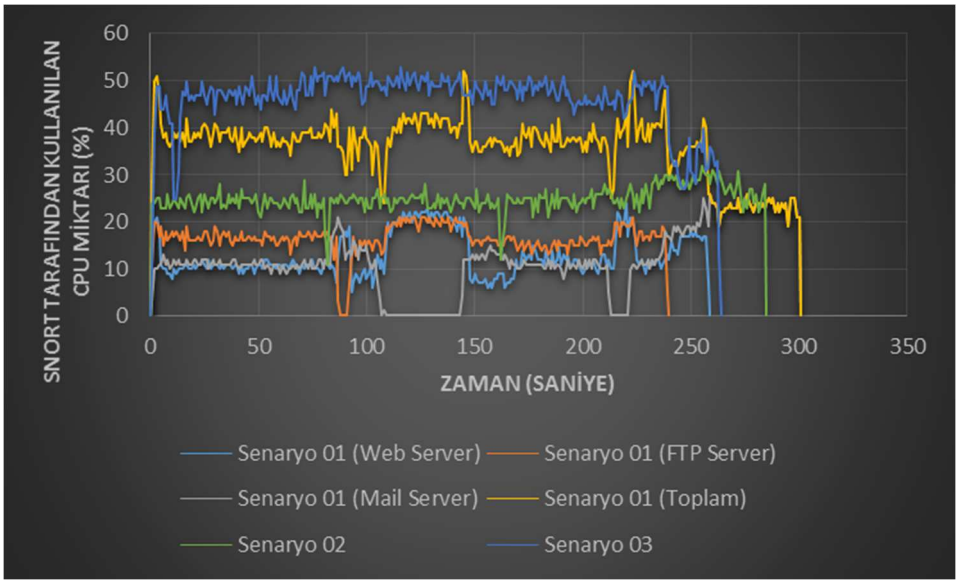
(a) UDP Flooding saldırısında CPU kullanımı (1,000,000 paket)



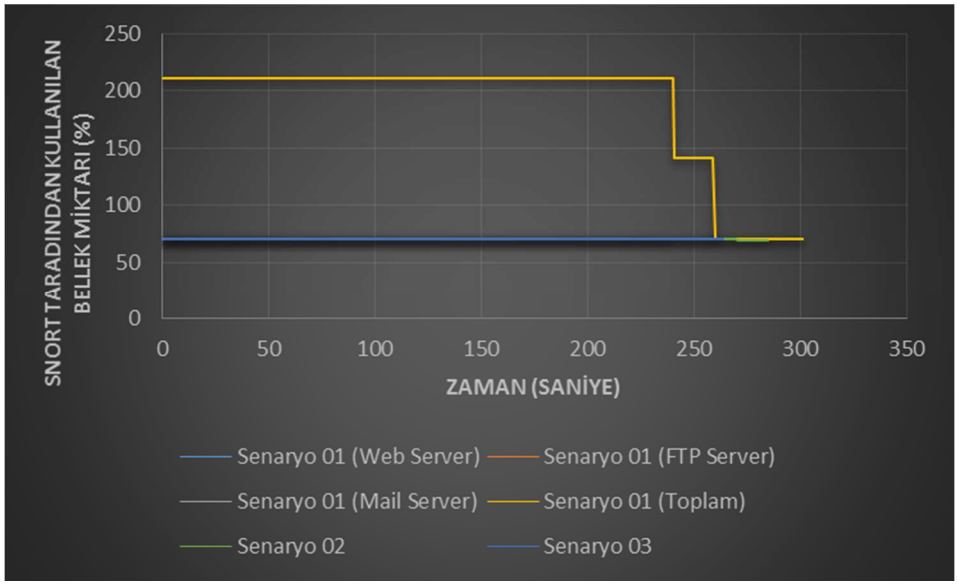
(b) UDP Flooding saldırısında bellek kullanımı (1,000,000 paket)

Şekil A.3 : UDP Flooding saldırısında donanım kullanımı (1m paket).

