BİLECİK ŞEYH EDEBALİ ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

SALDIRI TESPİT SİSTEMLERİNDE YAPAY SİNİR AĞLARININ KULLANIMI VE BAŞARIMLARININ İNCELENMESİ

Vedat MARTTİN
Yüksek Lisans Tezi

Tez Danışmanı
Yrd. Doç. Dr. Nazım İMAL

BİLECİK, 2014
Ref. No: 10038687

BİLECİK ŞEYH EDEBALİ ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

SALDIRI TESPİT SİSTEMLERİNDE YAPAY SİNİR AĞLARININ KULLANIMI VE BAŞARIMLARININ İNCELENMESİ

Vedat MARTTİN
Yüksek Lisans Tezi

Tez Danışmanı
Yrd. Doç. Dr. Nazım İMAL

BİLECİK, 2014

BILECIK SEYH EDEBALI UNIVERSITY
Graduate School of Science
Department of Computer Engineering

USING OF ARTIFICIAL NEURAL NETWORKS ON SYSTEMS OF INTRUSION DETECTION, AND INVESTIGATION OF IT'S ACHIEVEMENTS

Vedat MARTTİN
Master'sThesis

Thesis Advisor
Assist. Prof. Nazım İMAL

BILECIK, 2014

BİLECİK ŞEYH EDEBALİ ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ	**YÜKSEK LİSANS** **JÜRİ ONAY FORMU**

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 08.05.2014 tarih ve 20/1 sayılı kararıyla oluşturulan jüri tarafından 21.05.2014 tarihinde tez savunma sınavı yapılan Vedat MARTTİN' in "SALDIRI TESPİT SİSTEMLERİNDE YAPAY SİNİR AĞLARININ KULLANIMI VE BAŞARIMLARININ İNCELENMESİ" başlıklı tez çalışması Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak oy birliği/oy çokluğu ile kabul edilmiştir.

JÜRİ

ÜYE (TEZ DANIŞMANI) : Yrd.Doç.Dr.Nazım İMAL

ÜYE : Yrd.Doç.Dr. Uğur YÜZGEÇ

ÜYE : Yrd.Doç.Dr. Mehmet KOÇ

ONAY

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun .../.../..... tarih ve ...../..... sayılı kararı.

İMZA MÜHÜR

ÖZET

Bu çalışmada, önemi her geçen gün daha da artan ağ ve bilgi sistemlerine yapılan saldırıların analizi için, kullanılan saldırı tespit sistemlerinin (STS) türleri incelenmiş, bu doğrultuda Yapay Sinir Ağlarının (YSA) kullanılması araştırılmış ve örnek bir çalışma oluşturulmuştur.

STS'lerin oluşturulmasında KDD'99 (Knowledge Discovery and Data Mining Tools Competition) veri kümesi kullanılmıştır. Bu veri kümesinde belirli saldırı dosyaları kullanılarak, MATLAB programı üzerinde YSA kapsamlı çok katmanlı algılayıcılar (ÇKA) kullanan, örnek bir STS oluşturulmuştur. Geliştirilen model ile benzer konuda çalışacak yeni araştırmacılara önerilerde bulunulmuştur. Çalışmada eğitilen YSA ve KDD'99 tarafından oluşturulan örnek veri kümeleri ile testler yapılmış ve YSA'nın bilinmeyen saldırıları tespit ettiği gözlenmiştir. Saldırıların tespitinde ayrıca, YSA'nın normal veri akışı içine saklanan anormal veri akışını tespit ederek, kötüye kullanım saldırılarını başarılı bir şekilde sınıflayabildiği gözlenmiştir.

Anahtar sözcükler:yapay sinir ağları, çok katmanlı algılayıcı, saldırı tespit sistemleri, DoS, R2L, Probe saldırları, KDD'99.

ABSTRACT

In this study, in order to analysis of attacks to network and information systems that the importance of each passing day increased, examined species of used intrusion detection systems (IDS), in this direction use of Artificial Neural Networks (ANN) have been investigated and a case study has been created.

In forming of the IDS; KDD'99 (Knowledge Discovery and Data Mining Tools Competition) data set has been used. On this data set, using certain attack files, with ANN comprehensive multi-layer perceptron (MLP) on MATLAB program, a sample IDS has been created. With the developed model, suggestions have been made to the new researchers will work on a similar. In this study; tests made with data sets generated by the trained ANN and KDD'99 and has been observed detect of ANN that unknown attacks. Also in the detection of attacks, ANN detects to abnormal data flow in normally stored the data stream, grades abuse attacks succeeded in successfully.

Keywords:artificial neural networks, multi layer perceptron, intrusion detection system, DoS,R2L,Probe attacks, KDD’99.

TEŞEKKÜR

Bu çalışmanın yürütülmesi sırasında desteğini esirgemeyen danışmanım Yrd. Doç. Dr. Nazım İMAL'a, yardımlarıyla bana destek olan Yrd. Doç.Dr. Uğur YÜZGEÇ'e Bilecik Şeyh Edebali Üniversitesi Bilgi İşlem Daire Başkanlığı çalışma arkadaşlarıma, özellikle Daire Başkanı Murat FİDAN'a ve Öğr.Gör. Yusuf MUŞTU'ya, her türlü desteği ve anlayışı gösteren sevgili eşim Pakize Merve MARTTİN'e, yalnız bırakmayan ailelerimize ve çalışmam sırasında küçük veya büyük yardımını esirgemeyen herkese teşekkür ederim.

Vedat MARTTİN

Mayıs,2014

İÇİNDEKİLER

Sayfa No

ÇİZELGELER DİZİNİ

Sayfa No

ŞEKİLLER DİZİNİ

Sayfa No

SİMGELER VE KISALTMALAR DİZİNİ

ACK	Acknowledge
ADALINE	Adaptif/Uyumlu Doğrusal Nöron (Adaptive Linear Neuron)
ANN	Yapay Sinir Ağları (Artificial Neural Network)
ARP	Adres Çözümleme Protokolü
ARPANET	DARPA'nın oluşturduğu ilk bilgisayar ağı
BP	Geri Yayılım (Back Propagation)
ÇKA	Çok Katmanlı Algılayıcı
DARPA	Savunma İleri Araştırma Projeleri Teşkilatı (Defense Advanced Research Projects Agency)
DoS	Hizmet Engelleme (Denial of Service)
DF	Parçalama (Don't Fregment)
DVM	Destek Vektör Makinaları
FBI	Federal Araştırma Bürosu (Federal Bureau of Investigation)
FTP	Dosya Aktarım Protokolu (File Transfer Protocol)
HTTP	Bağlantılı Metin Aktarım Protokolu (Hypertext Transfer Protocol)
ICMP	İnternet Kontrol Mesajı Protokolü (Internet Control Message Protocol)
IDEVAL	Saldırı Tespiti Değerledirmesi Veri Seti (Intrusion Detection Evaluation Data Sets)
IDS	Saldırı Tespit Sistemi(Intrusion Detection Systems)
IMAP	İnternet Mesaj Erişim Protokolü(Internet Message Access Protocol)
IP	İnternet Protokolü (Internet Protocol)
IPSec	IP Security
IPv4	IP sürüm 4
IPv6	IP sürüm 6
KDD	Bilgi Keşfi ve Veri Madenciliği Aracı Yarışması (Knowledge Discovery and Data Mining Tools Competition)
LAN	Yerel Alan Ağı
LM	Levenberg-Marquardt
MAC	Media Access Controler
MATLAB	Mathworks firmasının bir ürünü

MF	Daha çok parça
MIT	Massachusetts Teknoloji Enstitüsü (Massachusetts Institute of Technology)
MLP	Çok Katmanlı Perseptron (Multilayer Perceptron)
NAT	Network Address Translator
NBO	Normal ağ trafiğinin başarım oranı (%),
NCP	Network Control Protocol
NMAP	Açık kaynaklı ağ haritalama programı
NN	Neural Networks
ORT	YSA çıktılarının aritmetik ortalaması
OS	İşletim Sistemi (Operating Sistem)
Pod	The ping of death
R2L	Uzaktan Yerele (Remote to Local)
RFC	Request For Comments
SBBO	Saldırı bulma başarım oranı (%),
SMTP	Basit Posta Gönderme Protokolü (Simple Mail Transfer Protocol)
SOM	Kendini Örgütleyen Haritalar (Self-Organizing Maps)
SRI	Stanford Research Institute
STS	Saldırı Tespit Sistemleri (Intrusion Detection Sistems)
SVM	Support Vector Machines
SYN	Senkronize (Synchronous)
TCP/IP	Transmission Control Protocol / Internet Protocol
TKA	Tek Katmanlı Algılayıcı
TÜBİTAK	Türkiye Bilim ve Teknoloji Kurumu
U2L	Kullanıcıdan Yerele (User to Local)
U2R	Kullanıcıdan Yöneticiye (User to Root)
UCLA	University of California at Los Angeles
UCSB	University of California at Santa Barbara
UDP	Kullanıcı Veri Birimi Protokolü (User Datagram Protocol)
ULAKNET	Ulusal Akademik Ağı
YSA	Yapay Sinir Ağları (Artificial Neural Networks)

1. GİRİŞ

1.1 Giriş

Günümüz teknoloji dünyasında önemli bir yer tutan bilgi ve bilgisayar sistemleri, internetin de sağladığı küresel ölçüler içerisinde yeni boyutlara ulaşmıştır. İnsanlar, işlerini büyük ölçüde kolaylaştıran bilgi teknolojilerinden yararlanırken güvenlikkonusunu da göz önünde bulundurmalıdır.

Bilgilerin dijital veri haline dönüştürülmesiyle dijital verilerin kıymetli bir hal alması, veri güvenliği konusunu meydana getirmiştir. Birçok kıymetli bilgi dijital veritabanlarında erişime hazır bekletilir olduğundan bu verilerin güvenliklerinin sağlanılması zorunluluğunu oluşturmuştur. Gerek veritabanlarında, gerekse bu veritabanına erişen yazılımlar yeterince korunmadığında saldırganların hedefi olacağı açıktır. Bu durumda saldırganlar çok daha elverişli bir hedefte bulunurlarsa siber saldırılara uğramak kaçınılmaz olacaktır. Bilgi teknolojilerininyoğun kullanılmasıyla birlikte bilgi güvenliği de önemli bir yer tutmaktadır. Günümüzde kullanıcıların sosyal medya hesaplarından,devletlerin sırlarına bazı web sayfalarında paylaşıldığı görülmektedir. Diğer taraftan bazı kişiler,çeşitli sebeplerle kurumsal yada özel web sitelerine habersiz sızmayoluyla(hacking) ya da mevcut sistemin zafiyetlerini araştırarak kendisi veya köle bilgisayarlar (zombi) kullanarak müdahale etmektedir. Yapılan işin nedenlerine bakıldığında; yeni bilgisayar kullanıcısının merakı ve kendini ispat etmesi, finansal kaynakları kullanabilme veya kendilerine çıkar sağlama, devletlerin sırlarını ele geçirme,sanal saldırı diğer bir tabirle siber saldırı yaparak kurumları ve hükümetleri zafiyete uğratma ya da çökertme, siber saldırıyla psikolojik harp teknikleriyle kurumları –hükümetleri etkileme olarak sıralanabilmektedir.

Saldırının olduğu yerde saldırıya karşı savunma da olacağından meydana getirilen saldırıların boyutlarına göre gerekli önlemleri almak gereklidir. Saldırının türüne göre tespit etme ve önlem almak adına saldırı tespit sistemleri(STS-IDS) ve saldırı önleme sistemleri (SÖS-IPS)geliştirilmiştir.

Yapılan çalışmada, KDD’99 saldırı veri kümesikullanılarak, bazı saldırı veri kümeleri MATLAB programı ile işlenerek yapay sinir ağları(YSA) kullanılarak

mevcutsistemin bu saldırıları tanıyabilmesi ya da ayırt edebilmesi gözlemlenerek analizler gerçekleştirilmiştir.

Gerçekleştirilen bu analizlerin veri ve bilgisayar güvenliği açısından faydalı sonuçlara ulaşma açısından araştırmacılara yardımcı olduğu düşünülmektedir.

1.2 Literatür Çalışması

YSA ile saldırı tespit sistemleri uygulamaları Cannady(1998)'nin çalışmalarıyla ortaya çıkmıştır. Buçalışmada Cannady(1998) saldırı tespitinde YSAyapısının nasıl kullanılacağı hakkında metotlar sunulmuş ve YSA kullanımının başarılı ve başarısız olduğu durumlar belirtilmiştir. Ağ temelli saldırı tespit sistemlerinin YSA ile kullanılabildiği ilk ciddi çalışmadır (Tanrıkulu, 2009).

Ryan vd.(1998) de yaptığı çalışmada bir YSA 'yı geri yayılım algoritmasıyla eğiterek ağ üzerindeki kullanıcıların bilgisayarındaki işletim sistemlerini tespit etmiştir. Ağa bağlı kullanıcıların yazdıkları komutlar izlenerek kötü ya da yanlış kullanımdan meydana gelen saldırılar kaydedilmiştir. Unix işletim sisteminde çalışan bu yazılım sinir ağı tespit sistemi (NNID) olarak da bilinmektedir.

Lipmann vd.(1999) çalışmasında kullandığı YSA'larını geri yayılım algoritmasıyla Ryan (1998) gibi eğiterek kötüye kullanım saldırı tespitinin bulunması amacıyla kullanılmıştır.

Ghost vd.(1999) çalışmasında program davranış özelliklerini çözümleyen bir YSA oluşturmuştur. Bu metotta, programların normal sistem ile saldırı durumunda davranışları karşılaştırılmıştır. Çalışma sonucu olarak olumsuz kullanım ve anormallik tespitinin beraber gerçekleştirildiği söylenebilir.

. Mukkamala vd.(2002)çalışmalarında KDD'99 veri kümesini kullanarak ÇKA ile beraber destek vektör makinesini (DVM) birlikte kullanarakhibrid bir yapı tasarlayarak başarılı sonuçlar elde etmiştir.

Lichodzijewski vd.(2002), yaptığı çalışmada DARPA veri kümelerini sadeleştirerek, danışmansız öğrenme yöntemlerinden biri olan kendini örgütleyen haritalara(SOM-Self Organizing Map) göre veri kümesinin kullanımı açıklamıştır.

Zanero(2004) çalışmasında, STS meydana getirmede SOM kullanmış ve başarılı sonuçlar almıştır. Moradi ve Zulkernine (2004), STS uygulamalarında çevrimdışı ağ kullanarak ÇKAyapısının saldırı sınıflandırmasında YSA ile başarılı sonuçlar alınabileceğini göstermiştir.

Erol (2005) yaptığı çalışmada STS'lerin detaylı bir sınıflandırmasını incelemiş ve kullanılan veri kümelerininsonuçlarını tüm yönleriyle sunmuştur.

Sammany(2007) DARPA'99 veri kümesinde ÇKA yapısında farklı katman ve daha az özellikli veri kümeleri kullanarakçalışma yapmıştır.

Öksüz(2007), Zanero(2004) ile benzer bir çalışma gerçekleştirerek, SOM kullanan STS'lerin başarılı olduğunu göstermiştir.

Güven(2007), çalışmasında Erol(2005) ile benzer şekilde STS'leri detaylısınıflandırılması ve KDD'99 veri kümesinin meydana gelmesinde kullanılan yöntemin benzerini kullanmıştır. Gerçekleştirdiği test ortamlarında,belli bir takvim dâhilindeinternetten topladıkları verileri kullanarak ayrı bir zeki saldırı tespit sistemi tasarısı sunmuşlardır.

Tanrıkulu (2009), DARPA eğitim veri kümesi ve internet ortamından toplanılan veri paketlerinin belirli ölçülerde sadeleştirilmesiyle, çok katmanlı bir YSA yapısında eğitimtestleri gerçekleştirerek, sonuçlarını karşılaştırmıştır.

Bu tezde gerçekleştirilen çalışmada, KDD'99 veri kümesinin %10'luk kısmında rastgele seçilerek oluşturulan eğitim ve test veri kümeleri YSA ile eğitilmiş ve eğitim sonrası benzer saldırılarda savunma başarım oranlarının tespiti gerçekleştirilmiştir.

1.3 Çalışmanın Kapsamı

Çalışmanın kapsamı dâhilinde testlerin yapıldığı araçların ve ortamların etkili olduğu düşünülerek çalışmanın alt ve üst sınırları belirlenmiştir.Çalışmanın gerçekleştirildiği bilgisayar, 2.4GHz işlemci hızında Intel i3 işlemcili, 3GB Ram bellek ve 64 bitlik Windows işletim sistemine sahiptir.

Veri kümeleri meydana getirilirken, KDD'99veri kümesinin %10'luk kısmı,

açıkkaynaklı bir kelime işlemci editöründe düzenlenerek, ifadeler sayısallaştırılmıştır. Oluşturulan eğitim ve test kümeleri, MATLAB programında sinir ağı aracı(NN-Neural Network Toolbox) kullanılarak işlenmiştir. Bu çalışma, çok katmanlı ağlarda geri yayılım algoritmasının bir türü olan Levenberg-Marquardt (LM) algoritması uygulanarak gerçekleştirilmiştir.

Tez kapsamında, birinci bölümde giriş ve önceki yapılan çalışmalar hakkında bilgiler, gerçekleştirilen çalışmanın kapsamı ve kullanılan yöntem hakkında bilgiler verilmiştir.

İkinci bölümde bilgi ve bilgi güvenliği, internet ve ülkemizde internet kullanımı hakkında ve internetin omurgası olan TCP/IP protokolü hakkında bilgiler verilmiştir.

Üçüncü bölümde saldırı teriminden bahsedilmiş, saldırı türlerinin sınıflandırılması, saldırı tespitinde kullanılan yöntemler ve çalışmada kullanılan saldırı türleri hakkında bilgiler verilmiştir.

Dördüncü bölümde YSA'nın yapısı, işleyişi, türleri ve kullanılan fonksiyonlar hakkında bilgiler verilmiştir.

Beşinci bölümde STS çalışmalarında kullanılan ve genel kabul görmüş saldırı veri kümeleri hakkında bilgiler verilerek çalışmada kullanılan veri kümesinin oluşturulması sunulmuştur.

Altıncı bölümde meydana getirilen veri kümesi MATLAB programı ile işlemeye uygun hale getirilmesi açıklanmıştır. Kullanılan yöntem burada detaylı olarak ele alınarak, yapılanuygulamada ekran görüntüleri ve grafik yorumları aktarılmıştır.

Yedinci bölümde yapılan çalışmanın sonuçları ve başarımları çıkartılmış ve geçmişte yapılan çalışmalarla karşılaştırılması sunulmuştur.Çalışma esnasında karşılaşılan zorluklardan bahsedilerek araştırmacılara önerilerde bulunulmuştur.

1.4 Çalışmanın Yöntemi

Bu çalışmada, KDD'99 veri kümesinin bir parçası olan %10'luk veri kümesi kullanılarak, eğitim için ele alınan tüm saldırılardan örneklemeleroluşturulmuştur. Bu örneklemelerden rastgele seçilerek meydana getirilen 3000 örnekli saldırı dosyası ve 2781 örnekli saldırı dosyaları kullanılmıştır. Test verisi olarak ise, rastgele seçilmiş200 örnekli ve 222 örnekli saldırı dosyaları kullanılmıştır.

Saldırı tespitinde YSA'ları eğitirken, Tanrıkulu(2009)'nun çalışmasında kullandığı "Doğrudan eğitim" ile "Ayrı ayrı eğitim" yöntemleri bu çalışmada farklı saldırı küme ve örneklemeler için kullanılmıştır. Doğrudan eğitim kapsamında YSA'lar ile atak3000.txt ve atak2781.txt dosyaları eğitilerek, içerisinde tespit edilmesi istenen saldırıların bulunduğu ornek200.txt ve ornek222.txt dosyaları işlenmiştir. Başarım sonuçları kaydedilerek,ayrı eğitim kapsamında saldırının örneklemesi yapılan Normal Imap-Pod ve Nmap saldırı dosyaları YSA'ları ile eğitilmiştir. Test veri kümeleriolan ornek200.txt ve ornek222.txt dosyalarındaki saldırılar tespit edilmeye çalışılmıştır.Başarım sonuçları kaydedilerek çalışma sonuçları değerlendirilmiştir.

2. BİLGİ VE BİLGİ GÜVENLİĞİ

2.1 Bilgi ve Bilgi Güvenliği

Bilgi kelime anlamı olarak "İnsan aklının erebileceği olgu, gerçek ve ilkelerin bütünü, bili, malumat" ya da " İnsan zekâsının çalışması sonucu ortaya çıkan düşünce ürünü, malumat, vukuf"[1]olarak geçmektedir. Bunun gibi birkaç tane daha birbiriyle karışan ya da karıştırılan kavramlar mevcuttur. Örneğin veri kelimesi "Bir araştırmanın, bir tartışmanın, bir muhakemenin temeli olan ana öğe, muta, done" olarak; bilişimde kullanılananlam olarak ise "Olgu, kavram veya komutların, iletişim, yorum ve işlem için elverişli biçimli gösterimi"[2] olarak karşılık bulmaktadır.

Günümüzde bilgiyi elde etmek kadar işlemek ve muhafaza etmek önemlidir. Bilginin artan değer olarak kabul gördüğü bu dönemde şirketlerin değerleri sahip oldukları mal varlıklarıyla değil, sahip oldukları yetişmiş insan gücü, var olan müşteri potansiyeli ve kendi yarattıkları bilgi birikimleriyle ölçülmektedir(Barutçugil, 2002). Buna en çarpıcıörnek 1998 yılında 25 milyondolarsermayeile kurulanve internetüzerinde arama hizmeti sunan Google Inc. 'in 2013 yılının son çeyreğinde piyasa değerinin110,92 Trilyon dolara ulaşmış olmasıdır[3].Google 'ınböylesine yüksek piyasa değerine ulaşması büyük fabrikalar ve yükseksayıdaki insan gücüne ihtiyacı duymasından değil, Google'ın kullandığı tek şey yenilikçi ve ihtiyaçları öngörerek yaratıcı düşünceye sahip olmasıdır.Diğer bir örnek ise teknolojisi sektörünün öndegelen beş büyük firması; Microsoft, Intel, Compaq, Dell ve Cisco, 1987 -1997 yıllarıarasındaki on yıllık dönemde 12 Milyar dolarlık piyasa değerini 588 Milyar dolaraçıkarmışlardır(Barutçugil, 2002).Bilgi yalnızca ekonomik bir değer olmakla kalmayıp, çağımıza ve toplumlaraşekil veren yegâne unsur olmuştur.Geçmişe baktığımız zaman ülkelerin gelişmişlikdüzeyinin, sanayi potansiyeli ve enerji tüketimi

---

[1]http://www. tdk. gov. tr/index. php?option=com_gts&arama=gts&guid=TDK. GTS. 53309fefc8b666. 01980246(03.03.2014)

[2]http://www. tdk. gov. tr/index. php?option=com_gts&arama=gts&guid=TDK. GTS. 53309ff64fd6a0. 11110218(03.03.2014)

[3]en. wikipedia. org/wiki/Google(03.03.2014)

ile belirlendiği görülmekte iken, bilişimtoplumu olarak adlandırılan bu döneme baktığımız zaman ise ülkelerin gelişmişlikdüzeyi ürettiği, işlediği bilginin miktarı ve bilişim ürünlerinin kullanımı ile belirlenmektedir.

Bilişim toplumunda bilginin kullanımı arttıkça üretim yapısı dadeğişmekte; bilgi, işgücü ve sermayeden de önemli bir faktör olarak üretime girmektedir(Canlı, 2009).

Bilgi güvenliği kavramının ise kısaca üç temel öğeden meydana geldiğini söylenebilir. Bunlar; gizlilik(confidentiality), bütünlük (integrity)ve kullanılabilirlik (availability)tir.

- **Gizlilik:** Bilginin yetkisizkisilerin erisimine kapalı olması ya da bilginin yetkisizkisilerce açığa çıkarılmasının engellenmesidir.

- **Bütünlük:** Bilginin yetkisiz kişilerce değiştirilmesi, silinmesi ya da herhangi bir şekilde tahrip edilmesi tehditlerine karşı içeriğinin korunmasıdır.

- **Kullanılabilirlik:** Bilginin her ihtiyaç duyulduğundaerişilebilir olması ya da kullanıma hazır durumda olmasıdır(Marttin ve Pehlivan, 2010).

2.2 İnternet

Bu bölümde, internet oluşumu ve gelişimi ile ülkemizde internetin kullanılmaya başlaması ve ULAKNET hakkında bilgiler verilmektedir.

2.2.1 İnternetin kısa tarihçesi

İnternet, DARPA (İleri Savunma Araştırma Projesi) adında bir geliştirme projesi kapsamında 1969 yılında ARPANET ismiyle hayata geçirilmiştir. Oluşturulan bu yapıya ABD deki dört üniversite UCLA(University of California at Los Angeles), SRI(Stanford Research Institute), UU(University of Utah) ve son olarak UCSB(University of California at Santa Barbara) birbirine bağlanarak İnternet' ağının ilk şekliortaya çıkmıştır.

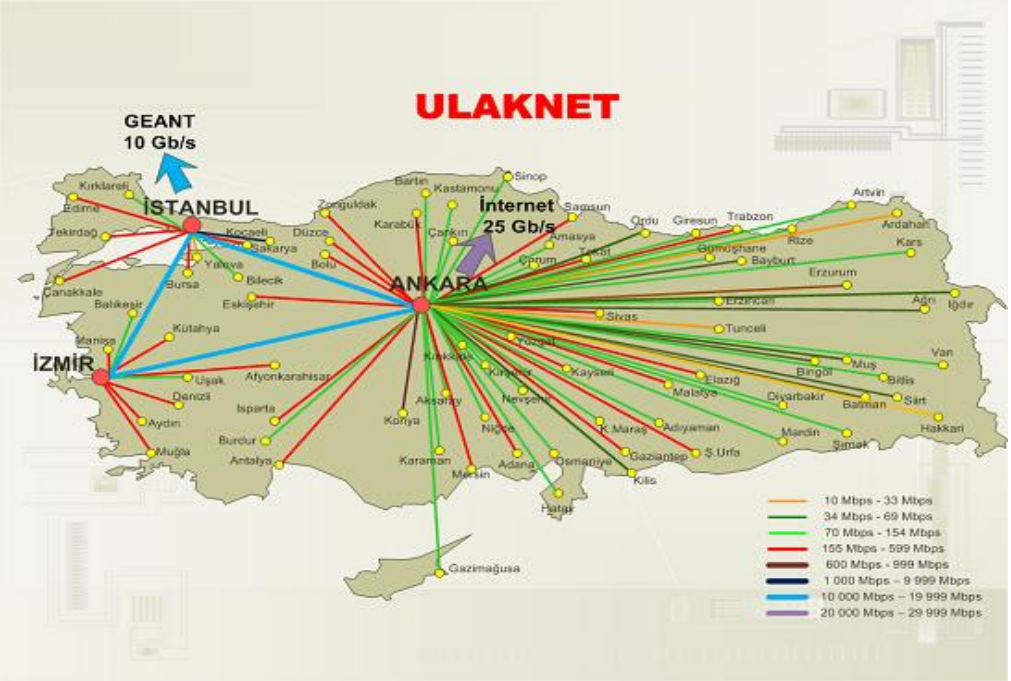
Teknoloji bilgiyi işlemede, bilgiye erişimde, insanların birbiriyle iletişimi ve bilgi paylaşımında da büyük kolaylıklarsağlamıştır.1990'ların başından günümüze kadar gelen süreçte hızla gelişen internet, insanlara bilgiye kolay ve çabuk erişim imkanı sağlamıştır Bu durum başdöndürücü birhızla büyümektedir(Canlı, 2009).

2.2.2 Ülkemizde İnternet ve ULAKNET

Ülkemizin internet ağının dahil olması 1993 yılında ODTÜ (Ortadoğu Teknik Üniversitesi) de gerçekleşmiştir.64Kbit/sn hızında olan bu hat, çok uzun bir süre, tüm ülkenin internete tek çıkışı olmuştur. Sonrasında internet öncelikle akademik ortamlardaolmak üzere tüm Türkiye'de yaygınlaştırmaya başlamıştır. Ege Üniversitesi'nden olan bağlantı, 1994 yılı başlarında 64Kbit/snhızı ile gerçekleştirilmiştir. Ardından sırasıyla, Bilkent Üniversitesi (Ekim-1995), Boğaziçi Üniversitesi (Kasım -1995) ve İTÜ(İstanbul Teknik Üniversitesi) (Şubat-1996) bağlantıları gerçekleştirilmiştir. 1996 yılı Ağustos ayında da Turnet çalışmaya başlamıştır.

1997 yılına gelindiğinde, akademik kuruluşların internet bağlantısını sağlayan TÜBİTAK kapsamındaki ULAKNET ağıyla çalışmaya başlamış ve üniversiteler daha hızlı bir omurga yapısıyla birbirlerine bağlanmıştır. 1999 yılı içerisinde, ticari ağ altyapısında büyük değişiklikler olmuş ve Turnet'in yerini TTNet adında yeni bir oluşum almıştır. 2000'lerin başında; ticari kullanıcılar TTNet omurgası üzerinden; akademik kuruluşlar ve ilgili birimler de ULAKNET omurgası üzerinden internet erişimine sahiptir[1]. Ayrıca bu iki omurga arasında yüksek hızlı bağlantı mevcuttur. ULAKNET ağ yapısı Şekil 2.1'de gösterilmiştir.

[1]http://yunus. hacettepe. edu. tr/~sadi/dersler/ebb/ebb467-guz2000/umut-p. Html (03.04.2014)



Şekil 2.1. Ülkemizdeki ULAKNET ağ yapısı [1].

Günümüzde ULAKNET ağında internet hızı 25Gb/sn ye kadar ulaşmıştır. ULAKNET hizmetinden faydalanan kurumlardan bazıları aşağıda sıralanmıştır:

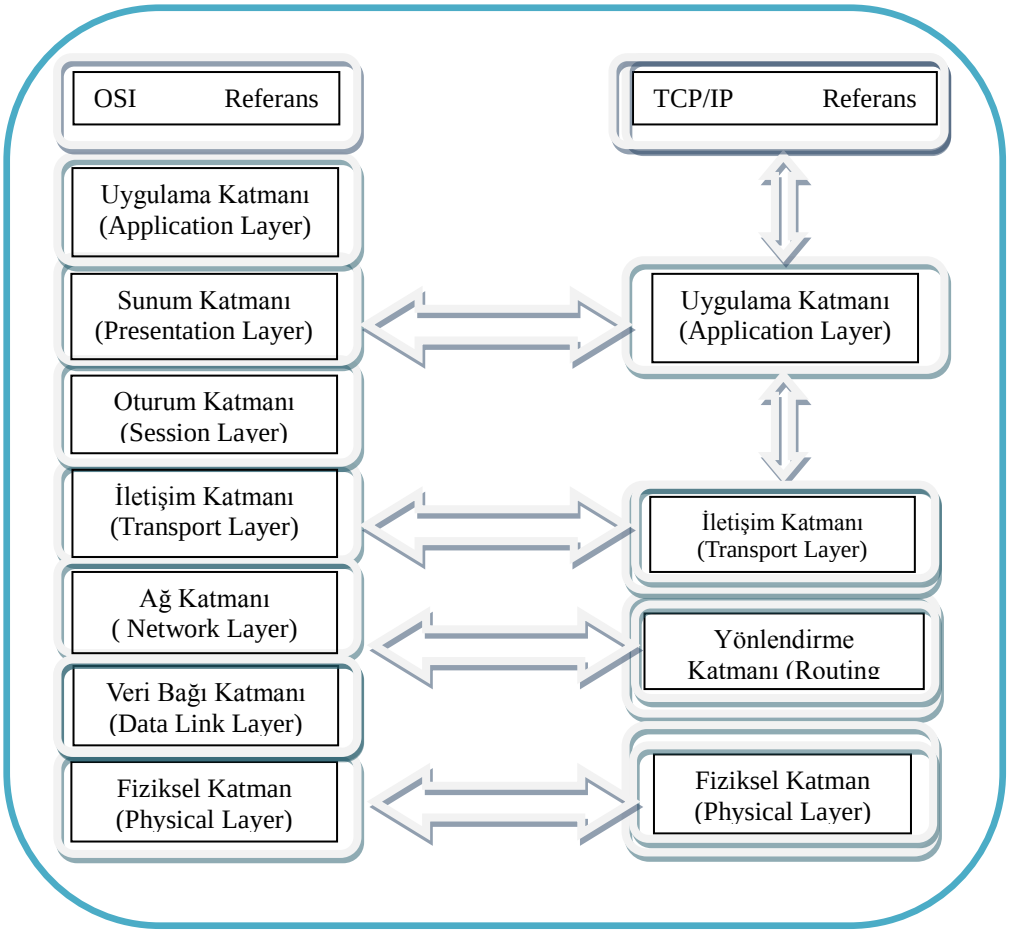
- Türkiye'deki tüm üniversiteler ile bunların fakülte ve diğer alt birimleri,
- TÜBİTAK birimleri,
- Askeri Okullar, Harp Akademileri ve Polis Akademileri,
- Türk Silahlı Kuvvetleri'nin Ar-Ge birimleri,
- Atatürk Kültür Dil ve Tarih Yüksek Kurumu,
- Milli Kütüphane,
- YÖK (Yüksek Öğretim Kurumu)
- ÖSYM (Ölçme Seçme ve Yerleştirme Merkezi)
- Türkiye Atom Enerjisi Kurumu
- Afet ve Acil Durum Yönetimi Başkanlığı kurumlarından oluşan toplam 204 birime Metro Ethernet hatları kullanılarak hizmet sağlamaktadır.

[1]http://ulakbim. tubitak. gov. tr/tr/hizmetlerimiz/ulusal-akademik-ag-0 (03.04.2014)

2.3 TCP/IP (Transmission Control Protocol / Internet Protocol)

İletişimde iki tarafın karşılıklı konuşması, iletişime geçmesi, iletişimisürdürmesi ve iletişimi sonlandırması için oluşturulan belli kurallar vardır. Karşılıklı mutabakat sağlanan kural ve politikaların tümüne "protokol" adı verilmektedir. İnternetin markalardan bağımsız olarak işleyebilmesi için protokoller kümesi oluşturulmuştur.İnternetin konuşma, anlaşma protokollerinden en çok kullanılanlarından birisi de TCP/IP protokolüdür. Kısaca TCP/IP'ye "internetin omurgası"denmektedir(Yıldırımoğlu, 2012).

İletişimde belli referans modeller tanımlanarak işleyiş katmanlara bölünerek düzenlenmiştir. Bunlardan birisi OSI diğeri TCP/IP referans modelidir.OSI referans modeli yedikatmandan oluşurken TCP/IP protokol ailesi dört katmandan oluşmaktadır.OSImodelindeki yedi katmanda yapılan tüm işlere karşılık, TCP/IP'de dört katmanda bu işler toplanmıştır.



Şekil 2.2. OSI ve TCP/IP modeli.

OSI Referans modelinde fiziksel ve veri bağı katmanı TCP/IP referans modelinde fiziksel katmana, ağ katmanı ise yönlendirme katmanına karşılık gelmektedir. Uygulama, Sunum ve Oturum katmanları ise TCP/IP de Uygulama katmanına karşılık gelmektedir.Bu kısımlar ele alınırsa:

- Fiziksel katmanda elektriksel sinyalleşmelerin yapıldığı ve MAC adreslerinin bulunduğu kısımdır.
- Ağ katmanı ve yönlendirme katmanı elektriksel sinyallerin ağa uygun şekilde işlenmesi ve yönlendirilmesi ile ilgili kısımdır.
- İletişim katmanı uygulama katmanı için gerekli hazırlıkların yapıldığı ara katmandır.
- Uygulama katmanı iletişim katmanından sonra son kullanıcının kullandığı programları ve programların işletildiği protokolleri kapsayan kısımdır.

Fiziksel katman dışında her katman kendisine özgü altprotokollerden oluşmaktadır. Herkatman farklı protokolleri desteklemektedir ve bu protokoller RFC (Request ForComments) başlığı altında çok sayıda dökümanda açıklanmıştır[1]. Örneğin TCP/IP protokol kümesindeki TCP'i açıklamak için RFC 793 sayılı doküman hazırlanmıştır.Benzerşekilde RFC 959 sayılı doküman FTP'yi (dosya transfer protokolü) tanımlamıştır.Paket iletim garantisi olmayan iletim protokolü UDP (kullanıcı veri paketi protokolü)ise RFC 768 sayılı doküman ile tanımlanmaktadır.TCP/IP protokolüzamanın gereksinimlerine göre yeni RFC'ler ile güncellenmektedir.

TCP/IP protokol kümesinde IP, ikinci katman olan Yönlendirme katmanında çalışmaktadır. Örneğin yine bu katmanda çalışan adres çözümleme protokolü (ARP) yerel alan ağı (LAN) içinde IP adresi bilinen alıcı bilgisayarın fiziksel adresini (MAC adresini) bulmak için kullanılan bir protokoldür. TCP üçüncü katman olan iletim katmanındaçalışmaktadır. Uygulama katmanında ise dosya transfer protokolü (FTP), dinamik isimsunucusu (DNS) ve uzaktan erişimi sağlayan TELNET gibi protokoller çalışmaktadır. İnternetin veri iletişimi IP paketleri ile yapılmaktadır.İnternet üzerindeki bilgisayarlar, TCP/IP'nin mimarisi gereği kullanıcı/sunucu

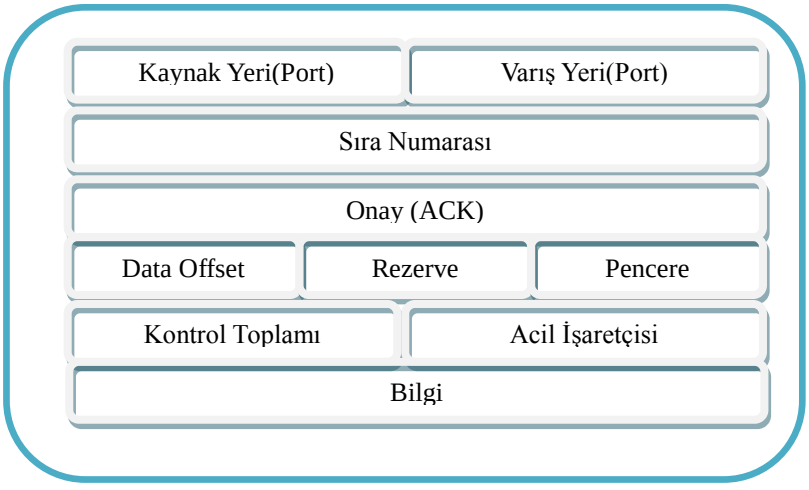
[1]http://www. faqs. org/rfcs/ (03.03.2014)

şeklindeçalışmaktadır.İletişimde olan iki bilgisayardan biri hizmet sunarken, diğeri hizmet alanpozisyonunda çalışmaktadır.

2.3.1 İletişim kontrol protokolü (TCP) katmanı

TCP, bir paketin bütünlüğünü ve iletişim garantisini sağlayan protokoldür. Uygulama katmanından gelen bilgileri yığınlar (segment) haline getirmek, veri bütünlüğünü koruyarak veri paketinin içeriğinin değiştirilmeden iletilmesini sağlamak ve iletişim esnasında paketi yerine ulaştırma işini garantilemeyi sağlayan, kaybolan bilgileri tekrar göndermekgibi bir çok işin yapıldığı protokolkatmanıdır (Tanrıkulu, 2009).

TCP katmanında her yığının başında paket sıra numarası ve kapı (port) bilgileri barındıran TCP başlığı bulunmaktadır. Kapı numaraları kaynak ve varışbilgisayarlarına ait kapı numaralarıdır.Sıra numarası ise TCP büyük veri paketlerinin parçalandıktan sonra karşı tarafta hangi sıraya göre sıralanması gerektiğinigöstermektedir. Şekil 2.3'de gösterildiği gibi TCP yığın yapısında kontrol toplamı sayesinde parçalanan veri paketleri yığın içindeki tüm verilerin matematiksel olarak toplanmakta ve bu verinin TCP başlığına yazılmaktadır. Alıcı tarafından veri geldiğinde, tekrar toplamı alınarak bu toplam kontrol toplamında yazılan değer ile karşılaştırılmaktadır.Toplam sonucu alınan değer aynı değil ise paket gönderim esnasında "veri yolda bozulmuştur" ya da "paket kaybı" olmuştur denilmektedir.



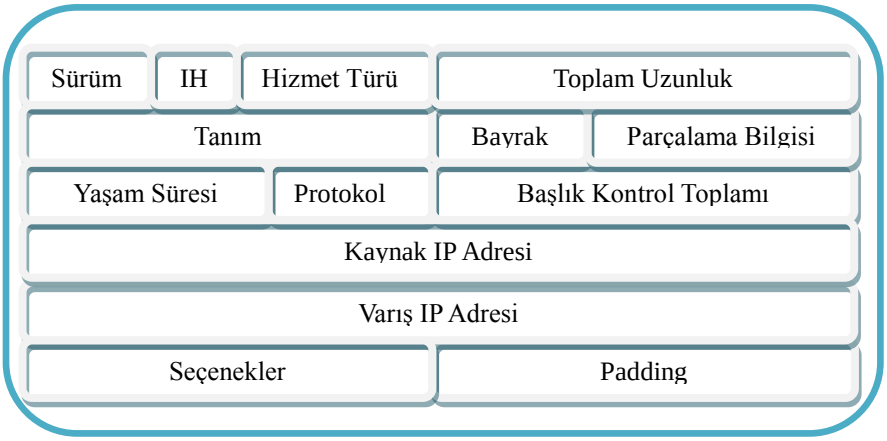
Şekil 2.3. TCP Protokolü yığın yapısı.

2.3.2 İnternet protokolü (IP)

İnternet,çok sayıda ağın birbirine bağlı olduğu ağlar topluluğudur ya da diğer bir anlamda dünya ağıdır. Ağüzerinde yüksek bant genişlikli hatlardan ve hızlı yönlendiricilerden oluşan bir diziomurga bulunmaktadır.Bu omurgalara bölgesel ve ulusal ağlar bağlanmıştır.İnternetiçindeki ağları ve bilgisayarları birbirlerine bağlayan, iletişim kurmalarını sağlayanprotokol IP'dir. Bu nedenle IP internetin ortak dilidir.IP, ağ içerisindeki paketlerin adreslenerek iletişimi ve yönlendirilmesinden sorumludur (Tanrıkulu, 2009).

2.3.2.1 İnternet protokolü (IP) paket yapısı

IP paket yapısı Şekil 2.4' de görüldüğü üzere erişim bilgileri, kullanılan protokol bilgisi ve kontrol bilgileri gibi verilerden içermektedir.



Şekil 2.4. IP paket yapısı.

- **Sürüm :** Paketin IP sürümü

- **IHL (IP Başlık Boyu - IP Header Lenght):** Başlık alanının kaç adet 32 bitlik sözcükten oluştuğunu göstermektedir.

- **Hizmet Türü:** Göndericinin ağdan beklediği güvenilirlik, hız ve gecikmenin düzeyini belirtmektedir. Ancak bu alanı mevcut yönlendiricilerin pek azı değerlendirilmektedir.

- **Toplam Uzunluk:** Başlık ve verinin uzunluğunun toplamını göstermektedir.

- **Tanım:** Alıcının parçaları (fragment) birleştirmek için kullandığı bir değerdir. Aynı IPpaketinin bütün parçalarının tanıtıcı değeri birbirinin aynıdır.

- **Parçalama Bilgisi:** IP paketinin DF (Don'tfragment) ve MF (More Fragment) bölümlerinden oluşan parçalama bilgilerini içermektedir. DFalıcının parçaları birleştiremediği durumlarda, paketinyönlendiricilerden geçerken parçalara bölmemesini gösteren 1 bitlik istek alanıdır.MF (More Fragment)kısmında bir veri bloğunun(datagram) son parçası dışındaki tüm parçalarında MF=1'dir.

- **Bayrak:** Üç adet bayrak bitinden oluşan kısımdır. İlk bit bilgisi içinde bulunduğu veri bloğununun kaçparçadan oluştuğunu belirtir. Eğer bu değer 1 ise gönderilen verinin tek datagramdanoluştuğu anlaşılmaktadır.Bu sayede alıcı veriyi aldıktan sonra başka mesajın olmadığını anlaşılmaktadır.İkinci bayrak bilgisi verinin parçalanıp birçok datagram haline dönüştürüldüğünü vegönderilen verinin en son veri bloğunun olduğunu belirtmektedir.Üçüncü bit alanı ise saklıtutulmaktadır.

- **Yaşam Süresi(Time to Live):** IP paketinin ağda sürekli dolaşmasını engelleyen bir alandır. Paketin alıcısına belirli bir süre içinde ulaşamaması durumunda yokedilmesini sağlamaktadır. Başlangıç için bu alana 255 veya daha küçük bir tam sayıyerleştirilmektedir.Her yönlendiricide bu alandaki değer bir eksiltilmektedir. Sayı 0(sıfır)'a ulaştığında paket çöpe atılmaktadır veyönlendirici kaynağa biruyarı paketi göndermektedir.

- **Protokol:** İletişim katmanında yürütülen protokol bilgisini göstermektedir.Bu protokollerTCP ve UDP protokolleridir.Bunlardan yalnız biri iletişim anında kullanılabilmektedir.

- **Başlık Kontrol Toplamı:** Başlıkta bir bozulma olup olmadığını belirlemeye yaramaktadır. Heryönlendiricide bu alandaki değer kontrol edilerekpaketin durumu incelenmektedir.Paket bozulmamışsa bir sonraki yönlendiriciye gönderilmekte ve

her pakette yeniden hesaplanmaktadır. Kullanılan yöntem yalnızca başlıktaki hataları ortaya çıkarmaktadır.

- **Kaynak ve Varış Adresleri:** Paketinin gönderildiği kaynak ve varacağıhedef IP adres bilgisinin bulunduğu kısımdır.

- **Seçenekler:** Bu alanda IP paketlerine eklenecek güvenlik, izlenecek yörünge, yönlendirici numaralarını ve gerçekzaman saatlerini gibi bazı ek bilgiler bulunmaktadır(Yıldırımoğlu, 2012).

3. SALDIRI VE SALDIRI TESPİT SİSTEMLERİ

3.1 Temel Kavramlar

- **Zayıflık (Vulnerability):** Bir bilişim sistemi ya da aktivitesinin barındırdığıgüvenlik yöntemlerindeki bozukluk ya da savunmasızlık, tasarım ya da uygulamadagerçekleşen aksaklıklar (yanlışlıkla ya da kasıtlı olarak) gibi etmenlerin sonucundaoluşan ve tehditler tarafından faydalanılabilecek özelliklere sahip güvenlik açıklarınaverilen isimdir[1].

- **Tehdit (Threat):** Sistem zayıflıklarını kullanarak sisteme zarar vericidavranışlarda bulunma olasılığı olan eylemlerdir. Tehdit, zayıflıkları istismar edebilecekpotansiyel tehlikedir.Bilgisayar güvenliğinde bilgi sistemini zorlayacak, bilgileri açığa çıkaracak, değiştirebilecek, hizmet vermeyi engelleyebilecek herhangi bir durum ya da olay[2] olarak da tanımlanabilir.Tehdit bilinçli (örn.güvenlik kırıcılar, suç organizasyonları) veyabilinçsiz (örn.bilgisayar arızaları veya doğal afetler) olabilmektedir.

    - "James P. Anderson a göre tehdit; bilişim sisteminde bilgiye erişmek bilgiyi kendi çıkarları için kullanmak ya da sistemi çalışmaz veya güvenilmez hale getirmek için var olanpotansiyel imkanlardır"(Canlı, 2009).

- **Saldırı (Attack):** Bir bilgisayar sistemine izinsizce girme, bir örün (web) sayfasını kirletme, bir Truva atı sokma, bir kodu kırma girişimi benzeri çabalardır[3].Saldırının başka bir tanımı olarak Canlı (2009) 'nın çalışmasındaki ifade gösterilebilir.

---

[1]*Vulnerability*, http://bilisimsozlugu. net/vulnerability, (10.04.2014)

[2]http://www. bilisimsozlugu. net/threat (10.04.2014)

[3]http://www. bilisimsozlugu. net/attack (10.04.2014)

- "Kurumve şahısların sahip oldukları bilgilere yetkisiz erişmek, zarar vermek, maddi/manevikazanç saglamak vb. için bilişim sistemleri kullanılarak yapılan her türlü hareket"(Canlı, 2009).

- **Saldırgan (Hacker):** Becerisini gizli bilgi kaynaklarına ulaşmak, bilgisayar ve ağlar üzerinde yasal olmayan işler yapmak üzere kullanan bilgisayar tutkunu, korsan[1]. Bilgisayar ve haberleşme teknolojileri konusunda bilgisahibi olan, bilgisayar programlama alanında standartın üzerinde beceriye sahip bulunanve böylece ileri düzeyde yazılım ve saldırı teknikleri geliştiren kişilerdir(Şeker,Uçar, 2008).

- **Nüfuz (Intrusion):** Bilgisayar sistemine, güvenlik önlemlerini aşarak yetkisi olmadan girme. Bilgi güvenliğinde, izinsiz giriş bir güvenlik gediği (security breach) örneğidir[2]. Intrusion kelime anlamı olarak zorla girme, izinsiz girme olarak tanımlanmaktadır.Bilişim terimi olarak güvenliği kırma, araya girme olarak da tanımlanmaktadır[3].

- **Sızma (Penetration):** Kelime anlamı olarak "girme ve giriş"[4] olarak geçmektedir. Bilişim sistemlerine karşı yapılmış yetkisiz erişim olarak da tanımlanabilir.

- **Güvenli (Secure):** Bilişim sistemlerinin kurulma aşamasında ve daha sonra öngörülen çeşitli tehditlere karşı önlemler alınarak kullanılabilir hale getirilmesidir. Güvenlik ve güvence altına almak olarak da tanımlanmaktadır[5].

- **Sömürü (Exploit):** Kelime anlamı olarak faydalanmak, istismar etmek, olarak tanımlanmaktadır.

---

[1]http://www. bilisimsozlugu. net/hacker (10.04.2014)

[2]http://www. bilisimsozlugu. net/intrusion (10.04.2014)

[3]http://tureng. com/search/intrusion (10.04.2014)

[4]http://tureng. com/search/penetration (10.04.2014)

[5]http://tureng. com/search/secure (10.04.2014)

3.2 Saldırı Türleri

İnternet kullanımının yaygınlaşmasıyla başka kullanıcıların bilgilerine erişmek isteyen kötü niyetli kişilerde artmaktadır.Bu kişiler değişik yöntemler kullanarak saldırı gerçekleştirirler. Saldırganların sürekli kendilerini yenilemelerive bilgisayar sistemlerinde var olan açıkları tespit etmeleri nedeniyle saldırı tiplerininçeşitliliğinin önemi giderek artmaktadır.Bu çalışmada STS'lerin gelişimi sırasında önemli bir yeriolan DARPA veri kümelerinin oluşturulması sırasında belirlenen ve akademik çalışmalarda hala geçerliliğinikoruyan saldırı tipleri esas alınmıştır.MIT Lincoln Laboratuarlarında yapılan buçalışmada, saldırılar bilgisayar sistemine yapılan atak türlerinin kullandıklarıyöntemlere göre dört gruba ayrılmış ve DoS(Denial of Service), U2R(User to Root), R2L(Remote to Local) ve Probing olarakadlandırılmıştır[1].

3.2.1 Hizmet aksattırma (DoS)

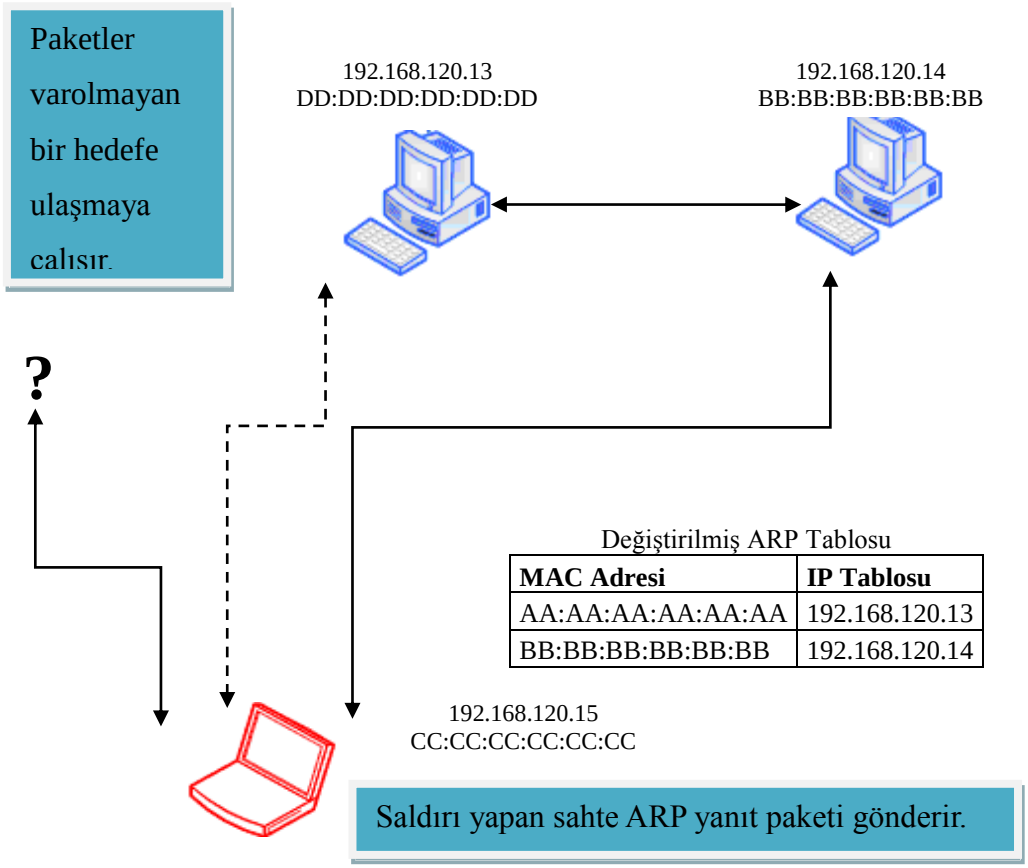
Genellikle hizmetin verilmesi engellenmek istenildiğinde sistemecevap verebileceğinden çok istek paketleri gönderilmesi ile hedefteki belleği şişirilerek gerçekleştirilen saldırı tipidir.Saldırı ARP tablolarındaki değerleri değiştirmeye yöneliktir.Saldırgan ARP tablolarında yer alan IP-MAC adres eşleştirmesinin gerçek olmayan değerlerle değiştirilerek iki sistem arasındaki iletişimi engellenemeye çalışmaktadır. Burada, hedef sistemin ARP tablosunda yer alan ve iletişimi engellenecek IP adresi için sahte ARP yanıt paketleri göndererek olmayan bir MAC adresinin hedef ARP tablosuna kaydedilmesini sağlamaktır (Canlı, 2009).

DARPAveritabanında isimlendirildiği haliyle, en çok bilinen DoS saldırı tipleri, SYN flood, Smurf, UDPstorm, Pingflood, Neptune, Mailbomb gibi saldırılardır (Mukkamala, 2002).

Şekil 3.1'de Hizmet Aksattırma(DoS) saldırısı örneği gösterilmektedir.

---

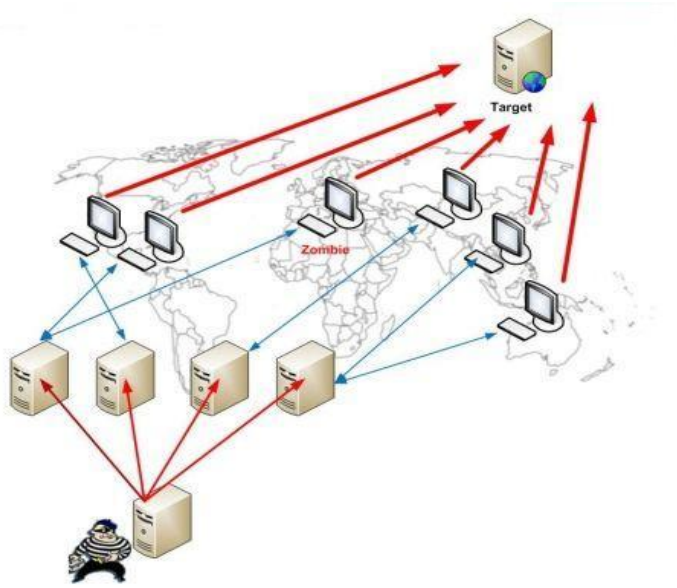
[1]http://www. ll. mit. edu/IST/ideval/ (10.04.2014)



Şekil 3.1. Hizmet aksattırma (DoS) saldırısı örneği.

DoS saldırısının dağıtık şekilde pek çok kullanıcı ya da zombi bilgisayarlar tarafında yapılmasıyla DDoS (Distributed Denial Of Service) saldırı tipi gerçekleştirilmektedir.

Şekil 3.2'de Dağıtık Hizmet Aksattırma(DDoS) saldırısı örneği gösterilmektedir.



Şekil 3.2. Dağıtık hizmet aksattırma (DDoS) saldırısı örneği[1].

3.2.2 Yetki yükseltme (U2R)

U2R saldırıları,kullanıcıların normal yetkilere sahip olan kendi hesaplarındanoturum açtıktan sonra yönetici yetkisine ulaşmaya çalışmasıdır.Bu şekilde yönetici yetkisiyle sistem üzerinde istedikleri bilgilere erişilebilinir.DARPA veritabanında isimlendirildiğihaliyle, en çok bilinen U2R saldırı tipleri Eject, Fbconfig, Fdformat, Loadmodule, Perl gibi saldırılardır (Mukkamala, 2002).

3.2.3 Uzaktan erişim (R2L)

Bu saldırı tipinde saldırgan saldırdığı makineye ağ üzerinden paketler yollayarakmakinenin açıklarından yararlanmaya çalışmaktadır.Bu konuda birçok araç olması ve bu araçlara erişimin kolay olması sebebiyle, sistemde var olan açıklar saldırgandan önce tespit edilipkapatılmamışsa, oldukça etkili ve kolay bir saldırı

[1]http://news.softpedia.com/newsImage/Master-Control-Server-for-Mydoom-DDoS-Botnet-Tracked-to-UK-3.jpg/ (15.04.2014)

yöntemidir.DARPA veritabanındaisimlendirildiği haliyle, en çok bilinen R2L saldırı tipleri Dictionary, Guest, Imap, Named, Sendmail gibi saldırılardır(Mukkamala, 2002).

3.2.4 Yoklama (Probe)

Probe ya da Probing saldırısı olarak da bilinen yoklama saldırısı, ağı veya bilgisayarı tarayarak zayıflıkları tespit etmek ve sistem yapısı ile ilgili genel bir bilgiye ulaşmak için yapılmaktadır. Sistem hakkında detaylıbilgi edinildikten sonra nasıl bir saldırı yapılması gerektiği belirlenmektedir.Yoklama saldırısı için kullanılan araçlar aynı zamanda güvenlik uzmanları tarafından sistemingüvenliğinin test edilmesi için de kullanılan araçlardandır. DARPA veritabanındaisimlendirildiği haliyle, en çok bilinen Probe saldırı tipleri, Ipsweep, Mscan, Nmap, Saint, Satan gibi saldırılardır (Mukkamala, 2002).

3.3 Tespit Edilecek Saldırılar

Bu çalışmamızda YSA'nın eğitiminde iki adet DoS saldırısının bulunduğu eğitim kümesi kullanılmıştır. Bu eğitim seti içerisinde Imap ve Pod saldırıları bulunmaktadır. Test kümelerinde ise bilinmeyen saldırı tespitinde yoklama (probe) saldırısı kapsamında NMAP saldırısı kullanılmıştır.