EK A



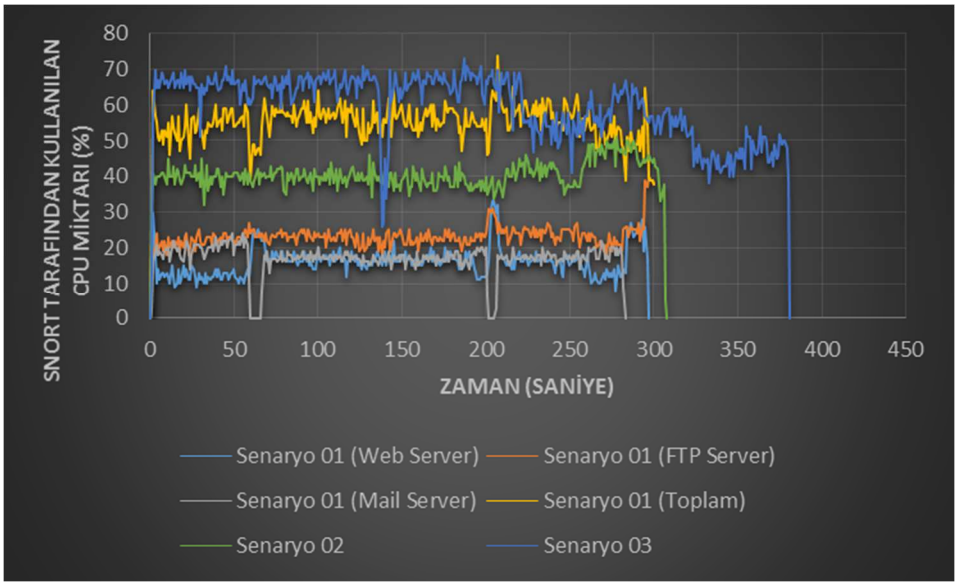
(a) ICMP Flooding saldırısında CPU kullanımı (2,000,000 paket)



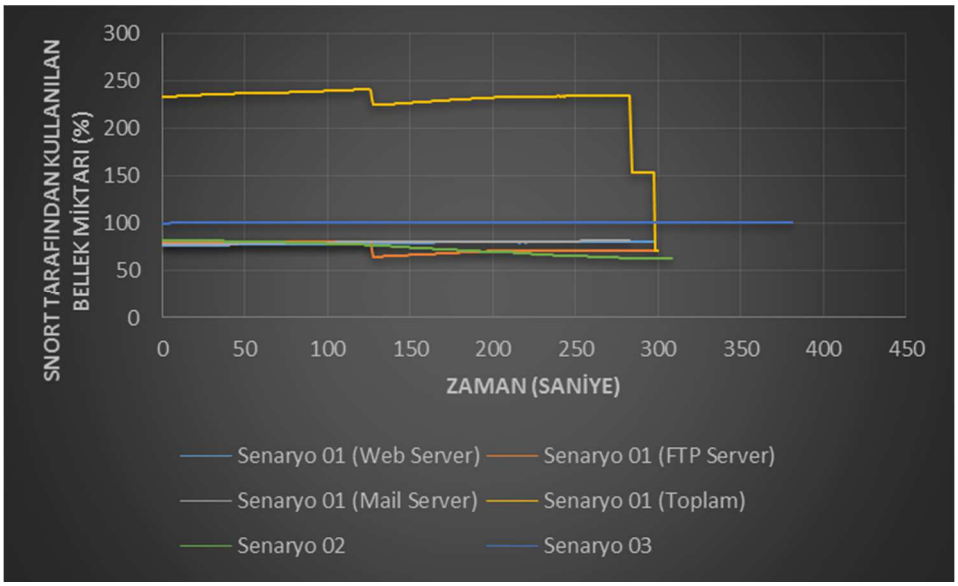
(b) ICMP Flooding saldırısında bellek kullanımı (2,000,000 paket)

Şekil A.4 : ICMP Flooding saldırısında donanım kullanımı (2m paket).