3.3.1 IMAP saldırısı

IMAP(İnternet Mesaj Erişim Protokolü), bir e-posta iletişim protokolüdür. 1986 yılında Stanford Üniversitesi'nde geliştirilmiştir. Eposta sunucularından mesaj çekmek konusunda yaygın protokollerden biridir. Genel kullanımda, bir kullanıcının e-posta istemcisini kullanarak yolladığı e-posta mesajları, önce kullanıcının oturum açtığı e-posta sunucusu tarafından kabul edilmekte ve genellikle SMTP kullanarak alıcının posta kutusunu içinde barındıran başka bir e-posta sunucusuna gönderilmektedir. Bu aşamada alıcının göndericinin mesajlarına ulaşabilmesi için bunu e-posta istemcisi ile çekmesi gereklidir. SMTP tek yönlü bir protokoldür. Kullanıcının isteği üzerine posta kutunuzda bulunan e-posta mesajının istemciye inmesini sağlayamaz. Bu aşamada yapılandırmaya bağlı olarak POP3 veya IMAP devreye girerek ilgili mesajın oturum açmış ve talep

etmiş istemciye çekilmesi sağlanmaktadır. IMAP ve POP3 kullanımı arasındaki temel fark IMAP ile E-Posta sunucusuna bağlantı kurulduğunda, kutuda birikmiş e-postaların sadece başlık bilgilerini istemciye getirirken POP3 ise bütün mesajları istemciye çekmektedir[1].

SMTP'nin tek yönlü çalışan bir protokol olduğundan, bu özelliği kullanarak saldırı amaçlı kullanılmaktadır.

3.3.2 Pod saldırısı

Pod saldırısı, bilgisayara bozuk veya kötü amaçlı ping göndermeyi içeren saldırı türüdür.Anlam olarak "ölümüne ping" ya da "ölümcül ping" olarak da tarif edilebilmektedir.Bir ping paket boyutu normal olarak 64 bayttır (veya IP başlığı dikkate alındığında 84 bayt); birçok bilgisayar sistemi maksimum IP paket büyüklüğü olan 65.535 bayttan daha büyük bir pingi işleyemez[2]. Bu boyutta bir ping göndermek ise hedef bilgisayar beklemediği bir durum olduğundan cevap veremez ve hizmet durdurulur.

Günümüzde bilgisayar sistemleri, Pod saldırılarını kolaylıkla engellemektedir (Tanrıkulu, 2009).

3.3.3 NMAP saldırısı

NMAP, bilgisayar ağları uzmanı Gordon Lyon (Fyodor) tarafından C/C++ ve Python programlama dilleri kullanılarak geliştirilmiş bir güvenlik tarayıcısıdır. Taranan ağın haritasını çıkarılabilmekte ve ağ makinalarında çalışan servislerin durumlarını, işletim sistemlerini, portların durumları gözlemlenebilmektedir.

---

[1]http://tr. wikipedia. org/wiki/IMAP (15.04.2014)

[2]http://docs.trendmicro.com/all/smb/wfbs-services/Server/Dell/v3.7/tr/docs/WebHelp/_WFBS-SVC/C07-ConfiguringGroupSecuritySettings/IntrusionDetectionSystem.htm (15.04.2014)

NMAP, özgür GPL(General Public License) lisanslı yazılımdır ve istendiği takdirde ilgili sitelerden indirilebilmektedir[1]. NMAP'ın kullanılabildiği işletim sistemleri Linux, Windows, MacOS, Solaris, *BSD olarak sayılabilir; fakat popülerliği öncelikle Linux daha sonrasında Windows işletim sistemindedir[2].

NMAP kullanılarak ağa bağlı herhangi bir bilgisayarın işletim sistemi, çalışan fiziksel aygıt tipleri, çalışma süresi, yazılımların hangi servisleri kullandığı, yazılımların sürüm numaraları, bilgisayarın güvenlik duvarına sahip olup olmadığı, ağ kartının üreticisinin adı gibi bilgiler öğrenilebilmektedir.

Ağdaki uzmanları tarafından mevcut sistemleri korumak ya da zafiyettaramak ve açıkları kapatmak amaçlı kullanılırken kötü amaçlı kişiler tarafından saldırı amaçlı kullanılabilmektedir.

3.4 Saldırı Tespit Sistemleri

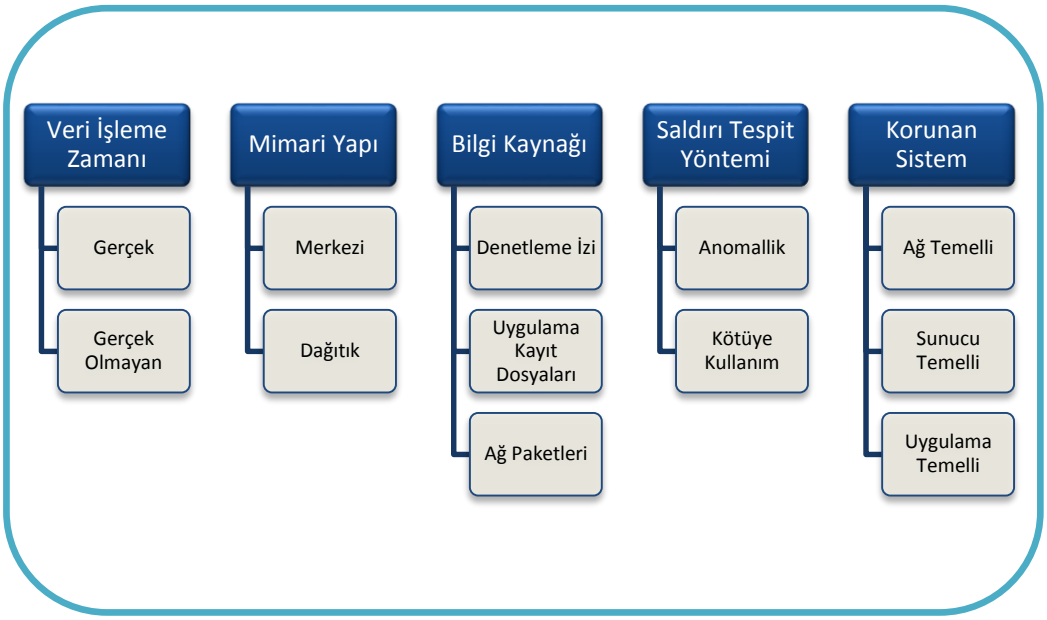
3.4.1 STS sınıflandırılması

STS'ler günümüze kadar farklı birçok kritere göre sınıflandırılmıştır. Bunlardan en çok bilinen sınıflandırma türü saldırı tespit yöntemine göre olup, anormallik tespiti ve kötüye kullanım tespiti olarak ikiye ayrılmaktadır(Anderson, 1995). Ancak STS'lerin mimari yapısı, korunan sistemin türü, verinin işlenme zamanı gibi farklı sınıflandırmalar da yapılabilir (Axelsson, 2000).

STS'lerin en çok kullanılan kriterlere göre sınıflandırılması Şekil 3.3'de gösterilmektedir.

[1]http://nmap.org/download.html sitesi gibi pek çok siteden indirilip kurulabilir. (15.04.2014)

[2]http://tr. wikipedia. org/wiki/Nmap (15.04.2014)



Şekil 3.3. Saldırı tespit sistemlerinin sınıflandırılması(Güven, 2007).

Bir STS, Şekil 3.3'de gösterilen sınıflandırma kriterlerinden her biri ile farklısınıflarda yer almaktadır(Axelsson, 2000). Bu, sınıflandırmanın hangi kritere göre yapıldığına bağlı olarak değişebilmektedir.

Örneğin; 1988'de geliştirilen IDES;

- Veri işleme zamanına göre gerçek zamanlı,
- Mimari yapısı bakımından merkezi sistemli,
- Kullanılan bilgi kaynağı olarak denetleme izi kullanan,
- Saldırı tespit yöntemi olarak anormallik tespiti yaklaşımı,
- Koruduğu sistemegöre sunucu temelli sınıfına dahildir (Jonas, 2000).

Belirlenen sınıflandırma kriteri sadecesaldırı tespit yaklaşımı ise IDES anormallik tespiti yaklaşımını kullanan sınıfının birüyesidir.Bu durumda diğer kriterlerden bahsedilmez. Tüm bu kriterler, aynızamanda bir STS'nin karakteristiğini ortaya koyması açısından önemlidir(Güven, 2007).

3.4.2 Veri işleme zamanı

STS'ler için veri işleme zamanı, izlenen olaylar ve olayların analizi arasında geçenzamanı ifade etmektedir.STS'ler "gerçek zamanlı" ve "gerçek zamanlı olmayan" şeklindeikiye ayrılmaktadır(Axelsson, 2000).

- **Gerçek zamanlı sistemler:** Veri, iletişim anında analiz edilir ve eğer bir saldırıtespit edilirse saldırıya cevap olarak gereken çevre değişimleri gerçekleştirilmektedir.Genellikle ticari uygulamalar gerçek zamanlı sistemlerdir.Bu tipSTS'ler, yoğun bilgi akışı olan ağlarda uygulanması zor ve maliyetli bir yöntemolmakla birlikte aktif olarak cevap üretilmesi gereken durumlarda da tek çözüm olansistemlerdir.

- **Gerçek zamanlı olmayan sistemler:** Saldırının profilinin çıkarılması ve olası saldırılara maruz kalınmaması için, geçmişe yönelik depolanan verilerüzerinde yapılan analizlerle sonuçlar veya saldırılar elde edilmektedir. STS'nin gerçek zamanlı olması şartı aranmadığı durumlarda kullanılan bu yöntem, sisteminbelirlenen güvenlik açıklarını kapatmak için kullanılmaktadır.

3.4.3 Mimari yapı

STS'lerin mimari yapısı, fonksiyonel bileşenlerin birbirlerine göre nasılyerleştirildiklerini anlatmaktadır(Murali, 2005). Temel fonksiyonel bileşenleri olarak; analizinyapıldığı sunucu ile çevresi, izlenen sistem ve problemler için izlenen hedef olarak sıralanabilir.STS'lerin mimari yapısı, merkezi sistem ve dağıtık sistem olmak üzere ikiye ayrılmaktadır(Jonas, 2000).

- **Merkezi sistem:** Tüm izleme, tespit ve raporlama işlemlerimerkezde kontrol edildiği sistemlerdir. Merkezi kontrol için fiziksel yakınlık önemli bir etken olmadığından çoğu STS'lerin bu kategoriye girmektedir.

- **Dağıtık sistem:** İzleme, tespit vecevap işlemleri, analiz işlemlerini merkezi olmayıp kurulan ajan (agent) tabanlı programlar ile yapılmaktadır. Genellikle çok geniş olan ağlarda kullanılan bir yapıdır.Örneğin, bir kurumun farklışehirlerde yer alan bilgisayar sistemlerinin tümünü kapsamına alan STS'ler buşekildedir(Güven, 2007).

3.4.4 Bilgi kaynağı

STS'lerde bilgikaynakları, bilgisayar veya ağ paketlerinin dinlenmesinden elde edilebildiği gibi, kullanıcı profillerinin davranış modellerinden de elde edilebilmektedir. Bilginin nasıl venereden toplanacağı, kullanılacak olan STS'nin amaçlarına göre değişmektedir.Bunlar;denetim izi, ağ paketleri, uygulama kayıt dosyalarıdır.

- **Denetim (hesap, günlük) izi**: Denetleme izleri, sistemde gözlenen olaylarınsıralaması bozulduğunda veya olaylarda değişiklikler meydana geldiğinde, yenidenyapılandırma ve test etmeyi mümkün kılmak için kullanılmaktadır. Kullanıcı tanımlamasistemleri ve veritabanı yönetim sistemlerinin çoğu denetleme izi bileşeni içermektedir.STS'ler bilgi kaynağı olarak denetleme izini, daha çok sistemde tanımlanmış olankullanıcıların veya grupların hareket profillerini çıkarmak için kullanılmaktadır.Kullanıcı profilleri, kullanıcıların günlük yaptıkları işler ve bu işlere yönelik sistemdeki yetkileri gözönünde bulundurularak oluşturmaktadır.Kullanıcı profilin doğruluğunun kanıtlanması için uzun süren bir gözlem ve inceleme gerekmektedir.

  Kullanıcının profili belirlenmişse, STS'ler belirlenenprofil dışına çıkılan hareketleri saldırı olarak algılamaktadır. Saldırı tespitindeyanlış alarmoranlarını azaltmak için kullanıcı profillerinin güncellenebilir olması gerekmektedir.

- **Ağ paketleri:** Koklayıcı(sniffer) programlar tarafından trafiğin dinlenmesiyle elde edilmektedir. Ağ paketlerinden elde edilen bilgiler sayesinde, sunucu tabanlı STS'lerdenfarklı olarak, ağ katmanında gerçekleşen saldırı olaylarını tespit etmek de mümkündür. Bunun için saldırı tespit sisteminin ağ yapılandırması sırasında en uygun yerekonumlandırılmış olması gerekmektedir.

- **Uygulama kayıt dosyaları**: Uygulama katmanında gerçekleşen saldırıları tespit etmektekullanılmaktadırlar.Bu veri kaynağı, diğer iki kaynaktan daha kolay elde edilmesiylebirlikte sağladığı saldırı tespit oranı sınırlıdır.

3.4.5 Saldırı tespit yöntemi

STS'lerde, saldırı tespit yöntemi olarak anormallik tespiti ve kötüye kullanım tespitiolmak üzere 2 farklı yaklaşım kullanılmaktadır(Anderson, 1995). Anormallik tespitine dayananyaklaşım, sistemdeki kullanıcı davranışlarını modellerken, kötüye kullanım (imza)tespitine dayanan yaklaşım, saldırganların davranışlarını modellemektedir.

- **Anormallik tespiti (anomaly detection):** Sistemde meydanagelen anormal olayları, normal olaylardan ayırt etme yaklaşımıdır. STS için anormallik, normal davranıştan sapma anlamına gelmektedir. Normal davranış, sistemin uzun bir süre analiz edilmesi ile elde edilmektedir.Sistemdeki kullanıcı profillerinin belirlenmesi anormallik tespiti için temel iştir.Normal davranış profili belirlendikten sonra, farklılık gösteren davranışlar saldırı olarak tespit edilmektedir.Anormallik tespitinde saldırıların doğru tespit edilmesi, normaldavranış profilinin ne kadar doğru belirlendiğiyle ilişkilidir.

  Anormallik tespiti yaklaşımının kötüye kullanım tespiti yaklaşımına göre avantajı, önceden bilinmeyen saldırıların da tespit edilebilmesidir. Dezavantajı ise yanlışalarm (false alarm) yani gerçekte saldırı olmayan davranışların da saldırı olarakbelirlenmesi oranının yüksek olmasıdır. Anormallik tespitinde istatistiksel yöntemler, yapay zeka teknikleri, yapay sinirağları, veri madenciliğigibi birçok farklı teknik kullanılabilmektedir.

- **Kötüye kullanım tespiti (misuse detection):** Saldırının imzası olarak da nitelendirilen bumodeller, daha önce karşılaşılmış saldırıların analiz edilerek kendine özgü karakteristik özelliklerinin çıkarılması ile elde edilmektedir.

Saldırılara yönelik imzaveritabanları oluşturulduktan sonra, bu imzalarla eşleşen hareketler saldırı olaraktespit edilmektedir.

Kötüye kullanım tespiti yaklaşımının anormallik tespitine göre avantajı, imzasıbilinen her saldırının tespit edilebilmesi ve yanlış alarm üretmemesidir.Dezavantajıise, imzası bilinmeyen saldırıların tespit edilememesi ve bu nedenle yanlış (saldırı olan bir davranışın saldırı olarak tespit edilmemesi) oranının yüksekolabilmesidir. Yanlış alarm oranının azaltılması için imza veritabanının yeni saldırıimzaları ile güncellenebiliyor olması gerekmektedir (Güven, 2007).

3.4.6 Korunan sistem

STS'ler korudukları sisteme göre üç gruba ayrılmaktadır. Korumak istenen sisteme göre; ağ, sunucu ya da uygulama temelli STS'ler olarak adlandırılmaktadır(Jonas, 2000).

- **Ağ temelli:** Ağdaki trafiği dinleyerek ağ sistemine yapılan saldırıları tespit etmeye yönelik çalışmaktadır. Ağ paketlerini yakalayıp bunları analiz ederek saldırı tespitiyapmaktadır.

- **Sunucu temelli:** Bilgisayar sistemi işletim sistemi hesap izlerinive sistem kayıtlarını depolarlar ve toplanan veriler üzerinde çalışmaktadır. İşletim sistemine yönelen saldırılar için hangi sistem çağrılarının ve hangi kullanıcıların sorumlu olduğu tespit edilebilmektedir.Sunucu temelli STS'ler, denetim izi ve sistem günlük dosyaları olmak üzere iki tür bilgi kaynağı kullanmaktadır.

  Sunucu temelli sistemlerin ağtemelli sistemlere göre bir avantajı, olayların olduğu yerel sunucuyu izlemeyetenekleri sayesinde ağ temelli STS'lerin yakalayamayacağı saldırıları tespitedebilir olmalarıdır. Bu saldırılar çoğu zaman fazla trafik yaratmayan R2L veya U2Rsaldırılarını içermektedir. Bazı sunucu temelli STS'ler merkezi STS'lere destek vermek için tasarlanmıştır.

- **Uygulama temelli:** Uygulama temelli STS'lerde bilgi kaynağıolarak genellikle uygulamaya ait günlük dosyalar kullanılmaktadır. Analiz motoruna, belirliuygulamalar için o uygulamalara has özellikler bildirildiği takdirde, yetkisini aşankullanıcıların gerçekleştirdiği saldırılar uygulama temelli STS'lerce tespit edilebilmektedir(Axelsson, 2000).

3.5 STS'lerde Kullanılan Teknikler

STS'lerde, anormallik ve kötüye kullanım (imza) tabanlı yaklaşımları modellemekiçin günümüze kadar birçok teknik kullanılmıştır (Patcha, Park, 2007). Elde edilen tekniklerverilerin modellenmesi, sınıflandırılması veya kural tablolarının oluşturulması içingeliştirilmiştir. Kullanılan tekniklerden elde edilen veriler sayesinde, saldırı tespityaklaşımlarının uygulanması için gerekli olan platform oluşturulmuştur(Güven, 2007). Bu teknikler sırasıyla verilmektedir.

- **Veri madenciliği:** Veritabanındaki saklı olayları ortaya çıkarmak için yapılan bilgiaçılımıdır.Paternleri ve veriler arasındaki ilişkileri bularak kural çıkarmak içinkullanılmaktadır.Hesap izleri kullanılarak normal kullanıcı aktiviteleritanımlanmaktadır(Lee vd. , 2001).

- **Kural tabanlı (Rule Based) sistemler:** Sistem trafiğini inceleyip kurallar oluşturulmaktavesaldırı tespiti sırasında belirlenen kurallara göre davranışlar sınıflandırılmaktadır(Ilgun vd. , 1995).

- **Açıklayıcı istatistikler (Descriptive Statistics):** Sistem ya da kullanıcı davranışları farklı değişkenlere (örn:kullanıcı oturum girişi, oturum kapatma, belli bir zaman periyodunda erişilen dosya sayısı, kullanılan disk alanı) göre ölçülerek istatistiksel bir model oluşturulmaktadır. Kullanıcı profilleri ve hesap izleri kullanılarak normal davranışların modeli oluşturulmakta ve anormallik tespit edilmektedir. Kullanıcı profilinin basit istatistiklerle oluşturulup, buradan uzaklık vektörlerini (distance vector) kullanarak karar alan sistemlerdir. Davranış profili oluşturulurken, kullanılan işlemci zamanı, bir zaman periyodundaki ağ bağlantı sayısı gibi farklı ölçütler de

kullanılabilir. İstatistiksel yaklaşımların dezavantajlarından biri, saldırganın bu istatistikleri öğrenerek ona göre davranış sergileyebilmesidir(Sundaram, 1996).

- **Eşik Değeri Tespiti:** Eşik değer tespiti oluşturulurken özel olayların tekrarlama sayısı ve zaman periyodu dikkate alınmaktadır. Sorun olarak eşik değerinin belirlenmesi ve özel olaylar için çerçevenin belirlenmesi gösterilebilmektedir. Örnek olarak; yanlış girişler, giriş/çıkış hata sayısı veya silme sayıları verilebilmektedir. Büyük STS'lerde alt bileşen olarak kullanılmaktadır.

- **Durum Geçiş Analizi:** Bir işin yapılması için birbirini takip eden durum sırası olduğu varsayılmakta ve buna göre seri oluşturulmaktadır. Saldırının tamamlanması için imza hareketleri oluşturulması gerekmektedir. Sızmaların senaryosu çıkarıldıktan sonra, anahtar hareketler, imzahareketler olarak tanımlanmaktadır. Durumlar, geçişler ve imzalar, durum geçiş diyagramıolarak grafiksel biçimde sunulmaktadır(Porras, 1992). Tüm davranışlar durumlara karşılık gelmektedir. Bir davranış daha önceden tanımlı durumlara benzer şekildehareketler yapıyorsa saldırı olarak tanınmaktadır.

- **Uzman Sistemler:** Belirli bir alanda bilgilerle donatılmış veproblem çözümünde uzman bir kişinin getirdiği şekilde çözümler getirebilen bilgisayar programlarıdır. (Axelsson, 2000).

- **Örüntü Eşleme (Pattern Matching):** Sistemde daha önceden tanımlanmış vekarşılaşılması istenmeyen bazı sözcüklerin tanınması için kullanılmaktadır. Örneğin "parola dosyasını kopyala" komutu görüldüğünde bunun bir saldırı olduğunu bu yöntem tespit etmektedir(Axelsson, 2000).

3.6 STS'lerin Başarı Kriterleri

Saldırı tespit sistemleri, hızlı gelişimi ve sağladığı güvenlik desteği nedeniyle büyükağlarda standart bir araç haline gelmiştir.STS'lerin günden güne yenilerinin eklenmesine karşın, ne oranda başarılı oldukları veya gereksinimlerin ne kadarını

karşıladıkları hakkında net bir cevap bulmak zordur.STS'lerin test edilmesi içingünümüze kadar birçok çalışma yapılmıştır, fakat bu çalışmalar bazıSTS'lerin karşılaştırılmasından öteye geçememiştir. Bu sebeple saldırı tespit sistemlerininbaşarılarını ölçmek için genel bazı kurallar ve hesaplanabilir değişkenlerortaya çıkarılmıştır. Mell ve arkadaşlarının yaptığı bir çalışmada STS'lerin başarı kriterleri belirlenmiştir(Mell vd., 2003). Bu kriterler, ilgili kaynak temel alınarak sırasıyla aşağıda altbaşlıklarda açıklanmıştır.

- **Kapsam:** Bir STS'nin ideal koşullarda hangi saldırıyı tespit edebildiği olarak tanımlanmaktadır. Örneğin; kötüye kullanım tespitine dayalı STS'lerde kapsam, imza sayısı ve buimzaların standart isim düzeni ile haritalanmasını içermektedir.Saldırılarda değişiklikler yapılaraktüretilen saldırılar bu ölçütü zorlaştırmaktadır.

- **Yanlış alarm olasılığı (probability of false alarms):** Bu ölçüt, verilen çevrede belli bir zaman aralığında STS tarafından üretilen yanlışpozitif alarm oranını tanımlamaktadır. Yanlış değerlendirmeler iki çeşit olabilmektedir.Bunlar, var olan bir değeri kaçırmak ya da var olmayan bir değeri varmış gibi tespitetmektir.Yanlış alarm tanımı ikincisi için geçerlidir. STS sistemlerinin başarımını ölçmekteönemli bir parametredir çünkü bir sistemde izin verilen yanlış alarm sayısı ve doğrutespit miktarı birbiriyle ilintilidir.Sistem parametreleri ikisinin de optimum olduğu noktaya ayarlanmalıdır(Güven, 2007).

- **Tespit etme olasılığı (probability of detection):** Bu ölçüt, bir kapsam dahilinde belli bir zaman aralığında STS tarafından doğru tespitedilebilen saldırı oranını tanımlamaktadır. Bir STS'nin var olan saldırıların ne kadarını yakaladığıönemli bir parametredir.STS'ler tespit etme oranını artırırken, yanlış alarmoranını da çok yükseltmemesi gerekmektedir.

- **STS'ye yönelik saldırılara karşı direnç:** Bu ölçüt, bir STS'nin doğru çalışmasını bozmak için yapılan girişimlere karşı nekadar dayanıklı olduğunu göstermektedir.

- **Yüksek bant genişlikli trafiği yönetebilmek:** Bu ölçüt, ağ tabanlı STS'lerde geçerli olmakta ve bir STS'nin geniş hacimli trafikle karşılaştığında ne kadar iyi olduğunugöstermektedir. Ağ tabanlı STS'lerin birçoğu trafik hacmi arttığında trafiği yönetmek için gelen paketleri düşürmeyebaşladığından, saldırıların olduğu paketleri de kaybedebilmektedir. STS'lerin bazıları, trafik yoğunluğu çok arttığı zaman saldırıtespitinde tamamen başarısız olabilmektedir. Örneğin, DoS ve DDoS saldırılarında sistemin güvenli olması önemli bir ölçüttür.

- **Olayları ilişkilendirebilme:** Bu ölçüt, bir STS'nin farklı kaynaklardan gelen saldırı olaylarını birbiriyle ilişkilendirebilme konusundane kadar iyi olduğunu göstermektedir.

- **Daha önce görülmemiş atakları tespit edebilme:** Bu ölçüt, bir STS'nin daha önce karşılaşmadığı bir saldırıyı tespit etmede ne kadariyi olduğunu göstermektedir.İmza temelli sistemler yeni gelen hiçbir saldırıyı tanıyamadıkları için sadece anormallik tespitine dayalı sistemler için geçerli olan bir ölçüttür.

- **Bir saldırıyı tanımlayabilme:**STS'lerin ilk önce saldırıların varlığını tespit etmektedir.Saldırı olduktan sonra iseağ yöneticisi tarafından çeşitli kayıtlarincelenerek ortaya çıkartılması da saldırı tipinin tespitidir.

- **Saldırı başarısını belirleyebilme:** Bilgisayar sistemlerine karşı gerçekleştirilen saldırıların hepsi başarı ile sonuçlanmaz yani sisteme zarar vermez.Gerekli önlemleri alınması için gerekli olan bir ölçüt olduğu söylenebilmektedir.

- **Diğer kriterler:** Kullanım vebakım kolaylığı, düşük kaynak tüketimi, gerçekleştirilebilirlik ve kalitegibi hemen hemen tüm yazılımlarda istenen genel kriterler, diğer ölçütlerdir.

3.7 STS Yazılımları

Saldırı tespit sistemlerinin, bilgi ve bilgisayar güvenliğindeki yerini almaya başlaması ile birlikte, bu konuda yapılan çalışmalarda artmıştır.Araştırmacılar, farklı ortamlarda, farklı sistemler üzerinde ve farklı tekniklerle birçok STS yazılımı geliştirmişlerdir (Axelsson, 2000). Bu çalışmalar, Çizelge 3.1'de listelenerek, karakteristik özelliklerine göre karşılaştırılmıştır(Axelsson, 2000 ve Jakson, 1999). Haystack, MIDAS, IDES, W&S, Computer Watch, NSM, NADIR, Hyperview, DIDS, USTAT, IDIOT, Janus, EMERALD, Bro, Ripper, Snort ve Snortsam bunlardan önemli olanlarıdır.

Çizelge 3.1. STS karşılaştırması(Güven, 2007).

Yıl	STS	STS Yöntemi	Veri İşleme Zamanı	KorunanSistem (Veri Kaynağı)	Mimari Yapı
1988	Haystack	Hibrid	Gerçek Olmayan	Sunucu	Merkezi
1988	MIDAS	Hibrid	Gerçek	Sunucu	Merkezi
1988	IDES	Anormallik	Gerçek	Sunucu	Merkezi
1989	W&S	Anormallik	Gerçek	Sunucu	Merkezi
1990	Comp-Watch	Anormallik	Gerçek Olmayan	Ağ	Merkezi
1990	NSM	Hibrid	Gerçek	Sunucu	Merkezi
1991	NADIR	İmza	Gerçek Olmayan	Sunucu	Merkezi
1991	Computer misuse detection system	Hibrid	Gerçek	Sunucu	Dağıtık
1992	DIDS	Hibrid	Gerçek	Hibrit	Dağıtık
1992	ASAX	İmza	Gerçek	Sunucu	Merkezi
1992	Intruder alert	İmza	Gerçek	Hibrit	Dağıtık
1993	USTAT	İmza	Gerçek	Sunucu	Merkezi
1994	DPEM	İmza	Gerçek	Sunucu	Merkezi
1994	IDIOT	İmza	Gerçek	Sunucu	Merkezi
1995	NIDES	Hibrid	Gerçek	Sunucu	Merkezi
1996	GrIDS	Hibrid	Gerçek Olmayan	Hibrit	Dağıtık
1996	CSM	İmza	Gerçek	Sunucu	Dağıtık
1996	JANUS	İmza	Gerçek	Sunucu	Merkezi
1996	Kane security monitor	İmza	Gerçek	Sunucu	Dağıtık
1996	Netranger	İmza	Gerçek	Ağ	Dağıtık
1996	Realsecure	İmza	Gerçek	Ağ	Dağıtık
1997	JiNao	Hibrid	Gerçek	Sunucu	Dağıtık
1997	EMERALD	Hibrid	Gerçek	Hibrit	Dağıtık

Çizelge 3.1. STS Karşılaştırması (Güven, 2007) (Devam Ediyor).

1997	Anzen fligh jacket	Hibrid	Gerçek	Ağ	Hibrit
1997	Netprowler	İmza	Gerçek	Sunucu	Dağıtık
1997	Securenet pro	İmza	Gerçek	Ağ	Dağıtık
1997	Sessionwall-3	İmza	Gerçek	Ağ	Dağıtık
1998	Bro	İmza	Gerçek	Ağ	Merkezi
1998	Centrax security suite	Hibrid	Gerçek	Hibrit	Dağıtık
1998	Cross-site for security	İmza	Gerçek	Ağ	Dağıtık
1998	Smartwatch	İmza	Gerçek	Sunucu	Dağıtık
1998	Stake out	İmza	Gerçek	Ağ	Hibrit
1998	Tripware	İmza	Gerçek Olmayan	Sunucu	Merkezi
1998	Snort	Hibrid	Gerçek	Sunucu	Merkezi
1999	Cybercop monitor	İmza	Gerçek	Sunucu	Dağıtık
2000	FIRE	Anormallik	Gerçek Olmayan	Ağ	Merkezi

- **Haystack:** Bu STS, 1988 senesinde Amerikan Hava Kuvvetlerinde kullanılan, OS/1100 işletimsistemini çalıştıran çok kullanıcılı Unisys 1100/60 ana bilgisayarları için tasarlanmışbir saldırı tespit sistemidir(Smaha, 1988).

  Haystack aslında altı farklı tür saldırının tespiti için tasarlanmıştır(Güven, 2007).

  - Kırma girişimleri (attempted break-ins)
  - Kılık değiştirilen saldırılar (disguised intrusion)
  - Güvenlik kontrol sistemine sızma (penetration of the security control sistem)
  - Sızma (leakage)
  - Hizmet aksattırma (denial of service)
  - Kötü niyetli kullanım (malicious use)

- **IDES:** IDES, Denning ve Neuman'ın 1985'li yıllarda yaptıkları çalışmalar ile gündemegelmiştir[1]. IDES için gereksinimleri ve tasarım modelini ortaya koyan buçalışmanın ardından, IDES üzerinde yapılan çalışmalar birçok araştırmacının

[1]http://www. csl. sri. com/programs/intrusion/history. html#IDES (10.02.2014)

dakatkıları ile 1987-1992 yılları arasında yoğun olarak devam etmiştir (Lunt, 1988, Dennis, 1987, Peddabachigari, 2007).IDES prototipi, denetleme verilerinde raporlanan kullanıcı davranışını, kullanıcınıngeçmiş hareket profiline göre normal olarak belirlemektedir(Peddabachigari, 2007). Zamanla değişebilenkullanıcı davranışlarına karşılık hareket profilleri her gün yenilenmektedir.IDES, bir uzman sistem bileşeni içerdiğinden, önceki saldırılardan edinilen bilgileredayanan özel davranışları, bilinen sistem açıklarını veya kuruluma özel güvenlikpolitikalarını tanımlayan kurallar içermektedir(Peddabachigari, 2007). IDES'in ilk sürümütek bir sunucu için tasarlanmış sonra yapılançalışmalarda tekrar dizayn edilerek, farklı birçok hedef sistem için kullanılabilir halegetirilmiştir.Pek çok sürümü olan yazılım, son olarak NIDES (Next-GenerationIntrusion Detection Expert Sistem) adını almıştır(Anderson, 1994).

- **MIDAS:** MIDAS, NCSC (Ulusal Bilgisayar Güvenlik Merkezi) tarafından, BilgisayarBilimleri Laboratuarı (Computer Science Laboratory) ve SRI (Stanford AraştırmaEnstitüsü) işbirliği ile, NCSC'nin ağa bağlanmış ana bilgisayarı Honeywell DPS-8/70 için saldırı tespiti sağlamak üzere geliştirilmiştir(Axelsson, 2000). MIDAS, sezgisel saldırı tespiti konsepti çevresinde geliştirilmiş uzman sistem tabanlıbir STS'dir(Güven, 2007).

- **W&S:** W&S (Wisdom and Sense – Bilge ve His), Los Alamos Ulusal Laboratuarlarındageliştirilen anormallik tabanlı bir çalışmadır.İlk çalışma 1989'da sunulmuştur.Bilge kısmı(W), geçmiştekidenetleme verilerini inceleyerek normal davranışı oluşturması anlamına gelmektedir. Hiskısmı(S) ise bunların kural haline getirilip uzman sisteme verildikten sonra anormaldavranışların yakalanması anlamına gelmektedir(Vaccarro, 1989).

- **NSM:** NSM'de IDES gibi revizyondan geçmiştir ve geliştirilmesi 1990-1994 yıllarına rastlamaktadır. NSM, ağı dinleyerek, ağın kullanımıyla ilgili bir profil geliştirmekte ve geçerlikullanımı onunla karşılaştırmaktadır.Elde edilen veriyle beklenen bağlantı verisikarşılaştırılmakta ve beklenen aralıkta çıkmayan her veri anormal olarak işaretlenmektedir(Axelsson, 2000).

- **NADIR:** NADIR (The Network Anomaly Detection and Intrusion Reporter) 1991-1993 yıllarıarasında Los Alamos Ulusal Laboratuarlarında (LANL-Los Alamos NationalLaboratory) bilgisayar ağları mühendisliği grubu tarafından geliştirilmiştir (McAuliffe, 1990).NADIR kullanıcılar hakkında haftalık istatistikler tutmakta, her hafta profil analizininsonucunda elde edilen detaylı raporlar oluşturulmaktadır.Bu istatistikleri sonra uzmansistem kurallarıyla karşılaştırmaktadır(Christoph, 1995).

- **Hyperview:** 1992 yılında geliştirilmiş olan Hyperview, diğer sistemlerden oldukça farklı iki ayrı bölümden oluşan birsistemdir. İlk bölüm davranışları izleyen ve sınıflayan bir uzmansistem, ikinci bölüm ise öğrendikleriyle eğitilen yapay sinir ağlarını içeren bölümdür(Axelsson, 2000).

- **USTAT:** 1993-1995 yılları arasında geliştirilmiş vedurum geçiş analizi yöntemi kullanılmıştır(Ilgun, 1993). STS'de, eğer bir davranış saldırı için tanımlı durum geçişlerini sağlıyorsa saldırı olarak sınıflandırılmaktadır.

- **DIDS:**1992 yıllarında geliştirilmiş dağıtık mimarili sistemleri kapsamaktadır(Snapp, vd. 1991). STS de ağın değişik noktalarından veriler toplanmakta ve veriler bir merkezde incelenmektedir.

- **IDIOT:** 1994-1996 yılları arasında CERIAS (Center for Education and Research inInformation Assurance and Security)'da Kumar tarafından geliştirilmiştir(Axelsson, 2000, Kumar, 1995). Kumar'ın tasarımı olan IDIOT, saldırı yöntemlerinin eşleştirme ve geçici karakteristiklerin karmaşıklığına dayalı olan bir sınıflandırma yöntemidir.

- **Janus:** 1994-1996'da Berkeley'de Wagner tarafından tez çalışması esnasında geliştirilmiş birsistemdir(Axelsson, 2000).

- **EMERALD:** EMERALD (Event Monitoring Enabling Responses to Anomalous LiveDisturbances) 1997-1998 yıllarında geliştirilmiş ölçeklenebilir, dağıtık bir STS'dir(Porras, Neumann, 1997).

- **Bro:** 1998'de Vern Paxson tarafından geliştirilmiş bir STS'dir. Gerçek zamanda ağtrafiğini pasif olarak gözlemleyerek saldırı tespiti yapmaktadır(Paxson, 1998). Birçok değişiközelliği bünyesinde barındırması açısından sıkça kaynak verilen bir STS'dir(Güven, 2007).
- **Ripper:** 1999 yılında geliştirilen bu sistem veri madenciliği yaklaşımını kullanmaktadır. RipperDARPA değerlendirmesine katılmış olan sistemlerden biridir (Axelsson, 2000).

- **Snort:** 1998 yılında Martin Roesch tarafından geliştirilen Snort, gerçek zamanlı trafikanalizi yapabilen ve paket kaydedebilen açık kaynak kodlu kural tabanlı bir saldırı engelleme ve tespit sistemidir. İmza ve anormallik tespiti yaklaşımlarının avantajlarını üzerinde barındırır[1]. Protokol ve içerik analizleri yaparak birçok saldırı ve sızma girişimini tespit etmekte ve çeşitli alarm mekanizmaları ile kullanıcıyı uyarmaktadır.

  Saldırı imzalarının düzenli olarak güncellenmesi sayesinde oldukça farklı sayıda ve türde saldırıyı tespit edebilmek mümkündür[2].Açık kaynak kodlu ve sürekli geliştirilmeye açık olması popülerliğini artırmış bununla birlikte ticari alternatifleri ile kıyaslanabilir duruma gelmesini sağlamıştır. Ticari yazılımlarda kaynak kodun açık olmaması, ordu, kamu kuruluşları ve özelsektör gibi üst düzey güvenlik isteyen kuruluşlar tarafından Snort'un daha çok tercih edilir olmasını sağlamaktadır.

- **Snort Sam:** Snortsam, Snort STS sisteminin bir eklentisidir. Snortsam iki ana kısımdan oluşmuştur. Bu kısımlardan biri Snort için çıkış sistemidir, diğeri de güvenlik duvarı(GD) üzerinde ajan vazifesi görecek kısımdır. Snortsam kurulduktan sonra Snort kurallarına "fwsam" anahtar kelimesi eklenmektedir. Snortsam'ın kullanımı bu anahtar kelimeyle yapılmaktadır. Snort üzerine yazılan kurallar snortsam ajanı ile GD'ye aktarılarak engellenmesi gereken paketleri durdurur.

---

[1]https://www.snort.org/snort (01.05.2014)

[2]https://www.snort.org/snort (01.05.2014)

4. YAPAY SİNİR AĞLARI

Saldırı tespit sistemleri saldırıyı yakalayabilmek için farklı yöntemler kullanmaktadır. Çalışmadakullanılan YSA kapsamında çok katmanlı algılayıcılar (ÇKA) ve Levenberg-Marquardt (LM) algoritmasıile birlikte tek katmanlı algılayıcılar(TKA), ADALINE, MADALINE modelleri detaylı olarak incelenmektedir.

4.1 Giriş

Günümüz yazılımları ile bilgisayar sistemleri her hangi bir olayı öğrenmekte, olaylar hakkında fikir yürütmekte ve diğer olaylar ile ilişkilerkurabilmektedir. Özellikle matematiksel olarak formüle edilemeyen olaylarınaraştırılmasında ve karmaşık problemlerin sezgisel bir şekilde çözülmesindebilgisayarlar kullanılmaktadır. 20.yy'ın ikinci yarısında ortaya atılan "yapay zeka" kavramı bilgisayarların sezgisel yeteneklerinin geliştirilmesiyle meydana gelmiştir. Yapay zeka bilimsel ve ticari amaçlı neredeyse tüm faaliyetlerde hızla artan sayıda kullanılmaktadır.Bu tür sistemlerin tümüne "Zeki Sistemler" denilmektedir. Yapay zeka teknolojileri kullandıkları metotlar ve teknolojiler nedeni ile farklılıklar göstermektedir(Tanrıkulu, 2009). Çalışmanın konusunda kullanılan YSA da yapay zeka teknolojilerindendir.

4.2 Yapay Sinir Ağlarının Tanımı

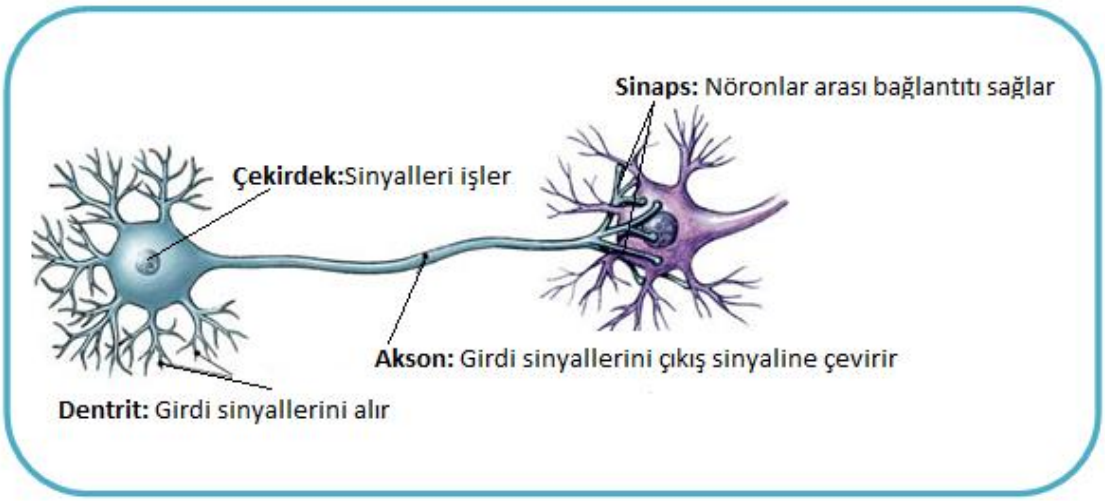
YSA, insanın düşünme ve gözlemlemeye yönelik doğal yeteneklerini oluşturan mekanizmalara benzer şekilde problemlere çözüm üretmektedir. YSA insan beyni gibi çalışarak öğrenme yetilerini taklit etmektedir. İnsanın, düşünme vegözlemleme yeteneklerini gerektiren problemlere yönelik çözümler üretebilmesiinsan beyninin denemelerden elde ettiği yaşantılardandır.

Haykin (1999) YSA'yı şöyle tanımlamıştır: "Bir sinir ağı, basit işlem birimlerindenoluşan, deneyimsel bilgileri biriktirmeye yönelik doğal bir eğilimi olan ve bunların kullanılmasını sağlayan yoğun bir şekilde paralel dağıtılmış bir işlemcidir. Bu işlemci iki şekilde beyin ile benzerlik göstermektedir:

- Bilgi, ağ tarafından bir öğrenme süreciyle çevreden elde edilir,

- Elde edilen bilgileri biriktirmek için sinaptik ağırlıklar olarak da bilinen nöronlararası bağlantı güçleri kullanılır".

Vural(2007)YSA'yı beynin bir işlevi yerine getirme yöntemini modellemek için tasarlanan bir sistem olarak tanımlamıştır. YSA, yapay sinir hücrelerinin birbirleri ile çeşitli şekillerde bağlanmasından oluşmaktadır. YSA, bir öğrenme sürecinde bilgiyi toplamakta, hücreler arasındaki bağlantı ağırlıkları ile bu bilgiyi saklamaktadır. YSA'nın öğrenme süreci, istenen amaca ulaşmak için YSA'nın ağırlıklarının yenilenmesini içeren öğrenme algoritmasını içermektedir. YSA biyolojik sinir ağlarının çalışma prensiplerini kullanarak oluşturulmuş bir modeldir. İnsan beyninde yer alan yaklaşık on milyar biyolojik sinir hücresi kendiaralarında bağlantılar kurmaktadır. Oluşturulan bağlantı sayısı yaklaşık altı trilyondur(Öztemel 2006). Bir biyolojik sinir hücresinin yapısı Şekil 4.1'de gösterilmektedir.



Şekil 4.1. Biyolojik sinir hücresi [1].

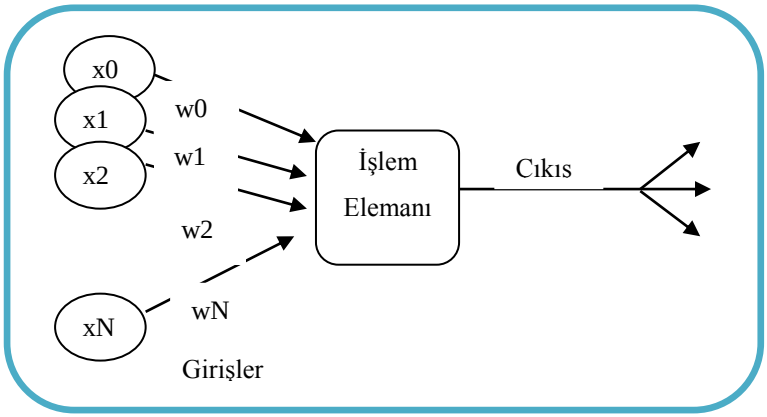
Bir biyolojik sinir hücresi; dendritler, hücre gövdesi, akson ve sinapslardan oluşmaktadır. Dendritler girdi sinyallerini toplamakta ve bilgileri sinapslara ileterek

[1]http://www. estanbul. com/merkezi-sinir-sistemi-yapisi-ve-isleyisi-188454. html#. UzVkVoVabIU adresindeki sinir hücresi şeklinden faydanılmıştır.(10.04.2014)

diğer hücrelere göndermektedir. Çevreden alınan sinyaller elektrik sinyallerine dönüştürülerek hücre gövdesinde çekirdek tarafından işlenerek çıkış sinyali oluşturulup akson aracılığıyla dendrite gönderilmektedir. Sinir sistemimilyarlarca sinir hücresi bir araya gelmesiyle oluşturmakta ve biyolojik hücrelerin yapısal özelliklerinden faydalanılarak yapay sinir ağlarını geliştirilmektedirler.

4.3 Yapay Sinir Hücresi

YSA'nın yapay sinir hücreleri (nöronları) bulunmakta ve bu hücreler işlem (proses) elemanı olarak adlandırılmaktadır. Genel özellikleri ile bir yapay sinir hücresinin yapısı Şekil 4.2' de gösterilmektedir.



Şekil 4.2. Yapay sinir hücresi.

Şekil 4.2'de girdilerx$_0$, x$_1$, x$_2$ ve x$_N$ şeklinde , girişlere verilen ağırlık değerleri ise w$_0$, w$_1$, w$_2$ ve w$_N$ olarak gösterilmiştir. Hücreye girdi olarak verilen bilgilerin önemini ve hücre üzerindeki etkisini göstermektedir.

Toplama fonksiyonu hücreye gelen net girdiyi hesaplamaya yarayan fonksiyondur. Fonksiyon gelen girdilerin kendi ağırlıklarıyla çarpımlarının toplamıdır. Kullanılan toplama fonksiyonları"E.4.1"de gösterilmektedir(Öztemel 2006).

$$\sum_{i=0}^{n} w_i x_i \qquad \text{(E.4.1)}$$

Burada $x_i$girdileri, $w_i$ise ağırlıkları, n ise bir hücreye gelen toplam girdi sayısını göstermektedir. Literatürde farklı toplama fonksiyonları kullanılmaktadır. Fonksiyonlardan bazıları Çizelge 4.1'de gösterilmektedir.

Problem için en uygun fonksiyonunu belirlemek için bulunmuş bir yöntem yoktur. Genellikle deneme yanılma yöntemi ile belirlenmektedir(Öztemel 2006).

Çizelge 4.1. Kullanılan fonksiyonlar[1].

Toplam $NET = \sum_{i=0}^{n} Xi.\ Wi$	Ağırlık değerleri girdiler ile çarpılır ve bulunan değerler birbirleriyletoplanarak Net girdi hesaplanır.
Çarpım $NET = \prod_{i=0}^{n} Xi.Wi$	Ağırlık değerleri girdiler ile çarpılır ve daha sonra bulunan değerler birbirleriyle çarpılarak Net Girdi Hesaplanır.
Minumum $NET = Min(Xi.\ Wi)$	n adet girdi içinden ağırlıklar girdilerle çarpıldıktan sonra içlerinden en küçüğü Net girdi olarak kabul edilir.
Maksimum $NET = Max(Xi.\ Wi)$	n adet girdi içinden ağırlıklar girdilerle çarpıldıktan sonra içlerinden büyüğü Net girdi olarak kabul edilir.
Çoğunluk $NET = \sum_{i=0}^{n} Sgn(Xi.\ Wi)$	n adet girdi içinden girdilerle ağırlıklar çarpıldıktan sonra pozitif ile negatif olanların sayısı bulunur. Büyük olan sayı hücrenin net girdisi olarak kabul edilir.
Kumülalif Toplam $NET = NET(eski) + \sum_{i=0}^{n} Xi.\ Wi$	Hücreye gelen bilgiler ağırlıklı olarak toplanır. Daha önce hücreye gelen bilgilere yeni hesaplanan girdi değerleri eklenerek hücrenin net girdisi hesaplanır.

Aktivasyon fonksiyonu (transfer fonksiyonu) ise, toplama fonksiyonundan gelen net girdiyi işlemden geçirerek hücrenin çıktısını üreten ve genellikle doğrusal olmayan bir fonksiyondur. Kullanılan hücre modeli çeşidine göre değişik aktivasyon fonksiyonları kullanılmaktadır.

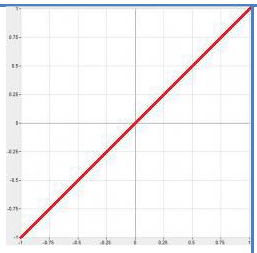
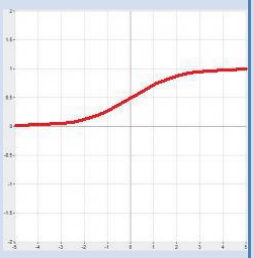
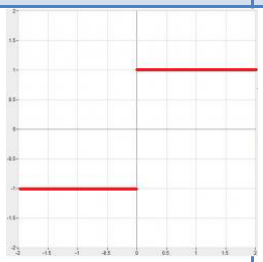
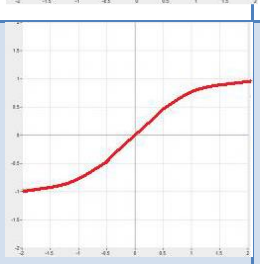
[1]http://www.ibrahimcayiroglu.com/Dokumanlar/IleriAlgoritmaAnalizi/IleriAlgoritmaAnalizi-5.Hafta-YapaySinirAglari.pdf (01.05.2014)

Çalışmamızda kullandığımız çok katmanlı algılayıcılı (ÇKA) YSA'larda hiperbolik tanjant fonksiyonu (tansig) aktivasyon fonksiyonu olarak kullanılmıştır. Tansig fonksiyonu türevi alınabilir, sürekli ve doğrusal olmayan bir fonksiyon olması nedeni ile doğrusal olmayan problemlerin çözümünde yaygın olarak kullanılmaktadır. Fonksiyonunmatematiksel tanımı "E. 4.2" de verilmektedir.

$$f(NET) = \frac{(e^{NET} + e^{-NET})}{(e^{NET} - e^{-NET})} \qquad \text{(E.4.2)}$$

Şekil 4.3'te aktivasyon fonksiyonlarından adım, eşik, logsig (logaritmik sigmoid) ve tansig (hiperbolik tanjant sigmoid) fonksiyonlarının grafikleri gösterilmektedir. İşlem elemanının çıktısı aktivasyon fonksiyonu kullanılarak hesaplanmaktadır. Hesaplanan çıktı diğer hücreye gönderilmektedir. Bunun yanında hücre çıktısını kendi girdisi olarak da gönderebilmektedir.

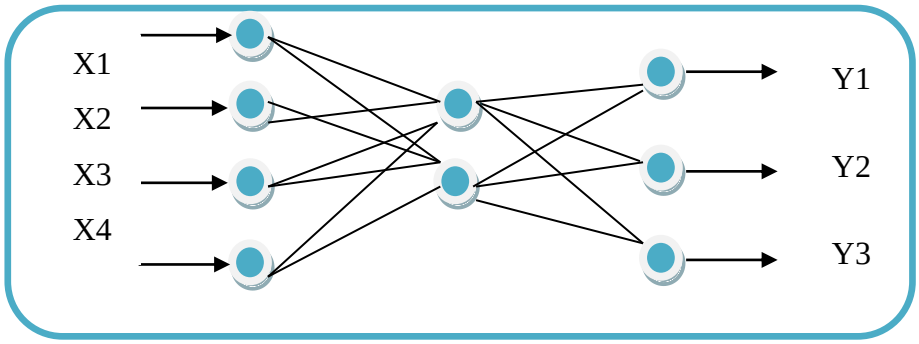
Çizelge 4.2. Aktivasyon fonksiyonları[1].

Doğrusal (Lineer)		$\boldsymbol{F(NET) = A.NET}$ (A sabit bir sayı)	Doğrusal problemler çözmek amacıyla aktivasyon fonksiyonu doğrusal bir fonksiyon olarak seçilebilir. Toplamafonksiyonundan çıkan sonuç, belli bir katsayı ile çarpılarak hücrenin çıktısı olarak hesaplanır.
Sigmoid		$\boldsymbol{F(NET)} = \frac{1}{1+e^{-NET}}$	Sigmoid aktivasyon fonksiyonu sürekli ve türevi alınabilir bir fonksiyondur. Doğrusal olmayışı dolayısıyla yapay sinir ağı uygulamalarında en sık kullanılan fonksiyondur. Bu fonksiyon girdi değerlerinin her biri için 0 ile 1 arasında bir değer üretir.
Adım (Step)		$\boldsymbol{F(NET)} = 1$ Eğer Net $>$ *Eşik* Değer $\boldsymbol{F(NET)} = 0$ Eğer Net $\leq$ Eşik Değer	Gelen Net girdinin belirlenen bir eşik değerin altında veya üstünde olmasına göre hücrenin çıktısı 1 veya 0 değerini alır.
Tanjant Hiperbolik		$\boldsymbol{F(NET)} = \frac{(e^{NET}+e^{-NET})}{(e^{NET}-e^{-NET})}$	Tanjant hiperbolik fonksiyonu, sigmoid fonksiyonuna benzer bir fonksiyondur. Sigmoid fonksiyonunda çıkış değerleri 0 ile 1 arasında değişirken hiperbolik tanjant fonksiyonunun çıkış değerleri -1 ile 1 arasında değişmektedir.
Eşik Değer		$\boldsymbol{F(NET)} = 1$ Eğer Net ≥ 1 $\boldsymbol{F(NET)} =$ NET $0 <$ *Eğer* Net < 1 $\boldsymbol{F(NET)} = 0$ Eğer Net ≤ 0	Gelen bilgilerin 0 dan küçük-eşit olduğunda 0 çıktısı, 1 den büyük-eşit olduğunda 1 çıktısı, 0 ile 1 arasında olduğunda ise yine kendisini veren çıktılar üretilebilir.
Sinüs		$\boldsymbol{F(NET)} = Sin(NET)$	Öğrenilmesi düşünülen olayların sinüs fonksiyonuna uygun dağılım gösterdiği durumlarda kullanılır.

[1]http://www.ibrahimcayiroglu.com/Dokumanlar/IleriAlgoritmaAnalizi/IleriAlgoritmaAnalizi-5.Hafta-YapaySinirAglari.pdf (01.05.2014)

4.4 Yapay Sinir Ağının Yapısı

YSA, yapay sinir hücrelerinin bir araya gelmeleriyle oluşmaktadır. Hücreler birbirleriyle katmanlar halinde ve her katman içinde paralel olarak bağlanarak tüm ağı oluşturmaktadır. Yapay sinir ağının katmanları Şekil 4.3'te gösterildiği gibi girdi, ara katmanlar ve çıktı katmanından oluşmaktadır. Girdi katmanı dış dünyadan alınan bilgileri işlem yapmadan ara katmanlara taşımakta, ara katmanlardaki işlem elemanları girdi katmanından gelen bilgileri işleyip, çıktı katmanına göndermektedir.



Şekil 4.3. Yapay sinir ağı katmanları.

4.5 Yapay Sinir Ağı Modelleri

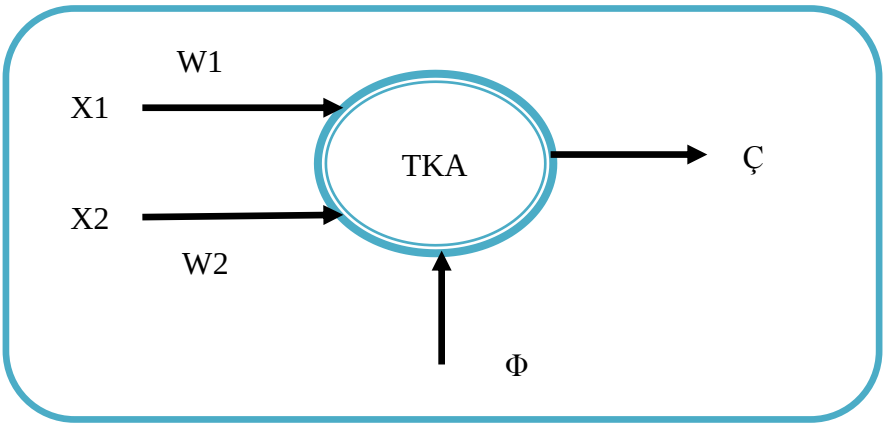
Bu kısımda ÇKA'yı ve ÇKA'nın oluşumuna katkı sağlayan ilk modeler incelenmiştir. YSA ile ilgili geliştirilen ilk modeller tek katmanlı algılayıcılar (TKA), basit algılayıcılar (perceptron) ve ADALINE/MADALINE modelleridir. Bu modeller çalışmada kullanılan ÇKA'ların temelini oluşturmaktadır.

4.5.1 Tek katmanlı algılayıcılar (TKA)

TKA girdi (x) ve çıktı (Ç) katmanlarından oluşmaktadır. Çıktı bütün girdi ünitelerine (x) bağlanmaktadır ve her bağlantının bir ağırlığı (w)vardır. İki girdi ve bir çıktıdan oluşan tek katmanlı bir yapay sinir ağı Şekil 4.4'te gösterilmektedir. TKA'larda işlem elemanlarının değerlerinin ve dolayısıyla ağın çıktısının sıfır olmasını engelleyen bir eşik değeri (Φ) vardır. Bu eşik değeri daima 1'dir. TKA çıktısı,girdi değerleriyle

ağırlık değerlerinin eşik değeri ile toplanması ile bulunmaktadır. Aktivasyon fonksiyonundan geçirilen girdi değeri, ağın çıktısı olarak hesaplanmaktadır. TKA'larda çıktı fonksiyonu 1 veya -1 değerleri alan doğrusal bir fonksiyondur.

Sınıflandırma problemlerinin çözümünde çıktı değeri 1 olanbirinci grubu, -1 olan ise ikinci grubu göstermektedir (Öztemel 2006).

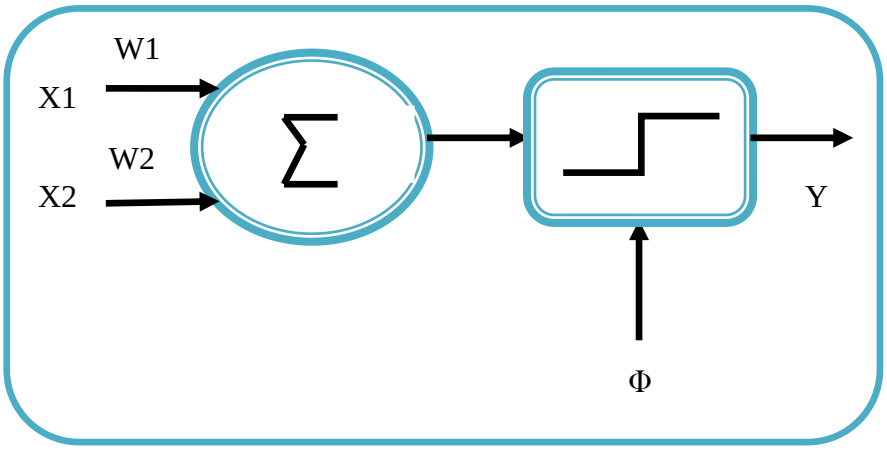


Şekil 4.4. Tek katmanlı algılayıcı modeli.