EK A



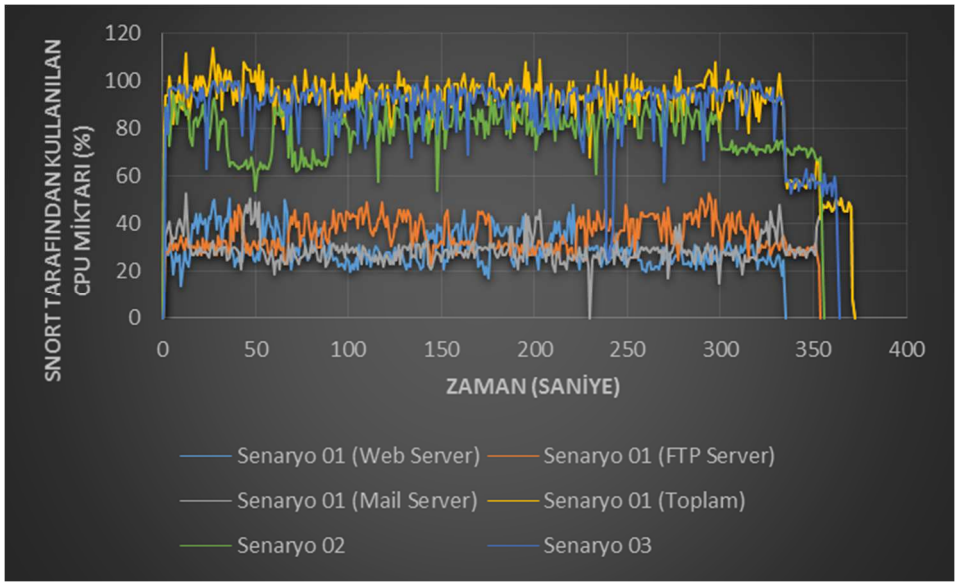
(a) TCP Flooding saldırısında CPU kullanımı (2,000,000 paket)



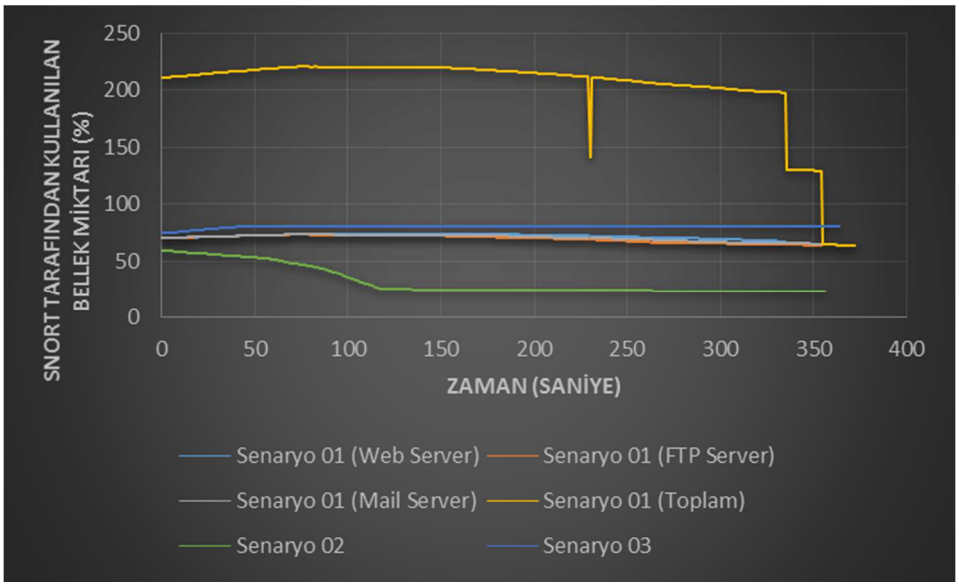
(b) TCP Flooding saldırısında bellek kullanımı (2,000,000 paket)

Şekil A.5 : TCP Flooding saldırısında donanım kullanımı (2m paket).