TKA'larda ikiönemlimodel bulunmaktadır. 1958 yılında Rosenblat tarafından geliştirilen basit algılayıcılar (perceptron), diğeri ise 1959 yılındaWidrow ve Hoff tarafından geliştirilen ADALINE modelidir.

4.5.1.1 Basit algılayıcılar

Basit algılayıcılar eğitilebilir tek bir yapay sinir hücresinden oluşur ve bu sinir hücresi birden fazla girdiyi alarak bir çıktı üretmesi prensibine dayanmaktadır. Çıktının hesaplanmasında eşik değer fonksiyonu kullanılmaktadır. Ağın çıktısı bir veya sıfırdan oluşan mantıksal değerler olması beklenmektedir. Ağın çıktısı beklenen çıktı değerinden farklı ise ağırlıklar ve eşik değerleri değiştirilmektedir (Tanrıkulu,2009). Basit algılayıcı ağın yapısı Şekil 4.5'te gösterilmektedir.

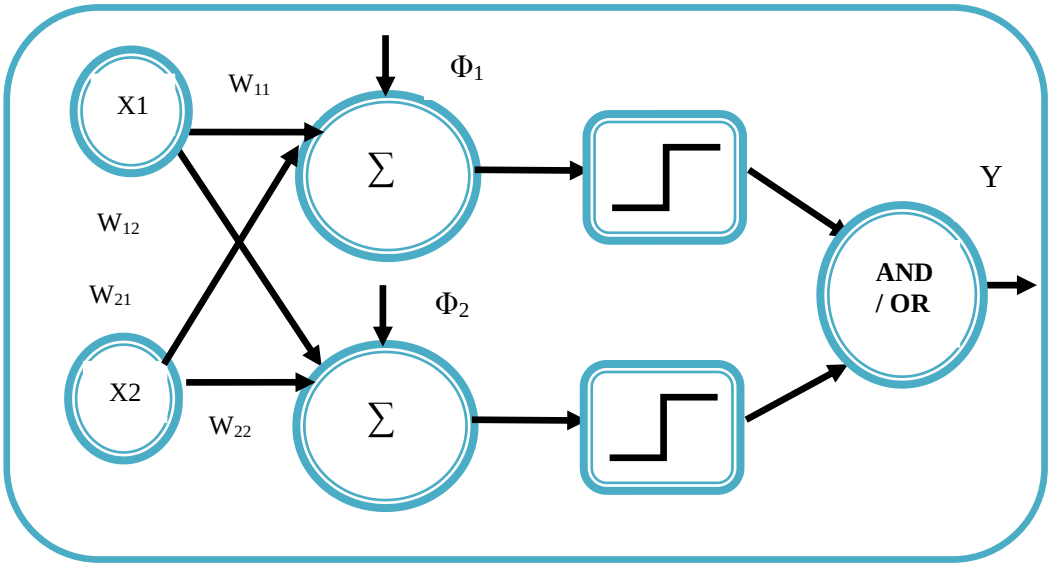


Şekil 4.5. Basit algılayıcı modeli.

4.5.1.2 ADALINE/MADALINE modeli

ADALINE/MADALINE modeli, 1959 yılında Bernard Widrow ve Marcian Hoff tarafından geliştirilmiştir. ADALINE, uyarlanabilen doğrusal elemanlar (ADAptiveLINear Elements)ağının kısaltmasıdır. MADALINE, çoklu uyarlanabilen doğrusal elemanlar(MADALINE-Multiple ADAptive LINear Elements) ağının kısaltılmasıdır ve birden fazla ADALINE işlemelemanının bir araya gelmesinden oluşmaktadır. ADALINE ağı en küçük ortalamaların karesi yöntemini kullanır. Ağın kullandığı delta öğrenme kuralı ile TKA'lardan farklılık göstermektedir. Delta öğrenme kuralı, ağın çıktısının beklenen çıktı değerine göre oluşan hatanın en aza indirilmesi içinağın ağırlık değerleri sürekli değiştirilerek gerekli döngülerin oluşturulmasıdır.

ADALINE ağının yapısı Şekil4.5'te gösterilen basit algılayıcı modelinin yapısına benzemektedir(Öztemel 2006). MADALINE ağı genel olarak iki katmandan oluşur. Her katman içinde çok sayıda ADALINE ünitesi bulunmaktadır. Şekil 4.6'da iki ADALINE ünitesinden oluşan bir MADALINE ağını göstermektedir. Tüm ADALINE işlemleri AND veya OR ile sonlandırılırlar. AND sonlandırıcısının kullanılması ile bütün ADALINE ünitelerinin değeri 1 olduğunda MADALINE ağının çıktısı 1 olmaktadır. Diğer durumda -1 (veya 0) değerini almaktadır. OR sonlandırıcısının kullanılması durumunda ADALINE ünitelerinin herhangi birisinin 1 değerini üretmesi MADALINE ağının çıktısının 1 olması için yeterlidir (Öztemel 2006).



Şekil 4.6. İki ADALINE ağından oluşan MADALINE ağ yapısı.

4.5.2 Çok katmanlı yapılar

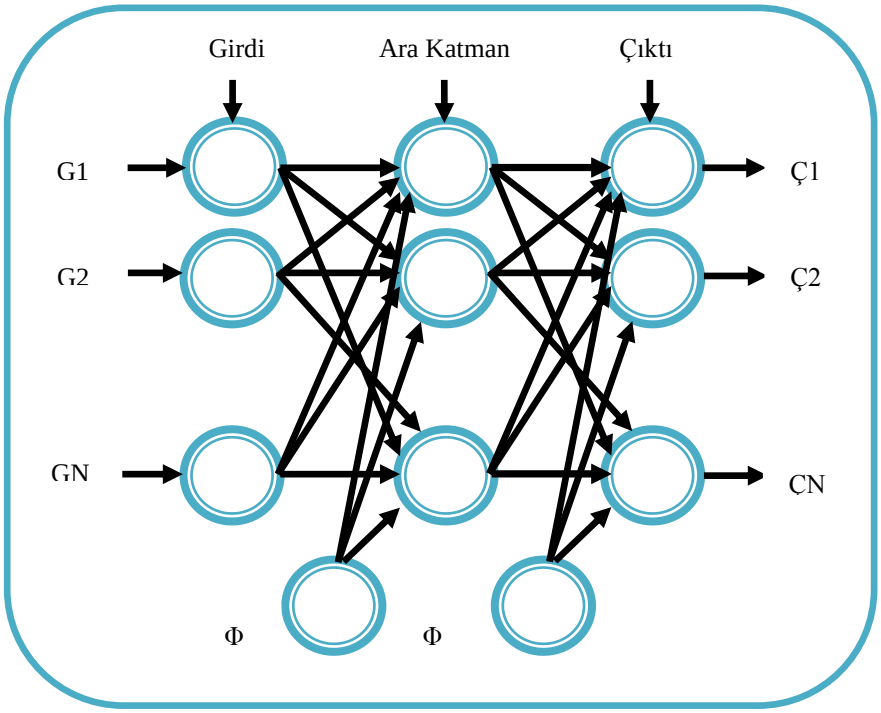
Problem çözümünde doğrusal ilişkilerin olmadığı durumlarda Rumelhart ve arkadaşları tarafından önerilen ÇKA'ların kullanılmasıyla YSA çalışmaları hız kazanmıştır. Öğrenme algoritması olarak ÇKA'larda genellikle geri yayılım (back propogation) veya hata yayma algoritmaları kullanılmaktadır.

4.5.2.1 ÇKA'nın yapısı

ÇKA'lar girdi, ara katmanlar ve çıktı katmanlarından meydana gelmektedir. Girdi katmanı, verilerin ağa girişini sağlamakta ve bir sonraki katmana veriyi aktarmaktadır. Girdi sayısında bir kısıtlama yoktur.Girdi katmanının çıktısı ara katmandaki tüm işlem elemanlarına gönderilmektedir. Ara katman, saklı katman olarak da tanımlanmaktadır. Girdi katmanından gelen bilginin işlenmesi bu katmanda olmaktadır. ÇKA ağında birden fazla ara katman ve her katmanda birden fazla işlem elemanı olabilir. Ara katmanda faaliyet gösteren işlem elemanları bir sonraki katmandaki tüm işlem elemanları ile bağlantılıdır. Çıktı katmanı ise kendinden önceki ara katmanlardan gelen bilgileri işleyerek çıktılar üretmektedir. Çıktı katmanında da ara katmanda olduğu gibi birden fazla işlem elemanı olabilir. Her işlem elemanı bir önceki

katmanda bulunan tüm işlem elemanları ile bağlantılıdır. Ancak çıktı katmanındaki her işlem elemanının sadece bir tane çıktısı vardır (Öztemel 2006).

Şekil 4.7'de ÇKA'nın temel yapısı gösterilmektedir. ÇKA'larda ara katmanının olması nedeni ile tek katmanlı ağlardan ayrılır. Ara katman sayısı arttıkça girdi verilerinden edinilen istatistiki bilgi artmakta dolayısı ile öğrenme daha doğru olabilmektedir. Ağın yapısına göre ÇKA ileri beslemeli veya geri yayılımlı bir ağ olarak tasarlanabilmektedir.



Şekil 4.7. Çok katmanlı YSA modeli.

4.5.2.2 Çok katmanlı ileri beslemeli ağ

Çok katmanlı ileri beslemeli ağ yapısı bir girdi katmanı, en az bir ara katman ya da saklı katman ve bir çıktı katmanından oluşmaktadır. Çıktı katmanındaki işlem elemanları, düzeltilebilen ağırlıklar yardımıyla, bir önceki ara katmandaki işlem elemanlarından bilgi almaktadır. Tek katmanlı ileri beslemeli YSA'lar, sadece doğrusal sınıflandırabilir fonksiyonları hesaplayabilmektedir. En az bir ara katmana sahip olan ileri beslemeli ağlar ile çok daha karmaşık fonksiyonlar hesaplanabilmektedir(Bolat ve

Kalenderli 2007). Girdi katmanındaki eğitim veri seti doğrudan birinci ara katmana uygulanmaktadır.

Birinci arakatmanın çıktısı, ikinci ara katmana girdi olarak verilmekte ve bu işlem ağın çıktısına kadar devam etmektedir. Katmanlardaki işlem elemanlarına yapılan girdiler bir önceki katmanın çıktılarıdır. Çıktı katmanındaki işlem elemanları girişte uygulanan verilerin ağda oluşturduğu toplam yanıtı vermektedir(Bolat ve Kalenderli 2007).

4.5.2.3 Geri yayılım ağı ve algoritması

Geri yayılım algoritmaları ÇKA tipi YSA'ların eğitiminde oldukça yaygın olarak kullanılmakta ve katmanlar arasında mevcut olanağırlıkların ağ çıkışında oluşan hatayı en aza indirecek şekilde düzenlenmesini amaçlamaktadır. Girdiler ve beklenen çıktı bilgileri ağı eğitmek amacıyla kullanılmaktadır. Girdiler olayın motifini belirleyen veri setleri olarak ağın girdi katmanına verildikten sonra ara katmanlardan geçerek sondaki çıktı katmanına ağırlıklar yardımı ile ulaşmaktadır. Ağdaki her bir işlem elemanı, kendisinde sonlanan ağırlık değerlerinin aritmetik toplamını aldıktan sonra sonucu kendinden sonraki katmanın tüm işlem elemanlarına ulaştırmaktadır(Tanrıkulu, 2009). Bu değerler kullanılan aktivasyon fonksiyonuna bağlı olarak oluşturulmaktadır.

Altun vd. (2007) çalışmasında katmanlar arasındaki ağırlıkların yenilenmesini sigmoid aktivasyon fonksiyonu kullanarak aşağıda gösterildiği gibi elde etmiştir. Çıkış katmanındaki her bir işlem elemanı için çıktı bilgisi;

$$O_k = \frac{1}{1+e^{-net_k}} \qquad \text{(E.4.3)}$$

şeklinde tanımlandığını varsaydığımızda "E.4.3"deki O_kçıktı katmanının aktivasyon değerini göstermektedir.

$$\text{net}_k = \sum_j W_{jk} O_j \qquad \text{(E.4.4)}$$

Benzer şekilde ara katman için aktivasyon değerlerinin ifadesi aşağıdaki gibi bulunmaktadır.

$$O_j = \frac{1}{1+e^{-net_J}} \quad \text{(E.4.5)}$$

$$net_j = \sum_i W_{ij} O_i \quad \text{(E.4.6)}$$

Ağırlıkların yenilenmesi

$$W_{jk} = W_{jk} + \Delta W_{jk} \quad \text{(E.4.7)}$$

eşitliği ile gerçekleştirilmektedir. Burada W_{jk} ağırlık yenileme değeridir. Geri yayılım algoritmasında ortalama kare hatası olarak bilinen hata kriteri kullanılabilmektedir.

$$E = \frac{1}{2} \sum_p \sum_k (t_{pk} - o_{pk})^2 \quad \text{(E.4.8)}$$

Hataların karesi alınarak beklenen değerden uzak olan çıkış değerlerinin toplam hatayı oluşturması sağlanmaktadır. Hatayı en aza indirmek için hatanın ağırlıklara olan bağımlılığı hesaplanmakta ve gradyana bağlı olarak ağırlıklar yenilenmektedir.

$$\Delta W_{jk} = -\eta(\partial E / \partial W_{jk}) \quad \text{(E.4.9)}$$

Zincir kuralı kullanarak diferansiyel denklem çözümü aşağıdaki E.4.10 eşitliğinde verildiği gibi elde edilmektedir.

$$(\partial E / \partial W_{jk}) = \delta_k o_j \quad \text{(E.4.10)}$$

Bu eşitlik bir önceki "E.4.9"da yerine konursa ağırlık yenileme değeri aşağıdaki gibi elde edilmektedir.

$$\Delta W_{jk} = -\eta(\delta_k o_j) \quad \text{(E.4.11)}$$

$$\Delta W_{ij} = -\eta(\delta_i o_i) \quad \text{(E.4.12)}$$

Burada δ_k ve δ_jsırası ile çıkış ve ara katman için hata terimi, η ise öğrenme oranını göstermektedir. Çıktı katmanı için hata terimi

$$\delta_{\mathrm{k}} = (\mathrm{t_k} - o_k) f^{'}(net_k) \quad \text{(E.4.13)}$$

ve ara katman için hata terimi ise aşağıdaki gibi hesaplanmaktadır.

$$\delta_j = f^{'}(net_k) \sum_k \delta_{ki} W_{kj} \quad \text{(E.4.14)}$$

Yukarıdaki ifadelerde $f^{'}$ katmanlar arası sigmoid aktivasyon fonksiyonunun türevidir. Bağlantılar için algoritmanın üreteceği ağırlık yenileme tek tek incelendiğinde ağırlık değerlerinin her bir katman için işlem elemanı aktivasyon seviyeleri dikkate alınarak verilmesi daha uygundur.

$$\Delta \mathrm{W_{jk}} = -\eta(O_k^{'} \Delta O_k O_j) \quad \text{(E.4.15)}$$

$$\Delta \mathrm{W_{ij}} = -\eta O_j^{'} \sum_k O_k^{'} W_{jk} \Delta O_k O_j \quad \text{(E.4.16)}$$

$$\delta_{\mathrm{k}} = O_k(1 - O_k)(t_k - O_k) \quad \text{(E.4.17)}$$

$$\delta_{\mathrm{j}} = O_j(1 - O_j) \sum_k \delta_k W_{jk} \quad \text{(E.4.18)}$$

$$O_k = f(\sum_j O_j W_{jk}) \quad \text{(E.4.19)}$$

$$O_j = f(\sum_i O_i W_{ij}) \quad \text{(E.4.20)}$$

$$\Delta O_k = (t_k - O_k) \quad \text{(E.4.21)}$$

Eşitliklerde kullanılan terimlerin tanımları aşağıda verilmektedir.

f : sigmoid aktivasyon fonksiyonu

δ : delta hata ifadesi

η : öğrenme oranı

t_k: beklenen değer

O_k: çıktı aktivasyon seviyesi

O'_k: çıkış aktivasyon seviyesinin türevi

W_{jk}: ara katman-çıktı katmanı arasında ağırlıklı bağlantı

ΔW_{jk} : ara katman çıktı katmanı arasında ağırlıklı bağlantılar için ağırlık yenileme değeri

W_{ij}: girdi ve ara katman arasında ağırlıklı bağlantı

ΔW_{ij}: girdi ve ara katman arasında ağırlıklı bağlantılar için ağırlık yenileme değeri

Elde edilen ifadelerden YSA'da girdi değerlerinin ağırlık değerlerinin belirlenmesinde ve işlem elemanlarının eğitiminde önemli rol oynadığı görülmektedir (Altun vd. 2007).

4.6 Yapay Sinir Ağının Öğrenmesi

YSA'da ağırlık değerlerinin belirlenmesi işlemi ağınperformansını etkileyen önemli bir etmendir. Ağın ağırlık değerlerinin belirlenmesine, "ağın eğitilmesi"denilmektedir. Eğitilen ağın problem ile ilgili doğru sonuçlar veren ağırlık değerlerine ulaşmasına da "ağın öğrenmesi" denilmektedir. Öğrenme modelleri danışmanlı ve danışmansız öğrenme modelleri olarak ikiye ayrılmaktadır.

4.6.1 Danışmanlı öğrenme

Danışmanlı öğrenme YSA'nın eğitilmesinde kullanılan en yaygın yöntem olduğu söylenebilmektedir. YSA'dan elde edilen çıktı değeri ile beklenen çıktı değerinin karşılaştırılması esasına dayanmaktadır. İlk önce ağ tarafından rastgele ağırlık değerleri atanmakta, ağırlıklar her döngüde değiştirilerek beklenen çıktı ile gerçek çıktı arasında daha yakın değerler üretilmektedir. Elde edilen değerler beklenen çıktı değerine yakın ya da kabul edilebilir değerlere ulaşması ağın öğrenmesinin gerçekleştiği anlamına gelmektedir.

Çalışılan sinir ağı öğrenmeye başlamadan önce eğitilmesi gerekmektedir. Eğitim, girdi ve çıktı verilerinin ağa sunulmasını içermektedir. Veriler, eğitim seti olarak tanımlanmaktadır. Yapılacak işlemin büyüklüğüne göre veri seti kullanılmalıdır. Ağdan önemli özellikleri ve ilişkileri öğrenmesi isteniyorsa, eğitim setinin gereksinim ölçüsünde büyük olması gerekmektedir. Ağdan örnek bir olay etrafında eğitilmesi isteniyorsa girdi, veri seti ve başlangıç ağırlık değerleri çok dikkatli seçilmelidir.

Ağı, başarılı bir şekilde eğitmek ve öğrenmenin gerçekleşmesi için, girdi ve çıktı verilerinin ağa nasıl verileceği önemli bir konudur. YSA'lar sadece sayısal girdi verileri

ile çalışabilmektedir. YSA'da kullanılacak verilerin sayısal değerlere dönüştürülerek ağın anlayabileceği duruma getirilmesi gerekmektedir. Eğitilen ağın performansı için eğitim veri setleri ile eğitildikten sonra test verileri ile denenmesi gerekmektedir. Test aşamasında ağın girdileri ezberlemediği ve bir uygulama içindeki genel örnekleri öğrendiğini göstermesi önemlidir(Anderson, McNeill, 1992).

Widrow ve Hoff tarafından geliştirilen delta kuralı ve Rumelhart ve McClelland tarafından geliştirilen genelleştirilmiş delta kuralı algoritması danışmanlı öğrenmeye örnek olarak verilebilmektedir.

4.6.1.1 Levenberg-Manquardt(LM)

LM algoritması, maksimum komşuluk fikri üzerine kurulmuş bir en küçük kareler hesaplama metodudur(Sağıroğlu, 2003). Bu algoritma, Gauss-Newton ve En Dik İniş(Steepest Descent) algoritmalarının en iyi özelliklerinden oluşmakta ve bu iki metodun kısıtlamalarını ortadan kaldırmaktadır. Genel olarak bu metot yavaş yakınsama probleminden etkilenmemektedir(Güven, 2007).

LM algoritmasında; $E(\underset{\sim}{w})$ 'nin hata fonksiyonu olduğu düşünülürse, *m* adetçıkış nöronu için hata terimi $e_i^2(\underset{\sim}{w})$ aşağıda verilmiştir(Sağıroğlu, 2003).

$$E(\underset{\sim}{w}) = \sum_{i=1}^{m} e_i^2(\underset{\sim}{w}) = \| f(\underset{\sim}{w}) \|^2 \qquad \text{(E.4.22)}$$

$$e_i^2(\underset{\sim}{w}) = (y_i - yd_i)^2 \qquad \text{(E.4.23)}$$

$$(J_k^T J_k + \lambda I)\delta \underset{\sim}{w}_k = -J_k^T f(\underset{\sim}{w}_k) \qquad \text{(E.4.24)}$$

Burada, amaç fonksiyonu *f(.)* ve onun Jakobiyeni *J'*nin bir noktada $(\underset{\sim}{w})$ bilindiği farzedilmektedir. LM öğrenme algoritmalarında hedef, parametre vektörü $(\underset{\sim}{w})$'nin,

$E(\underset{\sim}{w})$ 'yi en azyapacak şekilde bulunmasıdır. LM algoritmasının kullanılmasıyla yeni vektör 'dan eşitlik "E.4.24"yardımıyla hesaplanabilmektedir.

$$\underset{\sim}{w}_{k+1} = \underset{\sim}{w}_k + \delta \underset{\sim}{w}_k \qquad \text{(E.4.25)}$$

burada $\delta \underset{\sim}{w}_k$, eşitliğinden faydalanılarak hesaplanmaktadır. Eşitlik "E.4.25" de;

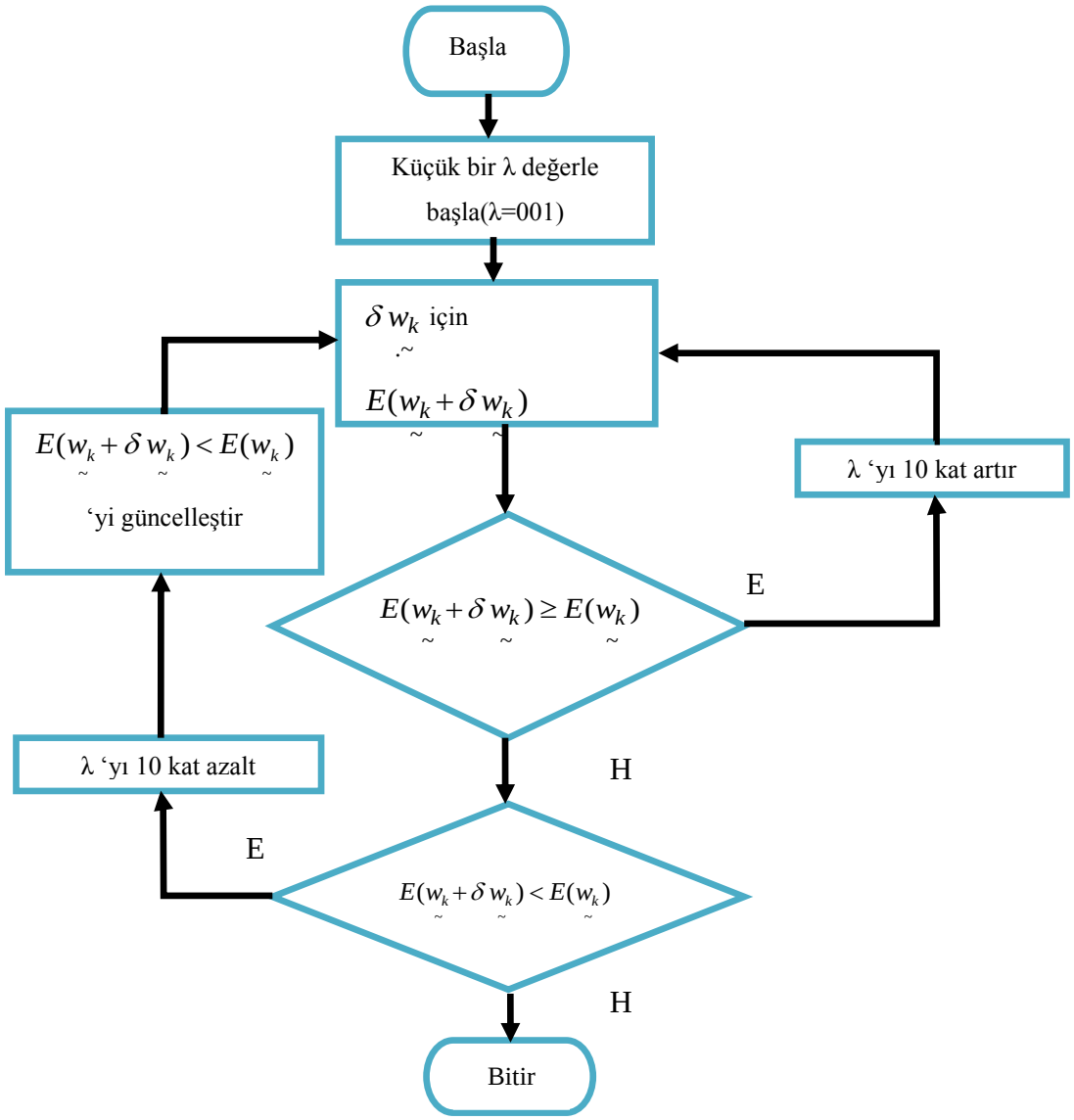
J_k :f 'in $\underset{\sim}{w}_k$ değerlendirilmiş Jakobyeni,

λ:Marquardt parametresi,

I: birim veya tanımlama matrisidir.

Hedef çıkışı hesaplamak için bir YSA'nın ağırlıklarının LM öğrenme algoritmasıkullanılarak eğitilmesi ağırlık dizisi $(\underset{\sim}{w})$ ya bir başlangıç değerinin atanması ile başlamakta ve hataların karelerinin toplamı e_i^2 'nin hesaplanmasıyla devam etmektedir. Her e_i^2 terimi, hedef çıkış (y) ile gerçek çıkış (y_d) arasındaki farkın karesini ifade etmektedir. Bütün veri kümesi için e_i^2 hata terimlerinin tamamının elde edilmesiyle, ağırlık dizileri Şekil 4.8'deki gösterilen hesaplama akışı içerisinde uygulanmaktadır.

Levenberg-Marquardt algoritmasının işleyişi Şekil 4.8'de gösterilmektedir.



Şekil 4.8. Levenberg-Marquardt algoritmasının işleyişi.

4.6.2 Danışmansız öğrenme

Danışmansız öğrenme yöntemi kullanan YSA'lar her hangi bir olayı kendi kendine öğrenmekte ve organize olmaktadır. Bu ağlar "kendini örgütleyen ağlar" olarak da adlandırılmaktadır. Danışmansız öğrenen YSA'lar girdi ağırlıklarını belirlemek için dışarıdan bir etkiye ihtiyaç duymazlar ve girdi verilerinde bir düzen arayarak ağın fonksiyonuna göre kendi performanslarını ayarlamaktadır. Danışmansız öğrenen YSA'lar gelecekte programlanabilen makinelerin yazılım sisteminde kullanılarak insanoğluna, zaman, işgücü ve kaynak tasarrufu konusunda faydalar sağlaması beklenmektedir.

5. VERİ KÜMELERİ

5.1 Saldırı Veri Kümeleri

Bir STS üzerine çalışmalar yaparken veri kümelerinin uygulamalar için genel kabul görmüş olması gerekmektedir. Literatürde bakıldığında STS kullanılan veri kümeleri ya da diğer tabirle veri setleri üzerinde yapılan çalışmalarda IDEVAL, IDEVAL98 ve KDD'99 veri setleri üzerinde yapıldığı görülmüştür. Günümüzde halen kullanılmakta olan IDEVAL ve KDD'99 veri kümeleri hakkında bilgiler verilmiştir.

5. 1. 1. IDEVAL veri kümeleri

MIT Lincoln Laboratuarlarında DARPA ve AFRL(Hava Kuvvetleri Araştırma Laboratuarı - Air Force Research Projects Agency) desteğiyle yapılan çalışmaların sonucunda, STS değerlendirilmesi ve karşılaştırılması için IDEVAL (Intrusion Detection Evaluation) adı verilen ilk standart veri kümesi gövdesi oluşturulmuştur. Oluşturulan gerçek zamanlı olmayan veri kümeleri 1998 ve 2000 yılları arasında değerlendirilmeler sonucu araştırmacılar tarafından kullanılmaya uygun hale getirilmiştir.

DARPA saldırı tespit değerlendirme çalışmaları 1998 ve 1999 DARPA IDEVAL veri kümeleri olmak üzere iki veri kümesi elde edilmiştir (MIT, 2013). Bu veri kümelerinin haricinde 2000 yılında özel senaryolar için üç farklı veri kümesi daha oluşturulmuş ve bu veri kümeleri araştırmacıların kullanımına sunulmuştur(MIT, 2013).

DARPA 2000 veri kümeleri sadece özel senaryoları içerdiğinden çalışmada kullanılmamıştır. IDEVAL 1998 ve 1999 veri kümeleri, gerçek zamanlı ve gerçek zamanlı olmayan değerlendirme olmak üzere iki kısımdan oluşmaktadır. Lincoln Laboratuarlarında oluşturulan veri kümeleri, STS'lerin gerçek zamanlı olmayan (off-line)değerlendirilmesinde kullanılır. IDEVAL 1998 ve 1999 değerlendirmesinin gerçek zamanlı kısmı hakkında ayrıntılı bilgi AFRL'den elde edilebilmektedir.

5.1.2 1998 DARPA

1998 DARPA veri kümesi, 1998 IDEVAL veri kümesinin gerçek zamanlı olmayan değerlendirme kısmı için geliştirilmiştir (MIT, 2013)

1998 IDEVAL eğitim veri kümeleri yedi haftalık verilerden oluşmaktadır. Bu verilerin oluşturulması sırasında kullanılan simulasyon ortamı ağında iç ve dış olmak üzere iki ortam oluşturulmuştur. İçerideki sunucularda denetim verileri ve ağ paketlerinin toplanması sağlanmıştır. Veri setinde toplanan saldırılar dört ana başlıkta toplanmıştır. Bunlar Çizelge 5.1'de gösterildiği gibi DoS–U2R-R2L ve Probe saldırılarıdır.