EK A



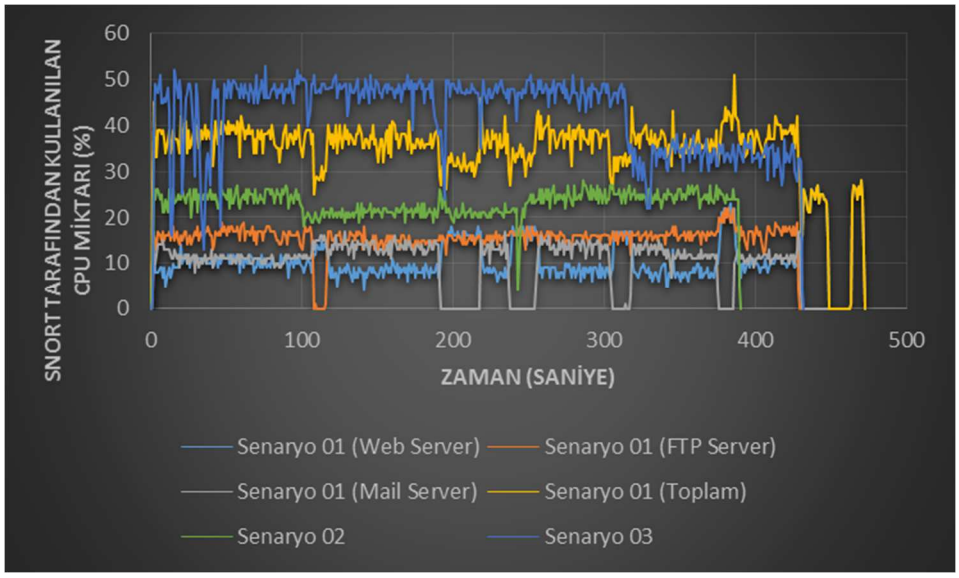
(a) UDP Flooding saldırısında CPU kullanımı (2,000,000 paket)



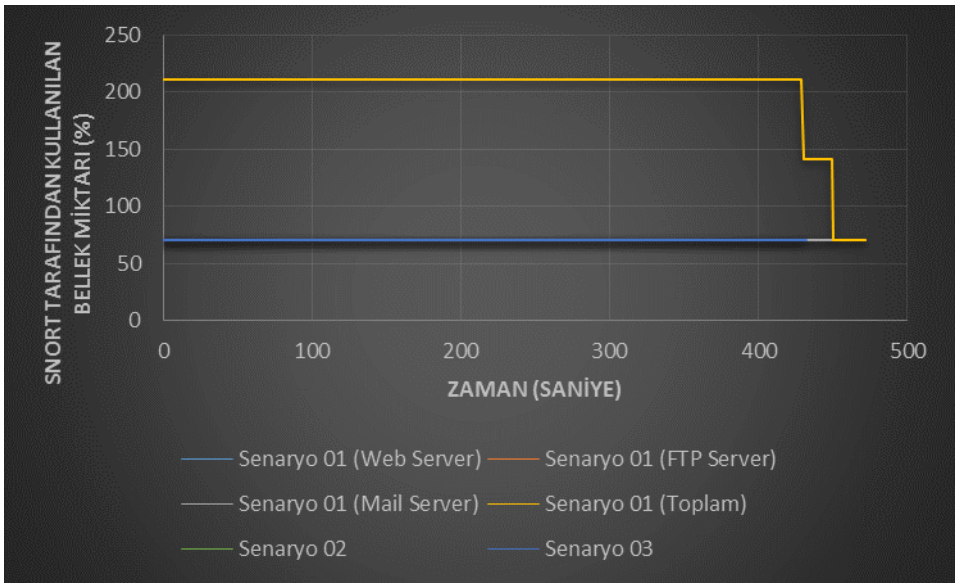
(b) UDP Flooding saldırısında bellek kullanımı (2,000,000 paket)

Şekil A.6 : UDP Flooding saldırısında donanım kullanımı (2m paket).

EK A



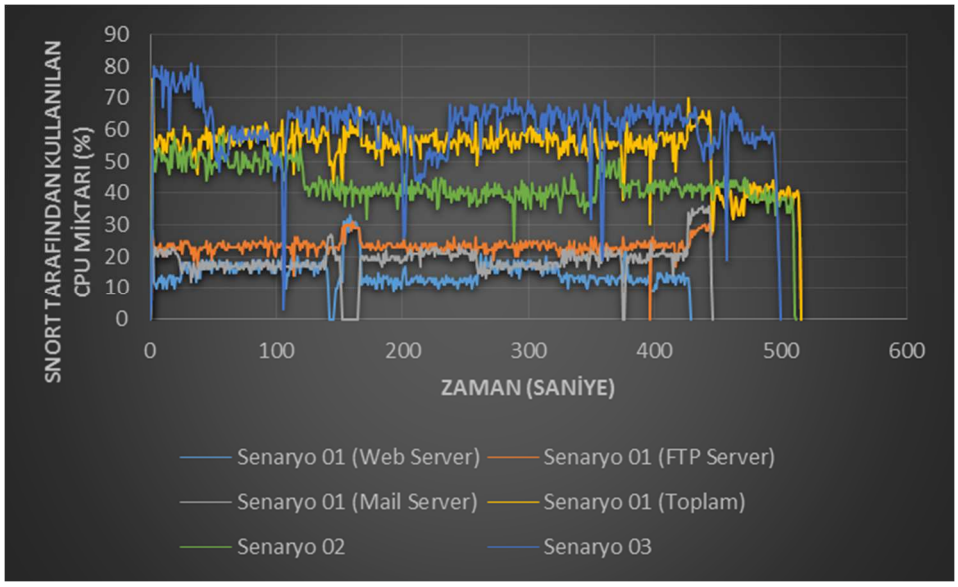
(a) ICMP Flooding saldırısında CPU kullanımı (3,000,000 paket)



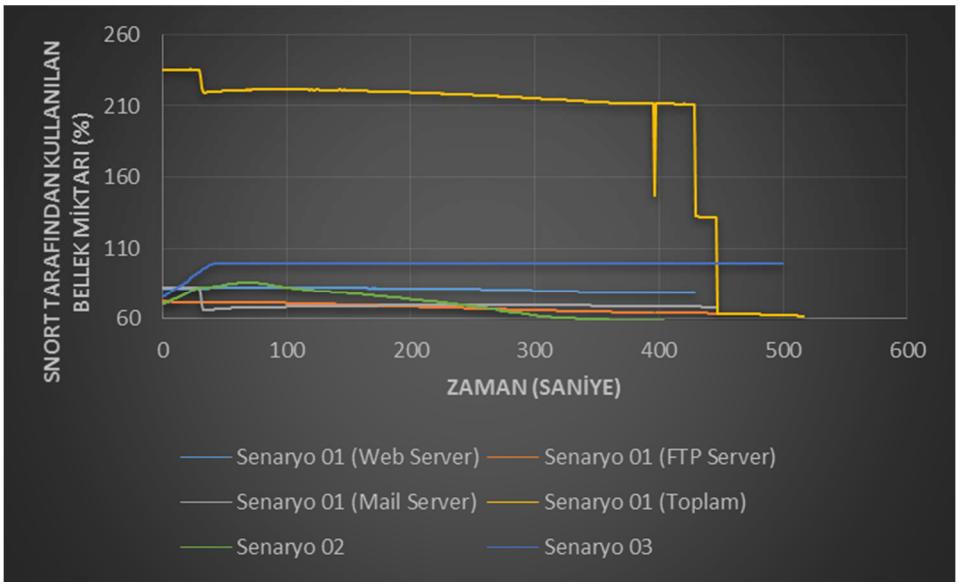
(b) ICMP Flooding saldırısında bellek kullanımı (3,000,000 paket)

Şekil A.7 : ICMP Flooding saldırısında donanım kullanımı (3m paket).

EK A



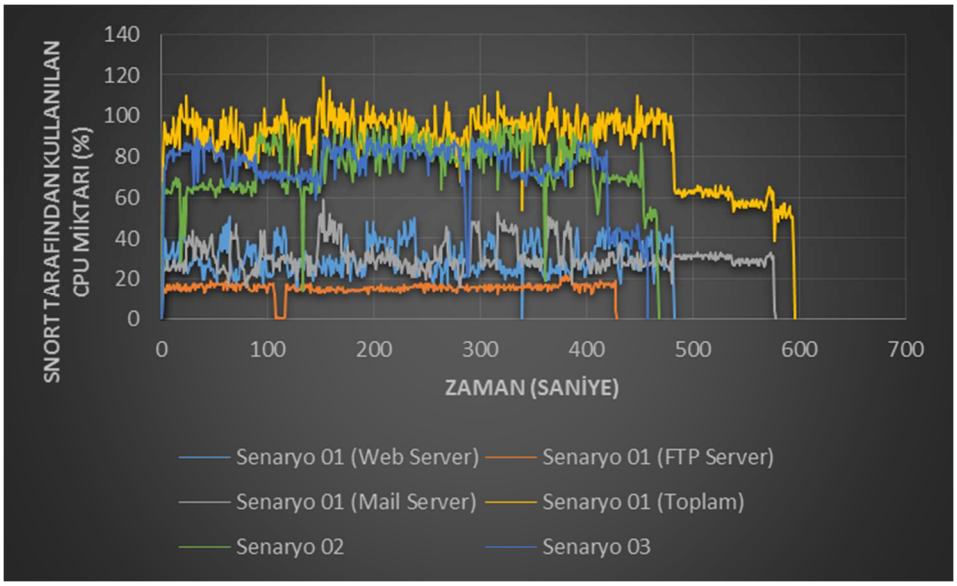
(a) TCP Flooding saldırısında CPU kullanımı (3,000,000 paket)