Çizelge 5.1. Test veri kümesinde yer alan ataklar(Güven, 2007).

	Solaris Sunucu	**SunOS**	**Linux**	**Cisco Yönlendirici**
DoS	back neptune ping of death smurf syslog land apache2 mailbomb process table UDP storm	back neptune ping of death smurf land apache2 mailbomb process table UDP storm	back neptune ping of death smurf teardrop land apache2 mailbomb process table UDP storm	snmp getattack
U2R	eject ffbconfig fdformat ps	loadmodule ps	perl xterm	
R2L	Dictionary ftp-write guest phf ftp-write httptunnel xlock xsnoop	dictionary ftp-write guest phf httptunnel xlock xsnoop	dictionary ftp-write guest imap phf httptunnel named sendmail xlock xsnoop	
Probe	ip sweep nmap port sweep satan mscan saint	ip sweep nmap port sweep satan mscan saint	ip sweep nmap port sweep satan mscan saint	ip sweep nmap port sweep satan mscan saint

5.1.3 1999 DARPA

DARPA 1999 veri kümesi, 1998'de geliştirilen veri kümesinin değiştirilmesi ile oluşturulmuştur. Farklı bazı atakların da veri kümesine dahil olması için böyle bir değişikliğe ihtiyaç duyulmuştur. DARPA 1998 veri kümesinin geliştirilmesinde kullanılan ağ yapısına eklenen bu yeni özellikler olarak NT iş istasyonları ve saldırıları, iç saldırılar, NT denetleme verileri eklenmesi sayılabilir. Bununla birlikte veri kümelerinde yapılan değişiklikler olarak eğitim verilerinde saldırı yapılmayan günler, eğitim ve test verisinde az örtüşen saldırılar ile saldırı sıralandırmasının tekrar kontrol edilmesi sayılabilir.

Geliştirilen DARPA veri kümelerinin asıl amacı olan, STS'lerin değerlendirilmesi işlemleri, simülasyon ağından toplanan ağ trafiği ve günlük kayıtları kullanılarak, gerçek zamanlı olmayan değerlendirme aşamasında gerçekleştirilmiştir. Gerçek zamanlı test için STS'ler AFRL'ye gönderilmiştir. Bu sistemler AFRL ağ test yatağına eklenmiş ve gerçek zamanlı olarak, normal aktiviteler arasında atak oturumları belirlenmeye çalışılmıştır (Güven, 2007).

DARPA veri kümesi için oluşturulan senaryo şekli EKLER kısmındadır.

5.1.4 KDD'99

KDD'99 veri kümesi 1999 yılında DARPA veri kümesinin bazı önişlemlerden geçirilmesi ile elde edilmiş 41 özellikten oluşur. Yapılan veri madenciliği kupasında yarışmacılardan DARPA99 veri setinin geliştirilerek daha fazla saldırı içerecek bir veri seti tasarlanması istenmiştir. Bu veri kümesinin tasarlanmasının amacı, son yıllarda farklı tekniklerle gerçekleştirilmek istenen STS'ler için eğitim vetest işlemlerinde kolaylık sağlamaktır. KDD'99 veri kümesi ile eğitim ve test sonuçlarının daha hızlı alınabilmesi ise araştırmacılar için büyük bir avantajdır. KDD'99 veri kümesi için oluşturulan senaryo şekli EKLER kısmındadır.

KDD'99 veri kümelerinde, 9 temel ve 32 adet türetilmiş olmak üzere toplamda 41 tane özellikten oluşan bir özellik haritası çıkarılmıştır. Bu 41 özellik üç temel kategoriye ayrılarak ifade edilmiştir(Güven, 2007).

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

Aşağıdaki çizelgelerdeüç kategori ve kategorileri içerisindeki veri özellikleri gösterilmiştir.

Çizelge 5.2'de saldırı veri kümelerinde TCP protokolünün özelliği olan saldırı servis tipi, protokol tipi, veri kaynak bilgilerinin bulunduğu 9 özellik mevcuttur.

Çizelge 5.2. İçerik özellikleri(Güven, 2007).

Özellik adı	Tanım	Tip
duration	Bağlantı uzunluğu	sürekli
protocol_type	Protokol tipi	ayrık
service	Servis tipi	ayrık
src_bytes	Kaynaktan hedefe veri	sürekli
dst_bytes	Veri byte sayısı	sürekli
flag	Bayrak	ayrık
land	Kaynak ve hedef IP aynı ise 1 değilse 0	ayrık
Wrong_fragment	Yanlış parçalama	sürekli
urgent	Acil paket sayısı	sürekli

Çizelge 5.3'de sunucuya bağlı özellikler olarak uzaktan erişim, kabuk(shell) bağlantı bilgileri, giriş(login) sayılarını göstermektedir.

Çizelge 5.3. Sunucu tabanlı trafik bilgileri(Güven, 2007).

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

Çizelge 5.4'de zamana bağlı özellik olarak sunucuda aynı zamanda aynı servisin çalışması durumunda elde edilen bilgileri göstermektedir.

Çizelge 5.4. Zamana bağlı özellikler(Güven, 2007).

Özellik adı	Tanım	Tip
count	Aynı sunucuya önceki iki bağlantıyla aynı bağlantıların sayısı	sürekli
serror_rate	"SYN" hata bağlantılarının yüzdesi	sürekli
rerror_rate	"REJ" hata bağlantılarının yüzdesi	sürekli
same_srv_rate	Aynı servise bağlantıların yüzdesi	sürekli
diff_srv_rate	Farklı servislere bağlantıların yüzdesi	sürekli
srv_count	Aynı servise önceki iki bağlantıyla aynı bağlantıların sayısı	sürekli
srv_serror_rate	"SYN" hata bağlantılarının yüzdesi	sürekli
srv_rerror_rate	"REJ" hata bağlantılarının yüzdesi	sürekli
srv_diff_host_rate	Farklı servislere bağlantıların yüzdesi	sürekli

Çizelge 5.5'de KDD'99 %10 luk veri setinde bulunan saldırı tipleri ve bu tiplere ait saldırı sayılarını göstermektedir.

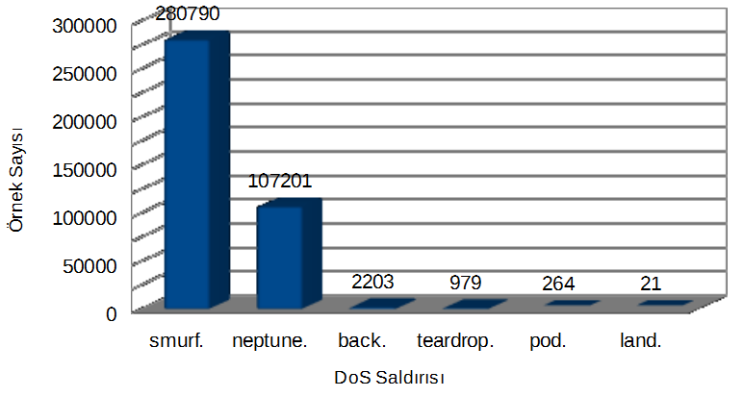
Çizelge 5.5. KDD'99 %10 luk veri setinde bulunan saldırı tipleri ve örnek sayıları.

Atak	Örnek sayısı	Kategori	Sayısal Degeri
smurf.	280790	DoS	1
neptune.	107201	DoS	1
back.	2203	DoS	1
teardrop.	979	DoS	1
pod.	264	DoS	1
land.	21	DoS	1
Toplam	**391458**		
normal.	**97277**	Normal	0
satan.	1589	Probe	2
ipsweep.	1247	Probe	2
portsweep.	1040	Probe	2
nmap.	231	Probe	2
Toplam	**4107**		
warezclient.	1020	R2L	3
guess_passwd.	53	R2L	3
warezmaster.	20	R2L	3

Çizelge 5.5. KDD'99 %10 luk veri setinde bulunan saldırı tipleri ve örnek sayıları (devam ediyor).

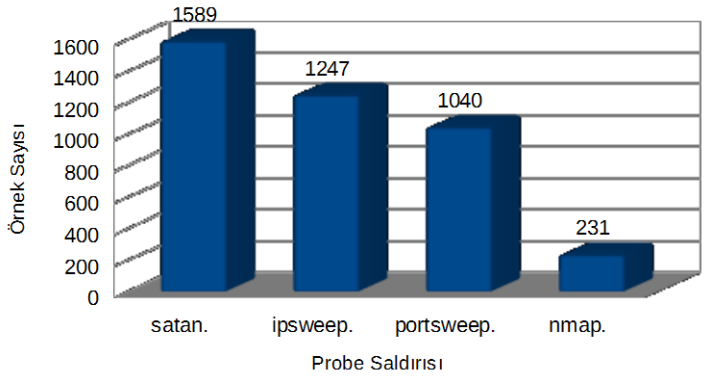
imap.	12	R2L	3
ftp_write.	8	R2L	3
multihop.	7	R2L	3
phf.	4	R2L	3
spy	2	R2L	3
Toplam	**1126**		
buffer_overflow.	30	U2R	4
rootkit.	10	U2R	4
loadmodule.	9	U2R	4
perl.	3	U2R	4
Toplam	**52**		
Genel Toplam	**494020**		

Şekil 5.1'deki grafikteKDD'99 %10 luk kısmının veri kümesinde kullanılan saldırı tiplerinden DoS saldısı ve örneklemelerini göstermektedir.



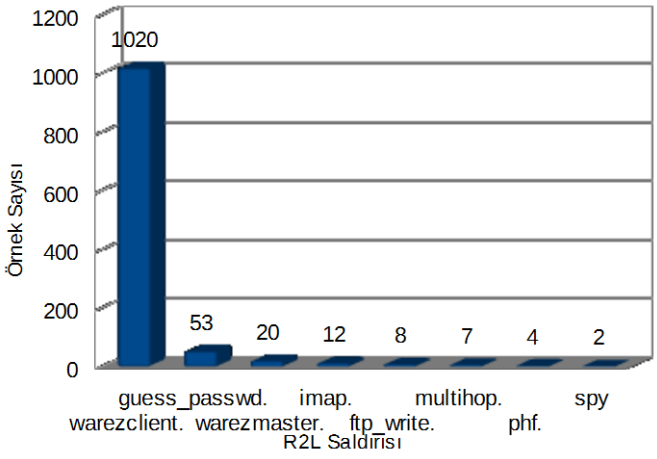
Şekil 5.1. KDD'99 veri kümesindeki Dos saldırı tipleri ve örnek sayıları.

Şekil 5.2'deki grafikteKDD'99 %10 luk kısmının veri kümesinde kullanılan saldırı tiplerinden Probe saldısı ve örneklemelerini göstermektedir.



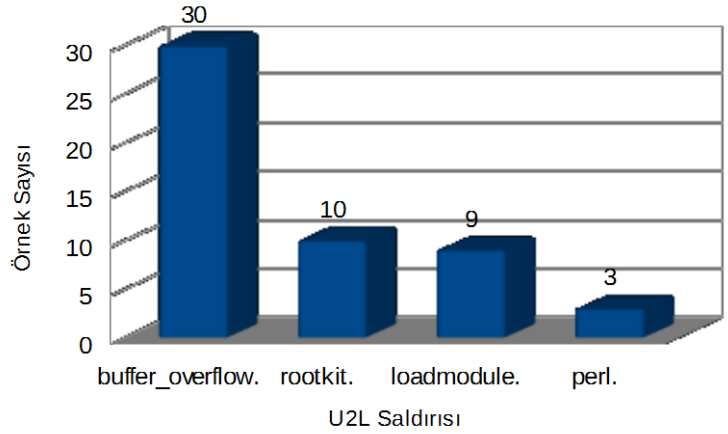
Şekil 5.2. KDD'99 veri kümesindeki Probe saldırı tipleri ve örnek sayıları.

Şekil 5.3'deki grafikte KDD'99 %10 luk kısmının veri kümesinde kullanılan saldırı tiplerinden R2L saldısı ve örneklemelerini göstermektedir.



Şekil 5.3. KDD'99 veri kümesindeki R2L saldırı tipleri ve örnek sayıları.

Şekil 5.4'deki grafikte KDD'99 %10 luk kısmının veri kümesinde kullanılan saldırı tiplerinden U2L saldısı ve örneklemelerini göstermektedir.



Şekil 5.4. KDD'99 veri kümesindeki U2L saldırı tipleri ve örnek sayıları.

Çizelge 5.6'da KDD'99 %10 luk veri setinde bulunmayan saldırı tipleri ve örnek sayıları verilmiştir.

Çizelge 5.6. KDD'99 %10 luk veri setinde bulunmayan saldırı tipleri ve örnek sayıları.

Atak	**Örnek sayısı**	**Kategori**
apache	794	Dos
mailbomb	5000	Dos
processtable	759	Dos
udpstorm	2	Dos
mscan	1053	Probe
saint	736	Probe
httptunnel.	138	R2L
named	17	R2L
sendmail	17	R2L
snmpgetattack	1040	R2L
xlock	9	R2L
xsnoop	4	R2L
ps	16	U2R
xterm	13	U2R
Toplam	**9598**	

5.2 Veri Kümelerinin Uygulamaya Uygun Hale Getirilmesi

KDD'99 veri kümesinden %10 luk kısmı olan örnek saldırı kümeleri içeriğindeki

42 alan bilgisi LibreOffice programıyla açılmış ve her sütuna bir özellik gelecek biçimde Şekil 5.5'de listelenmiştir. [1]

```
0 tcp http SF 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0. 00 0. 00 0. 00 0. 00 1. 00 0. 00 0. 00 49 49
1. 00 0. 00 0. 02 0. 00 0. 00 0. 00 0. 00 0. 00 normal.
0 tcp http SF 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0. 00 0. 00 0. 00 0. 00 1. 00 0. 00 0. 00 59 59
1. 00 0. 00 0. 02 0. 00 0. 00 0. 00 0. 00 0. 00 normal.
0 icmp ecr_i SF 1032 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 511 511 0. 00 0. 00 0. 00 0. 00 1. 00 0. 00 0. 00 255
255 1. 00 0. 00 1. 00 0. 00 0. 00 0. 00 0. 00 0. 00 smurf.
0 icmp ecr_i SF 1032 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 511 511 0. 00 0. 00 0. 00 0. 00 1. 00 0. 00 0. 00 255
255 1. 00 0. 00 1. 00 0. 00 0. 00 0. 00 0. 00 0. 00 smurf.
0 icmp ecr_i SF 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 13 13 0. 00 0. 00 0. 00 0. 00 1. 00 0. 00 0. 00 97
13 0. 13 0. 04 0. 13 0. 00 0. 00 0. 00 0. 00 0. 00 pod.
0 icmp ecr_i SF 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 14 14 0. 00 0. 00 0. 00 0. 00 1. 00 0. 00 0. 00 98
14 0. 14 0. 04 0. 14 0. 00 0. 00 0. 00 0. 00 0. 00 pod.
0 tcp private S0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 254 1 1. 00 1. 00 0. 00 0. 00 0. 00 0. 06 0. 00 255 1
0. 00 0. 07 0. 00 0. 00 1. 00 1. 00 0. 00 0. 00 neptune.
0 tcp private S0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 264 6 1. 00 1. 00 0. 00 0. 00 0. 02 0. 06 0. 00 255 6
0. 02 0. 06 0. 00 0. 00 1. 00 1. 00 0. 00 0. 00 neptune.
```

Şekil 5.5. KDD'99 %10 luk veri setinde etiketlenmiş biçiminden kesit.

Listelenen saldırı veri kümelerinin bazılarında sayısal olmayan özellikler işlenmesi için sayısallaştıma yoluna gidilmiştir. Yöntem olarak, Güven'in(2007) yaptığı tez çalışmasındaki sayısallaştırma metedolojisi kullanılmıştır. Herbir saldırı çeşidine ve protokol ve kod bilgilere karşılık sayısal değerler atanmıştır.

Çizelge 5.7'de Saldırı tiplerinin sayısal kod tanımlamaları gösterilmektedir.

Çizelge 5.7. Saldırı tiplerinin sayısal kod tanımlamaları(Güven, 2007).

Saldırı	**Sayısal Degeri**	**Saldırı**	**Sayısal Degeri**
normal	0	teardrop	20
back	1	warezclient	21
buffer_overflow	2	warezmaster	22
ftp_write	3	apache2	23
guess_passwd	4	named	24
imap	5	saint	25
ipsweep	6	sendmail	26
land	7	snmpgetattack	27
loadmodule	8	udpstorm	28
multihop	9	xlock	29
neptune	10	xsnoop	30
nmap	11	mailbomb	31
sperl	12	processtable	32

[1]https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html adresinden indirilebilir.(04.01.2014)

Çizelge 5.7. Saldırı tiplerinin sayısal kod tanımlamaları(Güven, 2007) (devam ediyor).

phf	13	mscan	33
pod	14	httptunnel	34
portsweep	15	ps	35
rootkit	16	xterm	36
satan	17	snmpguess	37
smurf	18	worm	38
spy	19	sqlattack	39

Çizelge 5.8'da Servis değerlerinin sayısal kod tanımlamaları gösterilmektedir.

Çizelge 5.8. Servis değerlerinin sayısal kod tanımlamaları(Güven, 2007).

Servis	**Sayısal Degeri**	**Servis**	**Sayısal Degeri**	**Servis**	**Sayısal Degeri**
http	0	whois	22	iso_tsap	44
smtp	1	domain	23	hostnames	45
finger	2	login	24	csnet_ns	46
domain_u	3	imap4	25	pop_2	47
auth	4	daytime	26	sunrpc	48
telnet	5	ctf	27	uucp_path	49
ftp	6	nntp	28	netbios_ns	50
eco_i	7	shell	29	netbios_ssn	51
ntp_u	8	IRC	30	netbios_dgm	52
ecr_i	9	nnsp	31	sql_net	53
other	10	http_443	32	vmnet	54
private	11	exec	33	bgp	55
pop_3	12	printer	34	Z39_50	56
ftp_data	13	efs	35	ldap	57
rje	14	courier	36	netstat	58
time	15	uucp	37	urh_i	59
mtp	16	klogin	38	X11	60
link	17	kshell	39	urp_i	61
remote_job	18	echo	40	pm_dump	62
gopher	19	discard	41	tftp_u	63
ssh	20	systat	42	tim_i	64
name	21	supdup	43	red_i	65

Çizelge 5.9'daProtokol bilgisinin sayısal kod tanımlamaları gösterilmektedir.

Çizelge 5.9. Protokol bilgisinin sayısal kod tanımlamaları(Güven, 2007).

Protokol	Sayısal Degeri
Tcp	0
Udp	1
Icmp	2

Çizelge 5.10'daBayrak bilgisinin sayısal kod tanımlamaları gösterilmektedir.

Çizelge 5.10. Bayrak bilgisinin sayısal kod tanımlamaları(Güven, 2007).

Bayrak(Flag)	Sayısal Degeri
S0	0
S1	1
S2	2
S3	3
SF	4
SH	5
OTH	6
REJ	7
RSTO	8
RSTOSO	9
RSTR	10

6. UYGULAMA

6.1 Kullanılan Saldırı Veri Kümeleri

Bu çalışmada kullanılan saldırı veri kümeleri için KDD'99 %10 luk veri bloğundan rastgele seçilen saldırı bilgilerinin tamamı(42 sütun) kullanılmıştır. Veri kümelerinde yapılacak testleri için saldırı olarak Normal(0)-Pod(14)- Imap(5) saldırı bilgilerinin bulunduğu ornek200. txt dosyası ile Normal(0)-Pod(14)-Imap(5)-Nmap(5) saldırılarının bulunduğu ornek222. txt dosyası oluşturulmuştur. YSA eğitimde kullanılacak veri kümesi olarak ise tüm saldırılarının bulunduğu atak3000. txt dosyası ile Nmap(5) dışında tüm saldırıların bulunduğu atak2781. txt dosyası oluşturulmuştur.

Çizelge 6.1'de bahsedilen saldırı dosyalarında bulunan saldırı tip ve örnek sayıları gösterilmektedir.

Çizelge 6.1. YSA eğitimde kullanılan saldırı veri kümeleri ve özellikleri.

		Saldırı Veri Kümeleleri			
		atak3000. txt	**atak2781. txt**	**ornek200. txt**	**ornek222. txt**
Saldırı	**Sayısal Degeri**	**Örnekleme**	**Örnekleme**	**Örnekleme**	**Örnekleme**
normal	0	604	604	130	130
back	1	212	212	0	0
buffer_overflow	2	23	23	0	0
ftp_write	3	7	7	0	0
guess_passwd	4	46	46	0	0
imap	5	10	10	10	10
ipsweep	6	926	926	0	0
land	7	19	19	0	0
loadmodule	8	8	8	0	0
multihop	9	6	6	0	0
neptune	10	638	638	0	0
nmap	11	219	0	0	22
perl	12	2	2	0	0
phf	13	3	3	0	0
pod	14	107	107	60	60
portsweep	15	35	35	0	0
rootkit	16	8	8	0	0
satan	17	29	29	0	0

Çizelge 6.1. YSA eğitimde kullanılan saldırı veri kümeleri ve özellikleri (devam ediyor).

smurf	18	58	58	0	0
spy	19	2	2	0	0
teardrop	20	14	14	0	0
warezclient	21	16	16	0	0
warezmaster	22	8	8	0	0
Toplam		**3000**	**2781**	**200**	**222**

Çalışmada kullanılan veri kümeleri Şekil 6.1'de gösterilmiştir. Çalışmada yöntem olarak Tanrıkulu(2009)'nun kullandığı metot kullanılmıştır. Çizelge 6.1'de belirtilen sayı ve tipte saldırı veri kümeleri oluşturup doğrudan eğitim yöntemiyle iki ayrı örnek yapılarak her örnek üç farklı deneme yapılarak tekrarlanmıştır.

Örneklerde bilinen saldırı dosyaları atak3000. txt Örnek-1 için YSA'ya öğretilerek test kümesi olan ornek200. txt içinde saldırıları tespit etmesi amaçlanmıştır. Örnek-2 için ise bilinmeyen saldırı tespiti için içinde bulunmasını istenilen Nmap(11) saldırı kümesi çıkartılarak oluşturulan atak2781. txt dosyası YSA'ya öğretilerek ornek222. txt içinde Nmap(11) saldırı kümesi bulunması amaçlanmaktadır.

Şekil 6.1. Deneylerde kullanılacak saldırı veri kümeleri.

Kullanılan Yöntem	**Bilinen**	**Bilinmeyen**
Doğrudan Eğitim	**Örnek-1**	**Örnek-2**
	atak3000. txt	atak2781. txt
	ornek200. txt	ornek222. txt
Ayrı Eğitim	**Örnek-3**	**Örnek-4**
	ornek200. txt	ornek222. txt
	normal0. txt	normal0. txt
	imap5. txt	imap5. txt
	pod14. txt	pod14. txt

.

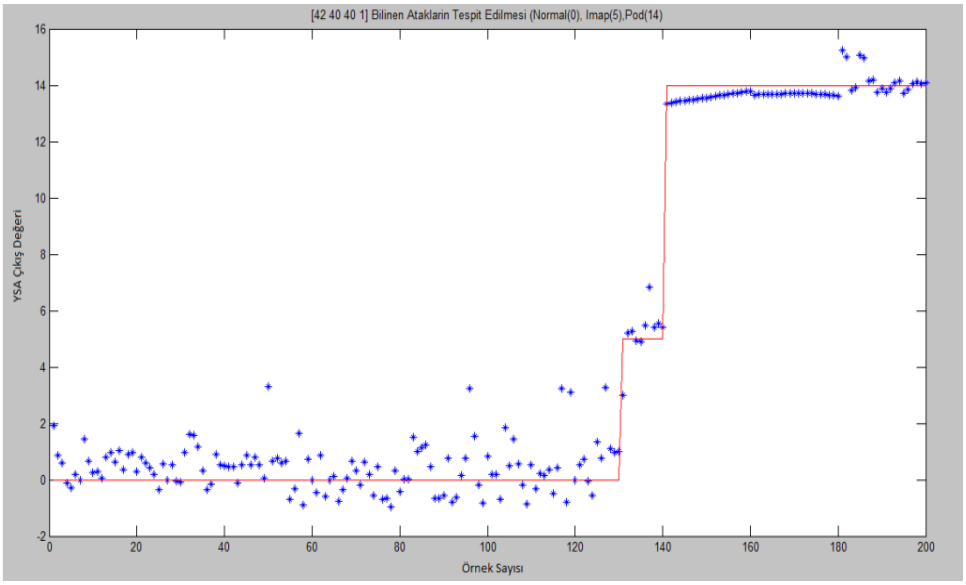
Ayrı eğitim yöntemiyle iki ayrı örnek yapılacak ve her örnek üçfarklı deneme olarak tekrar edilmiştir. Bilinen saldırı kümeleri için Örnek-3'te test verisinin içindeki saldırıların tespit edilmesi amacıyla seçilen normal0.txt, imal5.txt ve pod14.txt saldırı dosyaları teker teker eğitilmiştir. Örnek-4'te ise bilinmeyen saldırının tespiti için oluşturulan test kümesi ornek222.txt içindeki Nmap(11) saldırı kümesini normal0.txt, imal5.txt ve pod14.txt saldırı dosyaları teker teker eğitilerek yapılması amaçlanmıştır. Deneylerde kullanılan veri kümelerinin özeti EKLER kısmındadır. Büyük boyutlu saldırı veri kümelerinin tamamı CD ile birlikte verilmektedir.

6.2 Doğrudan Eğitim Yoluyla Saldırıların Tespit Edilmesi

Bu çalışmada bilinen saldırının tespit edilmesi ilk önce eğitim setini doğrudan eğiterek test verisindeki saldırı dosyalarını tespit etmesi beklenmektedir. Grafiklerde düz çizgi(-) eğitim verisini gösterirken yıldız işareti(*) eğitilmek istenen atak dosyalarını göstermektedir. Etiket kısmında YSA yapısı ve eğitilen atakların ismini göstermektedir. Yapılan denemeler aşağıda sırasıyla verilmektedir.

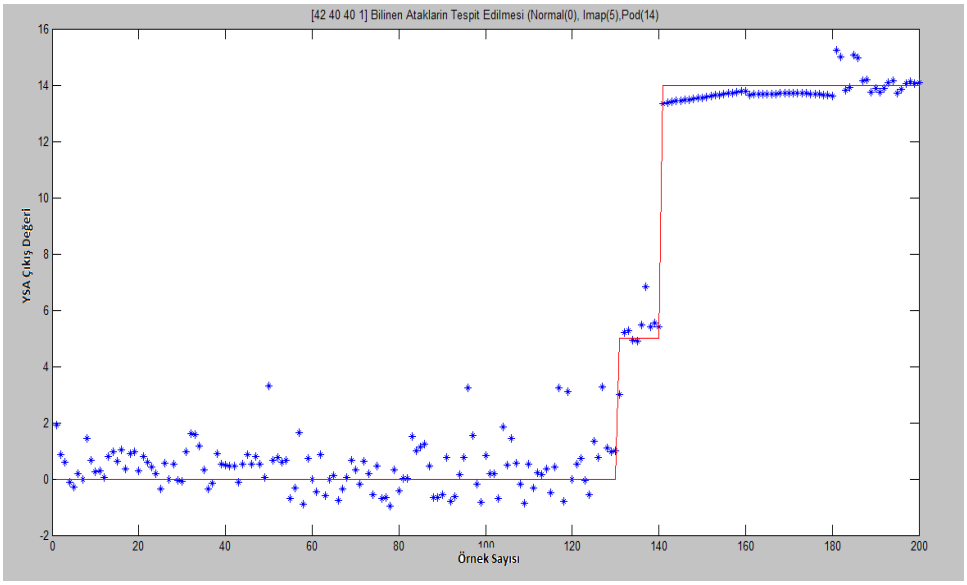
6.2.1 Bilinen saldırının tespit edilmesi

Şekil 6.2'de Pod(14) saldırısı kendi değerine yakınsadığı normal(0) ve imap(5) saldırılarının daha az yakınsadığını gözlenmiştir.



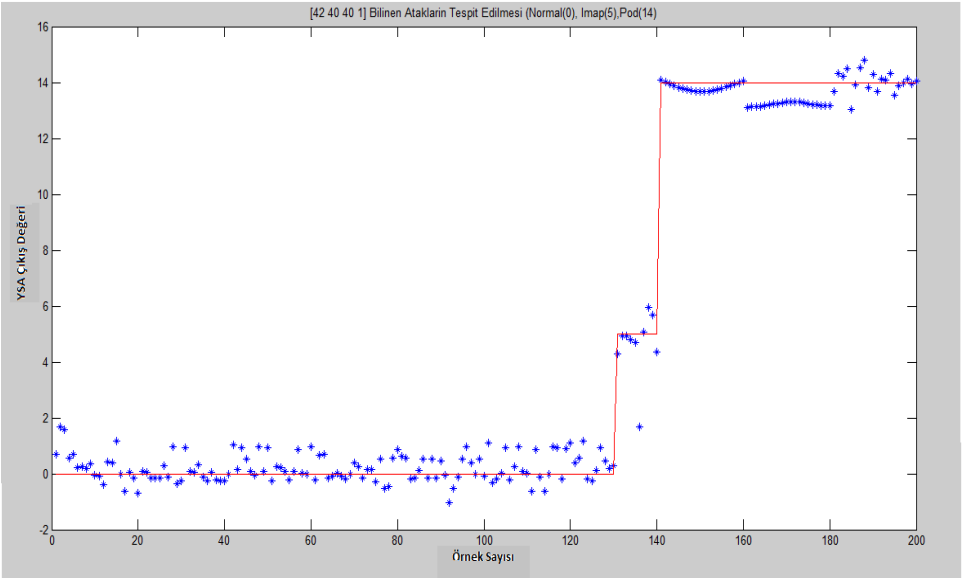
Şekil 6.2. Örnek-1 bilinen atakların tespiti deneme-1.

Şekil 6.3'de Pod(14) saldırısı ve imap(5) kendi değerlerine yakınsadığı normal(0) saldırılarının -2 ile 2 arasıdaki değerlere yakınsadığını gözlenmiştir.



Şekil 6.3. Örnek-1 bilinen atakların tespiti deneme-2.