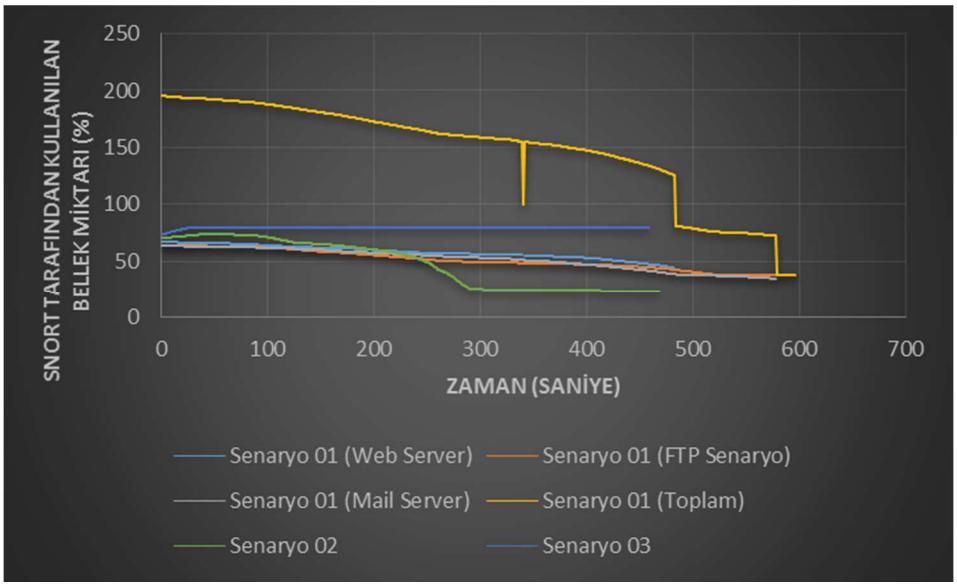
(b) TCP Flooding saldırısında bellek kullanımı (3,000,000 paket)

Şekil A.8 : TCP Flooding saldırısında donanım kullanımı (3m paket).

EK A



(a) UDP Flooding saldırısında CPU kullanımı (3,000,000 paket)



(b) UDP Flooding saldırısında bellek kullanımı (3,000,000 paket)

Şekil A.9 : UDP Flooding saldırısında donanım kullanımı (3m paket).

ÖZGEÇMİŞ

Uçman OKTAY 1984 yılında İstanbul'da doğdu. Orta öğretimini İzmir Karşıyaka Anadolu Lisesi'nde tamamladıktan sonra Hava Harp Okulu Bilgisayar Mühendisliği Bölümünden 2006 yılında mezun oldu. İleri seviyede İngilizce bilmektedir.

Ad Soyad **:** Uçman OKTAY

Doğum Yeri ve Tarihi **:** İstanbul / 11.03.1984

Adres **:** Hava Harp Okulu, Yeşilyurt / İstanbul

E-Posta **:** uoktay@hho.edu.tr

Lisans **:** Hava Harp Okulu Bilgisayar Mühendisliği

Yayın Listesi

- **Oktay U.,** Aydin M.A., Sahingoz O.K., Circular Chain VM Protection in AdjointVM. *The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE2013)*, May 9-11, 2013 Konya, Turkey.

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- **Oktay U.,** Sahingoz O.K., Proxy Network Intrusion Detection System for Cloud Computing. *The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE2013)*, May 9-11, 2013 Konya, Turkey.