Şekil 6.4'de Normal(0) saldırısı ve imap(5) kendi değerlerine yakınsadığı Podl(14) saldırılarının kendi değerine daha az yakınsadığını gözlenmiştir.

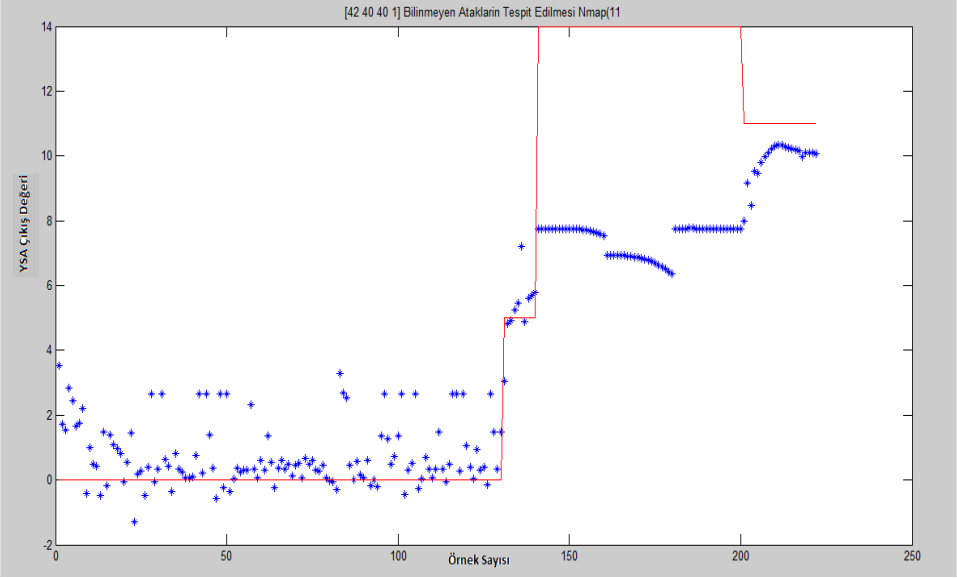


Şekil 6.4. Örnek-1 bilinen atakların tespiti deneme-3.

6.2.2 Bilinmeyen saldırıların tespit edilmesi

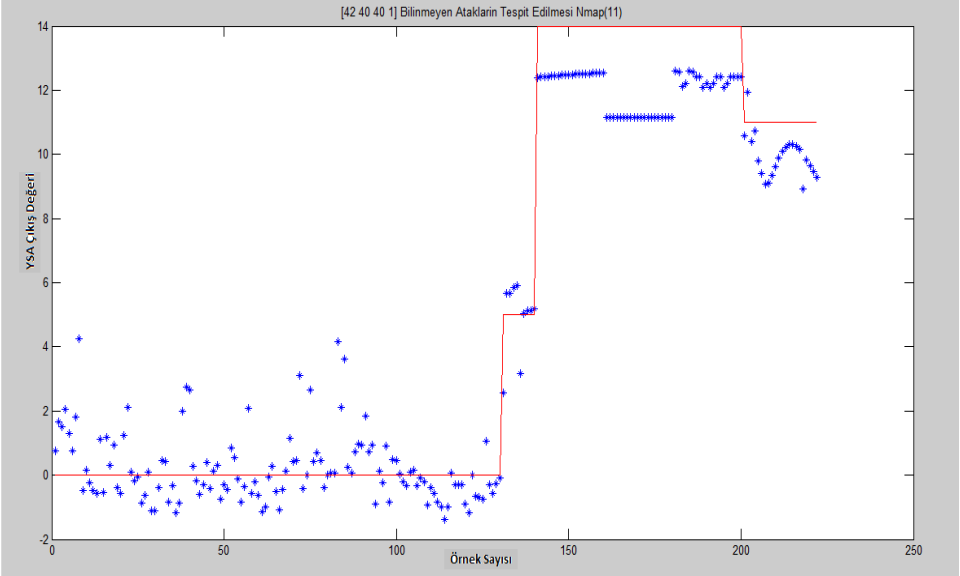
Bu çalışmada bilinmeyen saldırının tespit edilmesi ilk önce içinde bulunması istenen (saldırı verisi çıkartılan) eğitim setini doğrudan eğitmek yöntemiyle test verisindeki saldırı dosyalarını tespit etmesi beklenmektedir. Yapılan denemeler aşağıda sırasıyla verilmiştir.

Şekil 6.5'de Normal(0) saldırısı 0 ile 3 arasında değerler alsa da kendi değerine çoğunlukla yakınsamıştır. Imap(5) kendi değerlerine yakınsadığı Podl(14) saldırılarının kendi değerine yakınsamadığını, bilinmeyen saldırının 6-8 ve 10 değerlerine yakınsadığını gözlenmiştir.



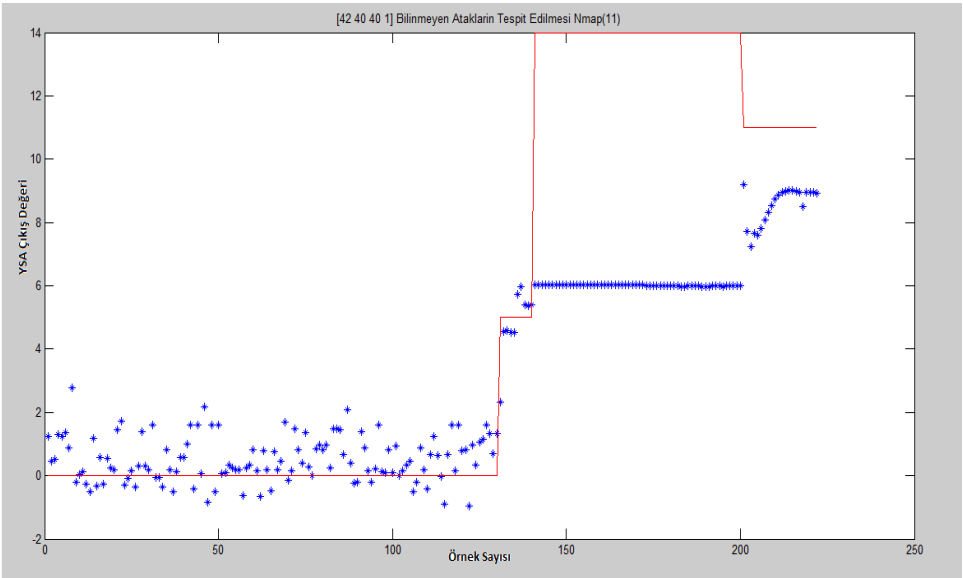
Şekil 6.5. Örnek-2 bilinmeyen atakların tespiti-deneme-1.

Şekil 6.6'da Normal(0) saldırısı 0 ile 3 arasında değerler alsa da kendi değerine çoğunlukla yakınsamıştır. Imap(5) kendi değerlerine yakınsadığı Podl(14) saldırılarının kendi değerine yakınsamadığını fakat 2 ayrı blok halinde 11 ve13 değerlerine yakınsadığı, bilinmeyen saldırının 9 ile 11 değerleri arasında değerler aldığını gözlenmiştir.



Şekil 6.6. Örnek-2 bilinmeyen atakların tespiti-deneme-2.

Şekil 6.7'de Normal(0) saldırısı 0 ile 2 arasında değerler alsa da kendi değerine çoğunlukla yakınsamıştır. Imap(5) 4ile 6 arasında değerler almıştır. Podl(14) saldırılarının kendi değerine yakınsamadığını blok halinde 6 değerlerine yakınsadığı, bilinmeyen saldırının 7 ile 9 değerleri arasında değerler aldığını göstermiştir.

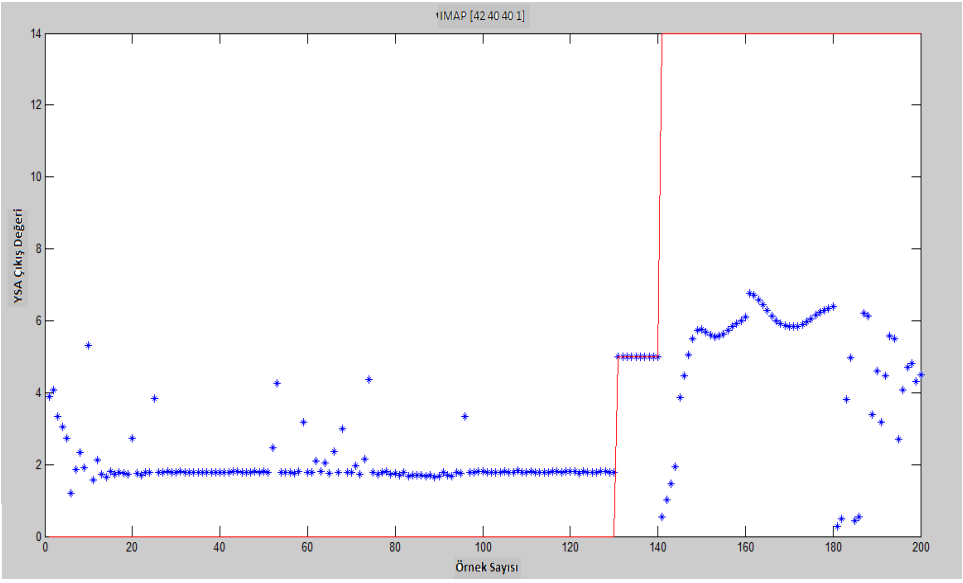


Şekil 6.7. Örnek-2 bilinmeyen atakların tespiti-deneme-3.

6.3 Ayrı EğitimYoluyla Saldırıların Tespit Edilmesi

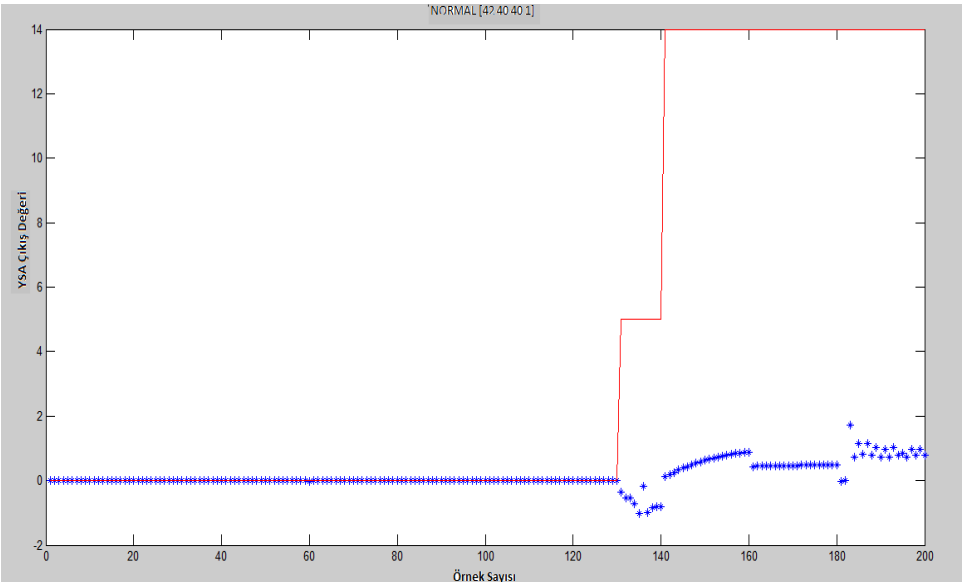
Çalışmada saldırı veri kümeleri ayrı ayrı eğitilerek test verisindeki saldırı kümelerini yakalaması beklenmektedir. Elde edilen grafikler sırasıyla aşağıda gösterilmektedir.

Şekil 6.8'de ayrı eğitim saldırı kümesinden normal(0) 2 değerine yakınsamış, Pod(14) 6 ile 8 değerleri arasında değişik değerler almış ve saldırıyı tanımamıştır fakat imap(5) kendi değerini yakalamış ve saldırıyı tespit etmiştir.



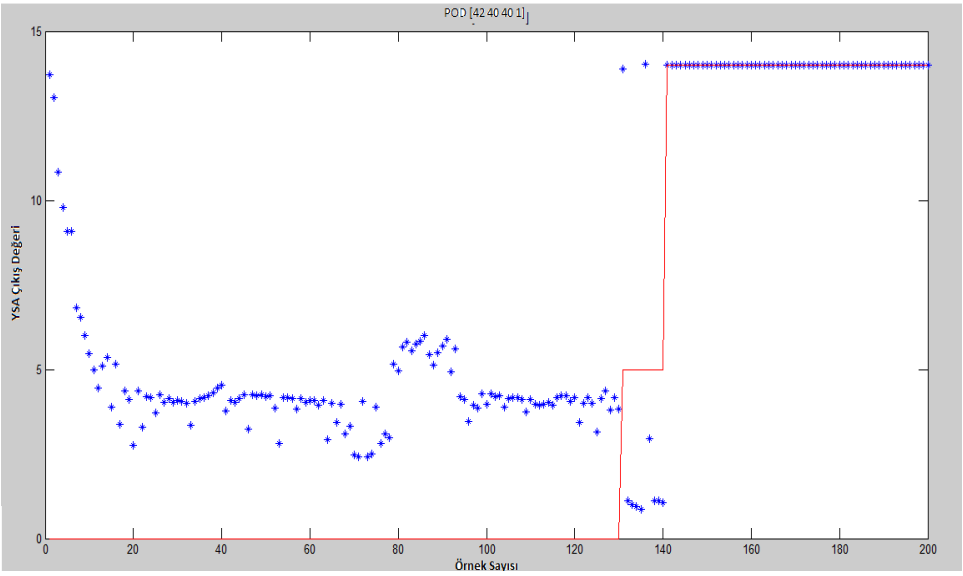
Şekil 6.8. Örnek-3 bilinen atakların tespiti-deneme-1-IMAP.

Şekil 6.9'da ayrı eğitim saldırı kümesinden normal(0) kendi değerini yakalamış ve saldırıyı tespit etmiştir. Pod(14) ve imap(5) saldırıyı tanımamıştır ve sıfıra yakınsamışlardır.



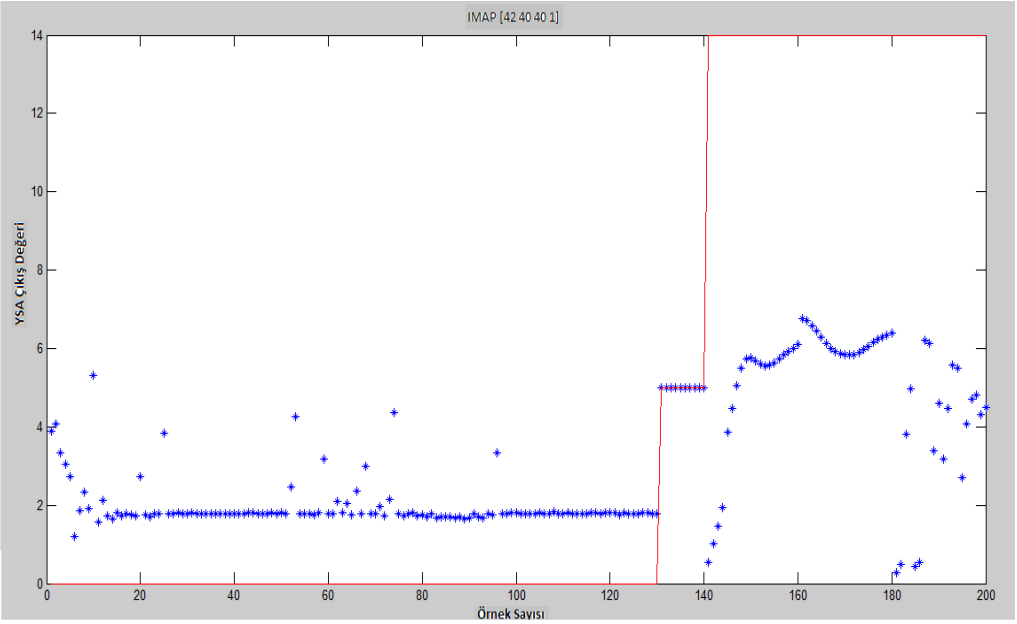
Şekil 6.9. Örnek-3 bilinen atakların tespiti-deneme-1-NORMAL.

Şekil 6.10'da ayrı eğitim saldırı kümesinden Pod(14) kendi değerini yakalamış ve saldırıyı tespit etmiştir. Normal(0) ve imap(5) saldırıyı tanımamıştır.



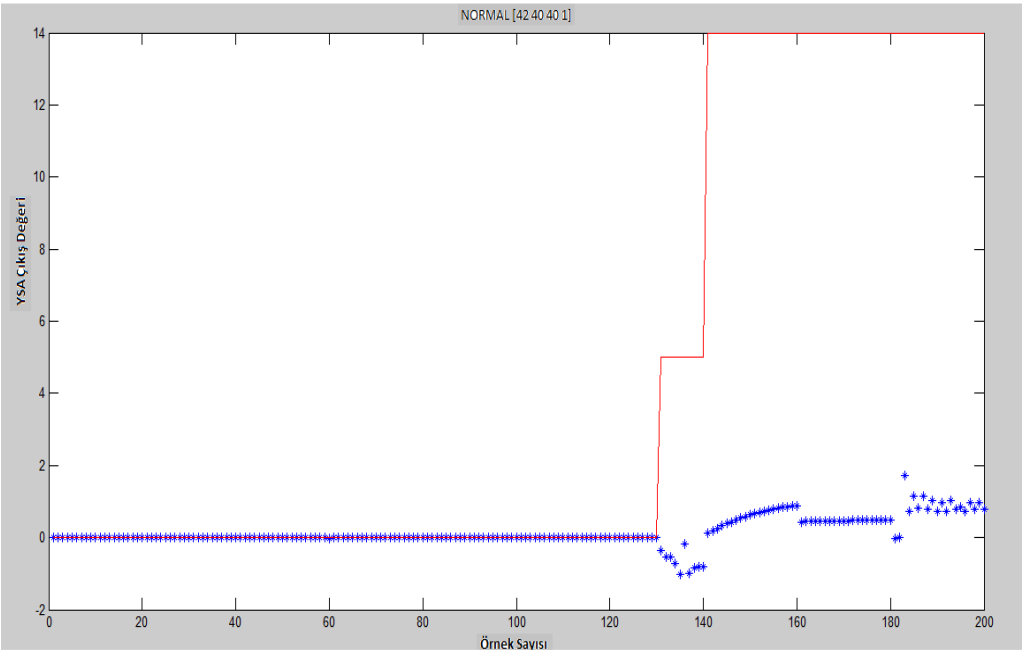
Şekil 6.10. Örnek-3 bilinen atakların tespiti-deneme-1-POD.

Şekil 6.11'de Imap(5) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir.



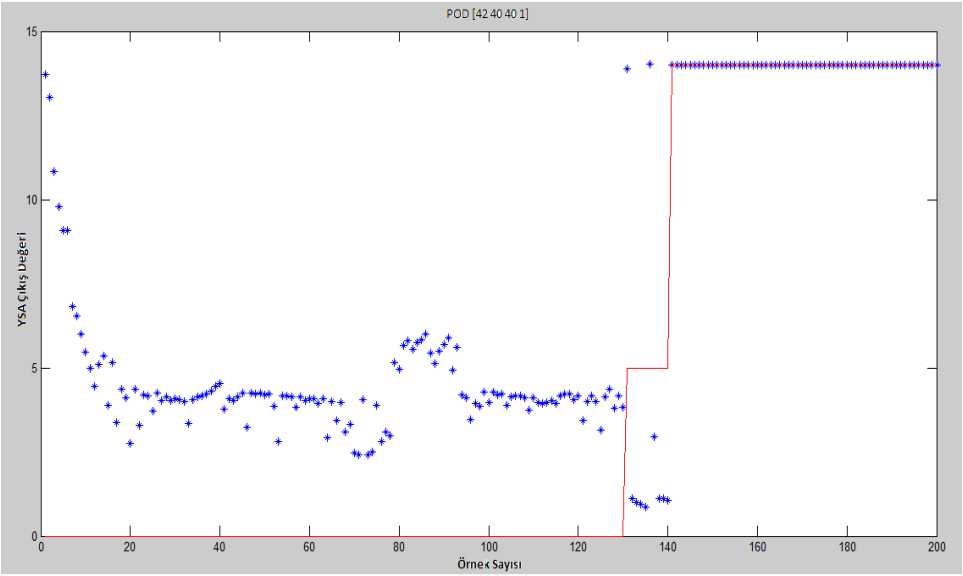
Şekil 6.11. Örnek-3 bilinen atakların tespiti-deneme-2-IMAP.

Şekil 6.12'de Normal(0) saldırı veri kümesi kendi değerini yakalamıştır.



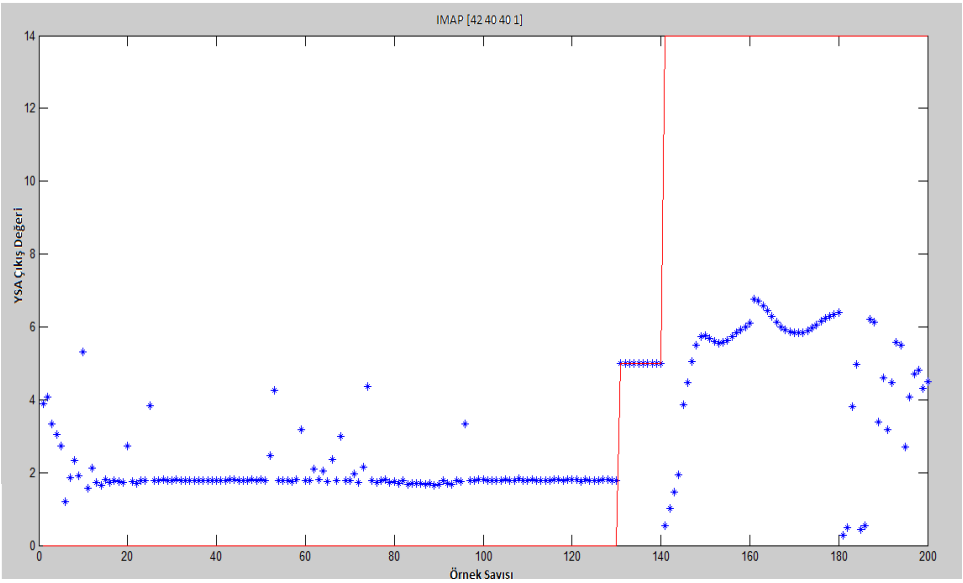
Şekil 6.12. Örnek-3 bilinen atakların tespiti-deneme-2-NORMAL.

Şekil 6.13'de Pod (14) saldırı veri kümesi ve saldırıyı tespit etmiştir.



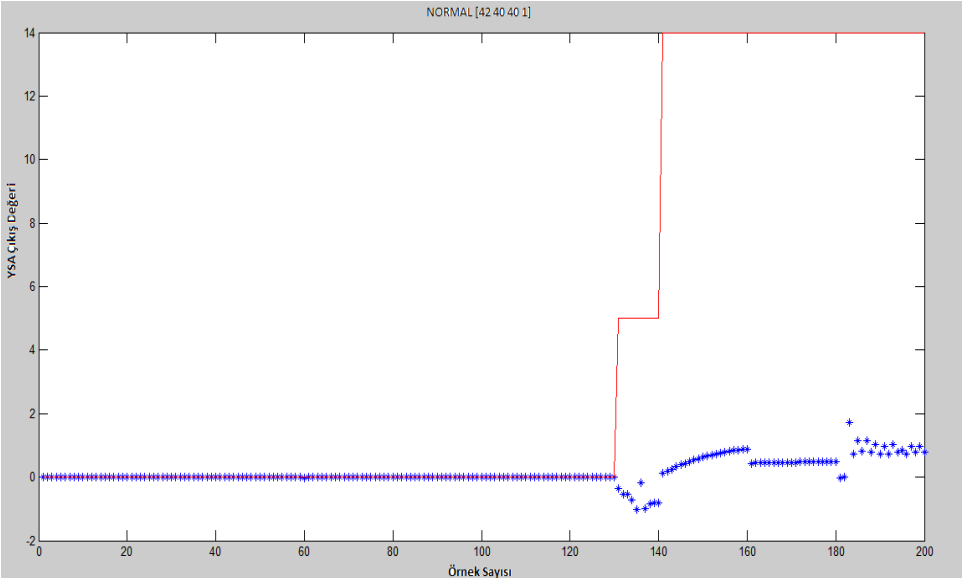
Şekil 6.13. Örnek-3 bilinen atakların tespiti-deneme-2-POD.

Şekil 6.14'de Imap(5) saldırı veri kümesi saldırıyı tespit etmiştir.



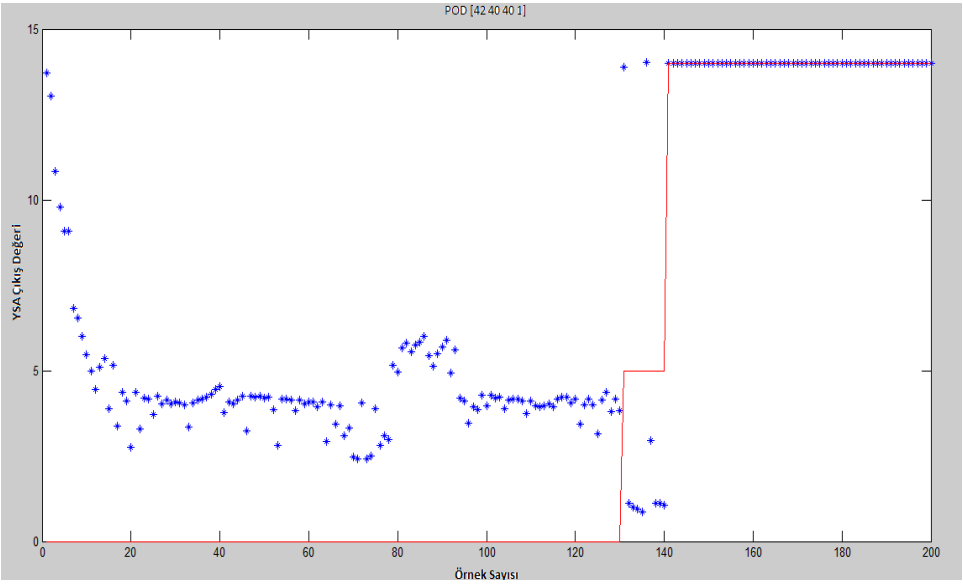
Şekil 6.14. Örnek-3 bilinen atakların tespiti-deneme-3-IMAP.

Şekil 6.15'de Normal(0) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir.



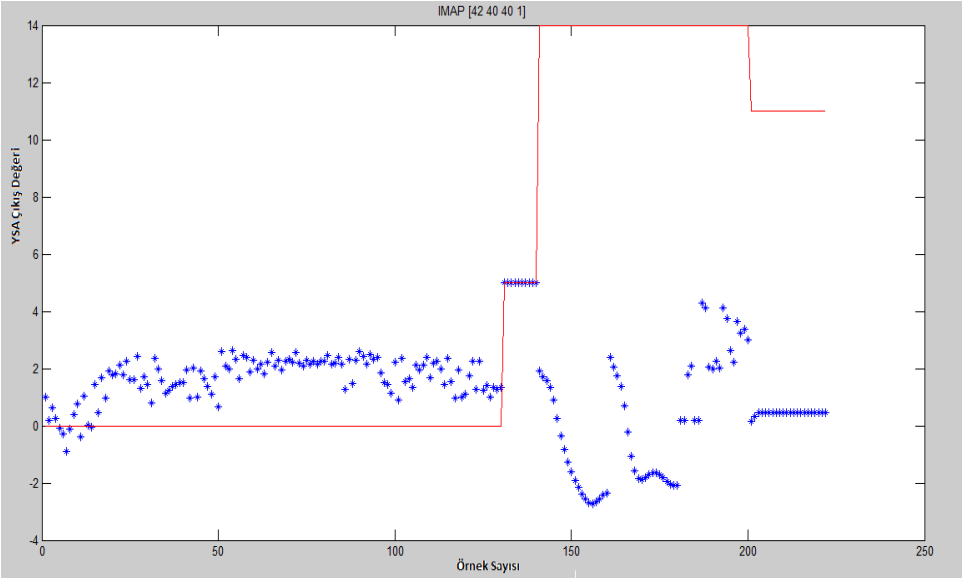
Şekil 6.15. Örnek-3 bilinen atakların tespiti-deneme-3-NORMAL.

Şekil 6.16'de Pod(14) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir.



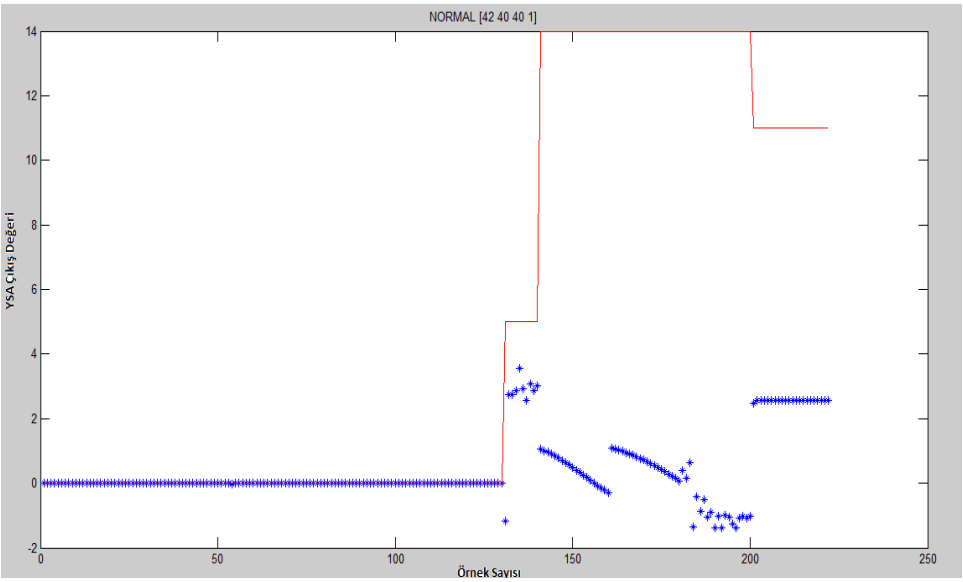
Şekil 6.16. Örnek-3 bilinen atakların tespiti-deneme-3-POD.

Şekil 6.17'de Imap(5) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırıblok halinde 0 değerine yakınsamıştır.



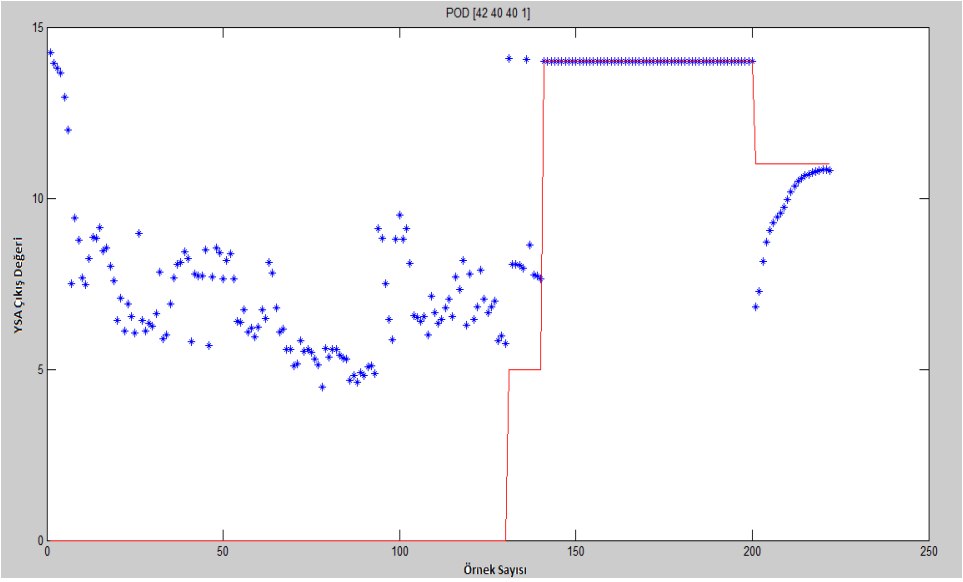
Şekil 6.17. Örnek-4 bilinmeyen atakların tespiti-deneme-1-IMAP.

Şekil 6.18'de Normal(0) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı blok halinde 2 değerine yakınsamıştır.



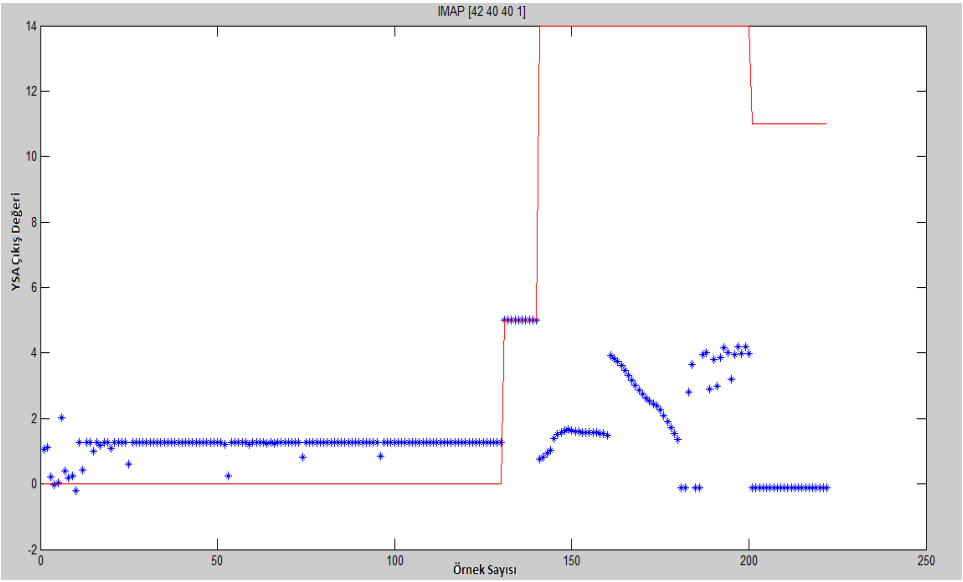
Şekil 6.18. Örnek-4 bilinmeyen atakların tespiti-deneme-1-NORMAL.

Şekil 6.19'daPod(14) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı 8-10 arasındaki değerlere yakınsamıştır.



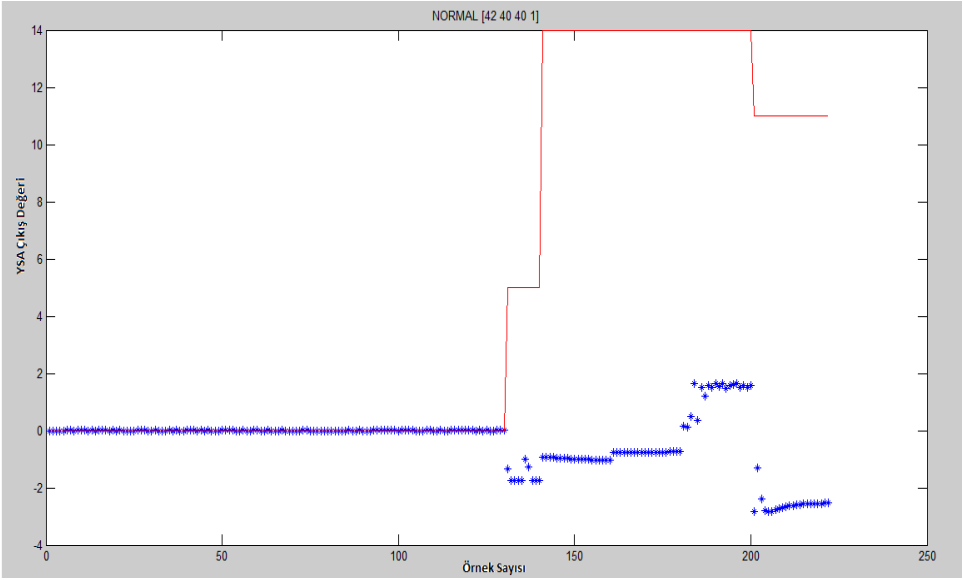
Şekil 6.19. Örnek-4 bilinmeyen atakların tespiti-deneme-1-POD.

Şekil 6.20'deImap(5) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı 0 değerine yakınsamıştır.



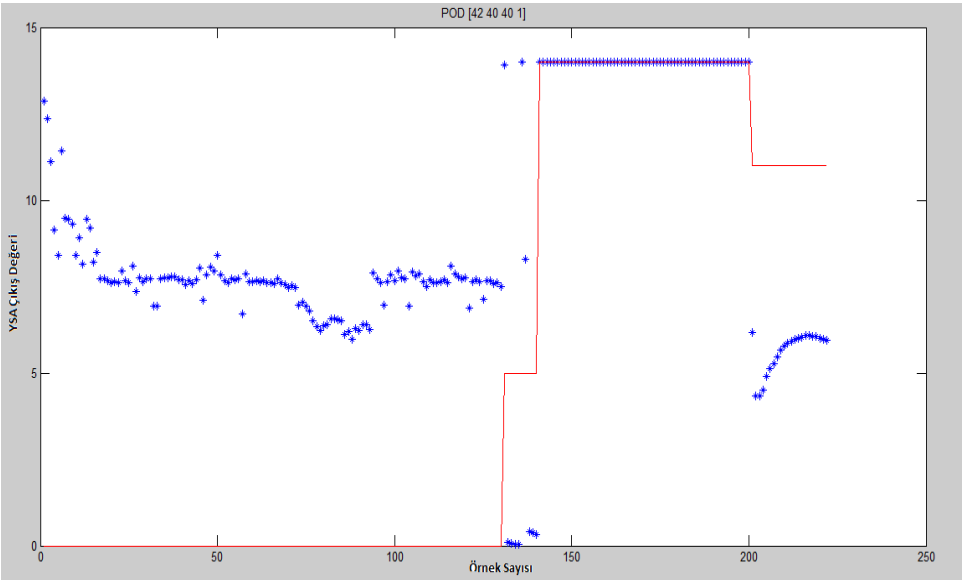
Şekil 6.20. Örnek-4 bilinmeyen atakların tespiti-deneme-2-IMAP.

Şekil 6.21'deNormal(0) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı -3 değerine yakınsamıştır.



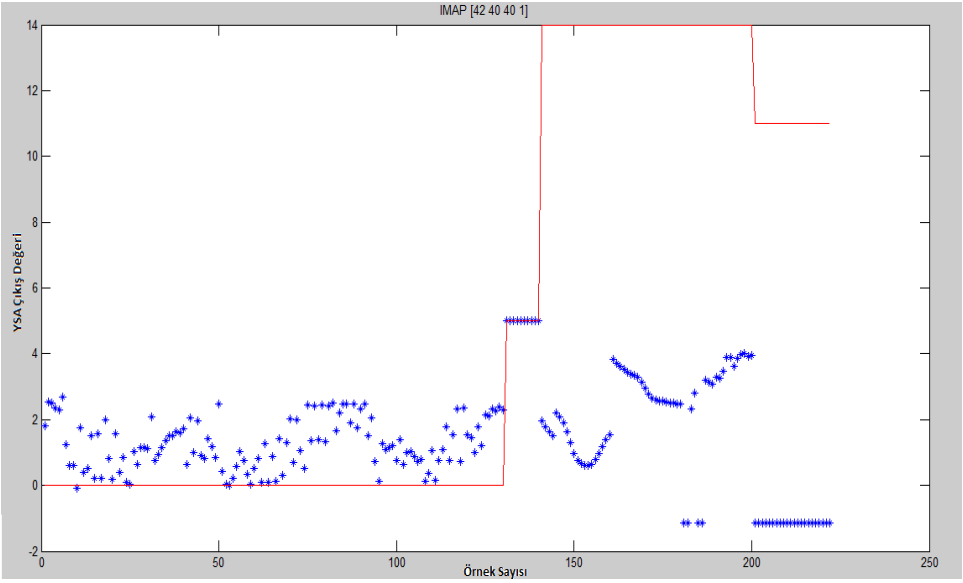
Şekil 6.21. Örnek-4 bilinmeyen atakların tespiti-deneme-2-NORMAL.

Şekil 6.22'dePod(14) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı 4-6 arasındaki değerlere yakınsamıştır.



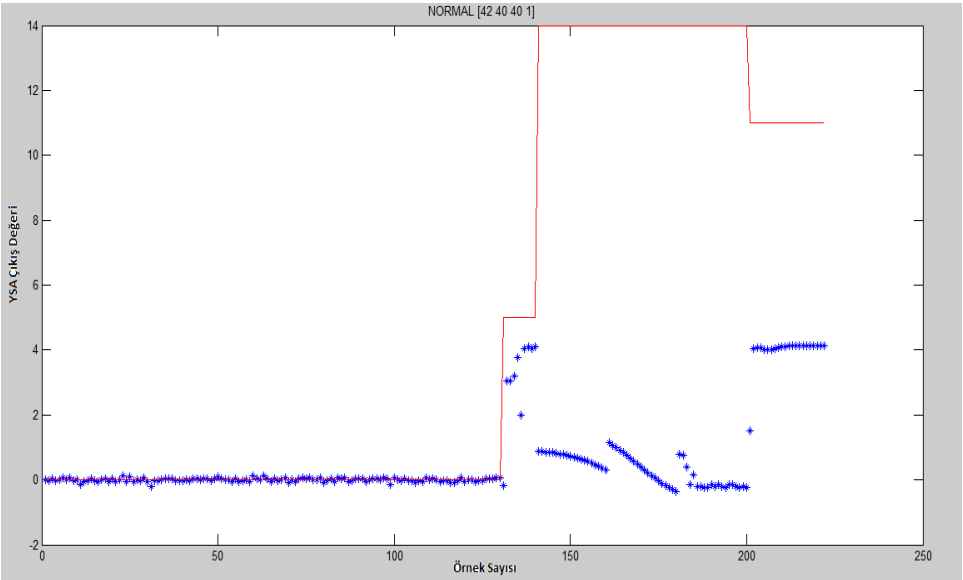
Şekil 6.22. Örnek-4 bilinmeyen atakların tespiti-deneme-2-POD.

Şekil 6.23'deImap(5) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı -1 değerine yakınsamıştır.



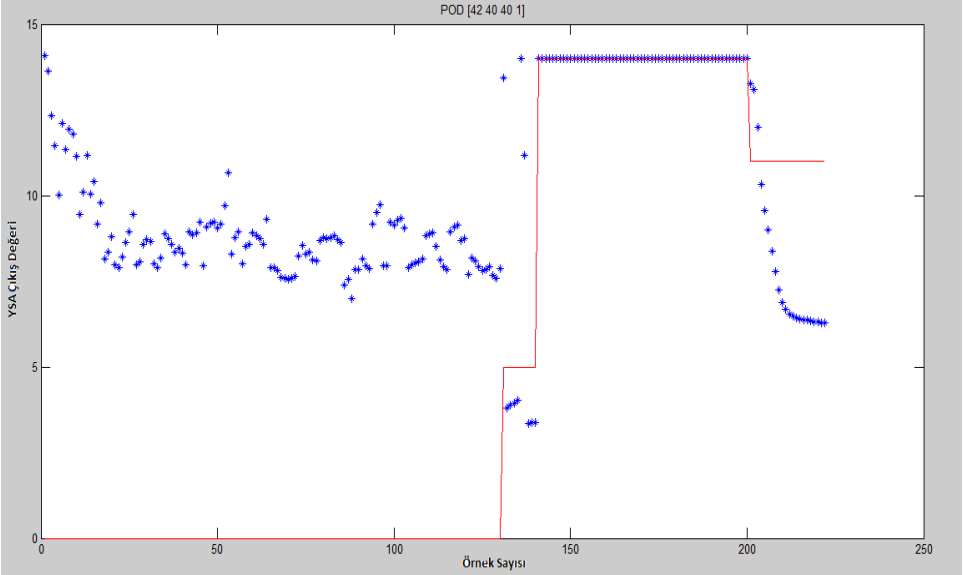
Şekil 6.23. Örnek-4 bilinmeyen atakların tespiti-deneme-3-IMAP.

Şekil 6.24'de Normal(0) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı 4 değerine yakınsamıştır.



Şekil 6.24. Örnek-4 bilinmeyen atakların tespiti-deneme-3-NORMAL.

Şekil 6.25'de Pod(14) saldırı veri kümesi kendi değerini yakalamış ve saldırıyı tespit etmiştir. Bilinmeyen saldırı 6 değerinden başlayarak Pod'un kendi değerine yakınsamıştır.



Şekil 6.25. Örnek-4 bilinmeyen atakların tespiti-deneme-3-POD.

Bilinmeyen atakların tespitinin ayrı ayrı eğitim yapılarak öğretilmesinde (Örnek-4) her atak dosyası kendi saldırısını tanımış ve tespit etmiştir. Dışarıdan ağa hiç öğretilmeyen atak için ise yapılan denemelerde yabancı saldırının tespit edildiği fakat farklı değerlere yakınsadığı gözlenmiştir.

6.4 Saldırı Başarımlarının Tespit Edilmesi

Çalışmada saldırı tespitinin test edilmesinde başarım oranlarının tespitinde aşağıdaki denklem kullanılmıştır. Bu denklem Tanrıkulu'nun(2009) çalışmasında da kullanılmıştır. Denklemde yakınacak değer saldırı tespitinde sayısallaştırdığımız saldırı tespit sıra numalarıdır. Ortalama ise YSA 'nın yapılan testler için denemelerin çıktı değerlerinin aritmetik ortalamasıdır. Örneğin, Imap saldırısı için yd=5 olması beklenir. Çıktı değerlerinin ortalaması ise ort=4.5 çıkmış olabilir. Eşitlikte değerler yerine yazılarak elde edilen sonuçlar başarım oranlarını vermektedir.

$$SBB = \frac{yd - |yd - ort| * 100}{yd} \qquad \text{(E.6.1)}$$

Eşitlik "E.6.1" de;

SBB= Saldırı bulma başarımı (%),

yd=Yakınsanması istenen değer,

ort=YSA çıktı değerlerinin aritmetik ortalamasını göstermektedir.

6.4.1 Normal Saldırı Verisinin Başarımını Tespit Edilmesi

Denklemde yakınsanacak değer saldırı tespitinde sayısallaştırdığımız saldırı tespit sıra numaralarıdır. Ortalama ise YSA 'nın yapılan testler için denemelerin çıktı değerlerinin aritmetik ortalamasıdır. Diğer saldırılardan farklı olarak Normal saldırı değeri 0(sıfır) sayısal değeri verildiği için YSA çıkışları -1 ve 1 arasındaki sıfır değerine yakınsayacaktır. Diğer saldırı bulma başarım denklemi sıfır değerine bölmek anlamsız olacağından "E.6.2" kullanılmıştır. Denklemde değerler yerine yazılarak elde edilen sonuçlar başarım oranlarını vermektedir.

$$NSBB = |1 - ort|| * 100 \quad \text{(E. 6.2)}$$

Denklemde;

NSBB= Normal saldırıyı bulma başarımı (%),

yd=Yakınsanması istenen değer,

ort=YSA çıktı değerlerinin aritmetik ortalamasını göstermektedir.

Hesaplamalar için kullanılan MATLAB kodları EKLER kısmındadır.

7. SONUÇLAR VE ÖNERİLER

7.1 Yapılan Deneylerin Başarım Oranları ve Süreleri

Yapılan çalışma kapsamında veriler iki farklı yöntem dört ayrı örnek ve her birindeki üç farklı deneme için elde edilen veriler çizelgelerde gösterildiği gibidir.

Yapılan Örnek-1 deneyinde bilinen saldırı kümeleri için doğrudan eğitim yoluyla üç ayrı deneme yapılmıştır. Deney süreleri, iterasyon (yenileme)miktarı ve başarım yüzdeleri Çizelge 7.1 de gösterilmiştir. Örnek-1 deneyinde 3 ayrı saldırı sonucun ortalamaları alındığında sırasıyla Normal(0) için %72.92, Imap(5) için %94.81, Pod(14) için %88.09 olarak bulunmuştur. Örnek-1 deneyinde eğitilen saldırı veri dosyasının(atak3000.txt) test verisinde(ornek200) bulunan saldırıları yakaladığı anlaşılmıştır.

Çizelge 7.1. Örnek-1 süre ve başarım bilgileri.

Deney: Ornek-1(Bilinen)				
Yöntem:Dogrudan Egitim				
YSA Yapısı: [42 40 40 1]				
Deneme-1				
Süre(sn)	Iterasyon	Basarım(%)		
		Normal(0)	Imap(5)	Pod(14)
492	19	56. 4138	96. 0634	98. 6488
Deneme-2				
Süre(sn)	Iterasyon	Basarım(%)		
		Normal(0)	Imap(5)	Pod(14)
1060	50	83. 7797	95. 7308	67. 8055
Deneme-3				
Süre(sn)	Iterasyon	Basarım(%)		
		Normal(0)	Imap(5)	Pod(14)
495	19	78. 5713	92. 6579	97. 8380

Yapılan Örnek-2 deneyinde bilinmeyen saldırı kümeleri için doğrudan eğitim yoluyla üç ayrı deneme yapılmıştır. Deneyde amaç olarak eğitim kümesinde olmayan bir saldırı olan Nmap(11) saldırısını YSA'nın öğrenmesi ve yakalaması hedeflenmiştir. Deney süreleri, iterasyonmiktarı ve başarım yüzdeleri Çizelge 7.2'de gösterilmiştir. Örnek-2 deneyinde 3 ayrı saldırı sonucun ortalamaları sırasıyla Normal(0) için %80. 03, Imap(5) için %85. 69, Pod(14) için %88. 65 ve Nmap(11) için ise %26. 32 olarak tespit ettiği görülmüştür. Örnek-2 deneyinde eğitilen saldırı veri dosyasının(atak2781.txt) sonucu test verinde(ornek222. txt) bulunan saldırılardan Normal(0), Imap(5) ve Pod(14) saldırılarını yüksek yüzdelik oranlarında yakaladığı fakat bilinmeyen saldırı olarak Nmap(11) saldırısını düşük yüzdelik oranıyla yakaladığı gözlenmiştir.

Çizelge 7.2. Örnek-2 süre ve başarım bilgileri.

Deney: Ornek-2(Bilinmeyen)					
Yöntem:Dogrudan Egitim					
YSA Yapısı: [42 40 40 1]					
Deneme-1					
Süre(sn)	Iterasyon	Basarım(%)			
		Normal(0)	Imap(5)	Pod(14)	Nmap(11)
489	21	93, 707	96, 4423	99, 9237	28, 0639
Deneme-2					
Süre(sn)	Iterasyon	Basarım(%)			
		Normal(0)	Imap(5)	Pod(14)	Nmap(11)
1104	50	72, 8587	99, 2002	96, 0477	27, 7629
Deneme-3					
Süre(sn)	Iterasyon	Basarım(%)			
		Normal(0)	Imap(5)	Pod(14)	Nmap(11)
422	18	73, 5434	61, 4515	69, 9791	23, 151

Yapılan Örnek-3 deneyinde bilinen saldırı kümeleri için ayrı eğitim yoluyla üç ayrı deneme yapılmıştır. Deneyde amaç olarak eğitim kümesinde yakalanması istenilen saldırı veri kümeleri teker teker ve ayrı olarak eğitilerek net1, net2, net3 YSA ağlarına öğretmek ve bulunmasını sağlamaktır. Deney süreleri, iterasyonmiktarı ve başarım yüzdeleri Çizelge 7.3'de gösterilmiştir. Örnek-3 deneyinde üç ayrı saldırı veri kümesi için farklı sürelerve iterasyonmiktarları çıkmasına rağmen aynı sonuçlar elde edilmiştir. Sonuçların ortalamaları sırasıyla Normal(0) için %99. 98, Imap(5) için %99. 99, Pod(14) için %99.99 oranlarında tespit edildiği görülmüştür. Örnek-3 deneyinde eğitilen saldırı veri dosyaları normal0.txt, imap5.txt, pod14.txt sonucu test verisinde(ornek200.txt) bulunan saldırılardan Normal(0), Imap(5) ve Pod(14) saldırılarını yüksek yüzdelik oranlarında yakaladığı ve tanıdığı gözlenmiştir.

Çizelge 7.3. Örnek-3 süre ve başarım bilgileri.

Deney: Ornek-3 (Bilinen)				
Yöntem:Ayrı Egitim				
YSA Yapısı: [42 40 40 1]				
Deneme-1				
Süre(sn)	Iterasyon	Basarım(%)		
		Normal(0)	Imap(5)	Pod(14)
1125	50	99, 9835	99, 9954	99, 9956
Deneme-2				
Süre(sn)	Iterasyon	Basarım(%)		
		Normal(0)	Imap(5)	Pod(14)
1062	50	99, 9835	99, 9954	99, 9956
Deneme-3				
Süre(sn)	Iterasyon	Basarım(%)		
		Normal(0)	Imap(5)	Pod(14)
973	45	99, 9835	99, 9954	99, 9956

Yapılan Örnek-4 deneyinde bilinmeyen saldırı kümeleri için ayrı eğitim yoluyla üç ayrı deneme yapılmıştır. Deneyde amaç olarak eğitim kümesinde yakalanması istenilen saldırı veri kümeleri teker teker ve ayrı olarak eğitilerek net1, net2, net3 YSA ağlarına öğretmek ve saldırıların bulunmasını sağlamaktır. Deney süreleri, iterasyon miktarı ve başarım yüzdeleri Çizelge 7.4'de gösterilmiştir. Örnek-4 deneyinde üç ayrı saldırı veri kümesi için elde edilen sonuçların ortalamaları sırasıyla Normal(0) için %99. 99, Imap(5) için %99.94, Pod(14) için %99. 99 ve Nmap(11) için ise %0 oranlarında tespit edildiği görülmüştür. Örnek-4 deneyinde eğitilen saldırı veri dosyaları normal0. txt, imap5.txt, pod14.txt sonucu test verisinde (ornek222.txt) bulunan saldırılardan Normal(0), Imap(5) ve Pod(14) saldırılarını yüksek yüzdelik oranlarında yakaladığı ve tanıdığı gözlenmiştir. Bilinmeyen ve bulunması istenilen Nmap(11) saldırısını ise yakalayamadığı gözlenmiştir.

Çizelge 7.4. Örnek-4 süre ve başarım bilgileri.

Deney: Ornek-4 (Bilinmeyen)					
Yöntem:Ayrı Egitim					
YSA Yapısı: [42 40 40 1]					
Deneme-1					
Süre(sn)	Iterasyon	Basarım(%)			
		Normal(0)	Imap(5)	Pod(14)	Nmap(11)
540	10	99, 9999	99, 9962	99, 9929	0
Deneme-2					
Süre(sn)	Iterasyon	Basarım(%)			
		Normal(0)	Imap(5)	Pod(14)	Nmap(11)
780	8	99, 9597	99, 9859	99, 9926	0
Deneme-3					
Süre(sn)	Iterasyon	Basarım(%)			
		Normal(0)	Imap(5)	Pod(14)	Nmap(11)
780	8	99, 8733	99, 9908	99, 9965	0

Yapılan çalışmada iki farklı yöntem ile dört farklı örnek,oniki deneme sonucu oluşturulan veri başarımları Çizelge7.5'de gösterilmiştir. Bilinen saldırının bulunmasındaki başarım oranı ortalaması%92.64 olarak saldırı tespitinin başarılı olduğunu gösterirken, bilinmeyen saldırının tespiti noktasında elde edilen oran %72.58 başarılı olamadığını göstermektedir.

Çizelge 7.5. Çalışmadaki ortalama başarım oranları.

YSA Yapısı	İşlem Eleman Sayısı	Veri Seti	Özellik Sayısı	Bilinen Saldırıların Bulunması Ortalama Başarım Oranı(%)	Bilinmeyen Saldırıların Bulunması Ortalama Başarım Oranı(%)
ÇKA	[42 40 40 1]	KDD'99	42	92, 6352	72, 5799

Yapılan çalışma, YSA'nın bir eğitim kümesi üzerinden sinir ağının tamamını öğrenmesinde başarılı olduğundan, Normal(0) ile tüm saldırıları bilgilerinin tespitinde başarılı olmuştur. Sinir ağının tamamını öğrenen YSA yapısında ağırlık önce rastgele seçilsedekarşılaştırmalı olarak ağırlıklar değiştirilmekte ve saldırı tespitinde ayırt edici olmaktadır. YSA aslında kendine öğretilen bir veriyi bulmaya çalışmaktadır. Bunu yaparken ise, farklı ağırlık değerleri, farklı süreler ve farklı iterasyonlar denemektedir. Örneğin, Imap(5) saldırısını öğrenen bir YSA, test veri kümesi içinde Pod(14) ve Normal(0) saldırıları bir grup olarak algılarken, Imap(5) saldırısını ayrı bir grup olarak tanımlamaktadır. Dolayısı ile sadece öğrendiği saldırıyı diğer gruplardan ayırabilme yeteneğine sahip olan bir YSA yapısı, bilinmeyen bir saldırı kümesi ile karşılaştığında, bu saldırıyı sadece kendisinden farklı olan gruba ait olarak tanımlayacaktır.

Sonuçta, YSA eğitimini aldığı saldırı kümesi dışında kalan tüm saldırıları ve normal trafiği, tek bir değere yakınsama durumu ilegöstermiştir. Bu nedenle, gerçekleştirilen uygulamalar ve denemelerde tek tek saldırıları öğrenen ağlar başarısız olduğundan, tüm saldırılar aynı değerlere yakınsama durumu göstermiştir. Tüm ağı öğrenen YSA'larda ise başarım oranlarının yüksek olması olumlu olarak

gözlemlenmiştir. Ağın büyüklüğüne göre iterasyonlar öğrenme sürelerinin fazla olması ise olumsuz olarak deneylerimizde gözlenmiştir.

Çalışmada kullanılan saldırı atak dosyalarının özeti ve kullanılan MATLAB kodları EKLER kısmında ayrı ayrı bulunmaktadır. Saldırı dosyalarının tamamı ve ek bilgiler CD ile beraber sunulmuştur.

Çizelge 7.6'da verilen tabloda gerek daha önce, gerekse bu uygulamada YSA nin ÇKA üzerinde eğitilerek yapılmış çalışmaların özeti gösterilmiştir. Çalışmalarda, Cannady(1998) ve Ryan(1998) kendi veri setlerini kullandığı farklı ÇKA yapısı kullandığı ve özellik sayısında farklılık olduğu gözlenmiştir. Mukkamala vd. (2002) ve Güven (2007) KDD'99 veri kümesinin tüm veri değerleri için, gerçekleştirdikleri farklı ÇKA yapısı denemelerinde buldukları başarımlar gösterilmiştir. Güven(2007) örnekleme sayısını sabit tutarak KDD'99 daki farklı saldırıları sınıflandırmaya çalışmıştır. Mukkamala vd.(2002) çalışmalarında elde ettiği % 99.25'lik başarım oranın bulunmasında ÇKA/DVM'nin birlikte çalıştırılmasından kaynaklandığı görülmektedir. Sammany(2007) DARPA'99 veri kümesinde ÇKA yapısında farklı katman ve daha az özellik kullarak başarım gerçekleştirmiştir. Tanrıkulu (2009) DARPA'98 veri kümesini kullanarak, internet ortamından elde ettikleri saldırı dosyalarını, IP adreslerini dört ayrı özellik yardımıylaiki farklı yöntem ve deneme kullanarak, başarımlar elde etmiştir.

Çizelge 7.6. Yapılan çalışmaların başarım oranları.

Çalışmayı Yapan	YSA Yapısı	İşlem Eleman Sayısı	Veri Seti	Özellik Sayısı	Başarım Oranları(%)
Cannady (1998)	ÇKA	[9 * * 1]	Cannady'in Oluşturduğu Veri Seti	10	91
Ryan (1998)	ÇKA	[100 30 10]	Kendi veri seti	100	96
Mukkamala (2002)	ÇKA, DVM	[41 40 40 1]	KDD'99	41	99, 25
Sammany (2007)	ÇKA	[50 30 3]	DARPA '99	35	93, 43
Güven (2007)	ÇKA	[41 6 8 1]	KDD'99	41	92, 5
Tanrıkulu(2009)	ÇKA	[12 40 40 1]	DARPA '98	5	99, 15
Marttin(2014)	ÇKA	[42 40 40 1]	KDD'99	42	83,11

Burada yapılan çalışmada ise, Mukkamala vd.(2002), Güven(2007) ve Öksüz(2007) gibi KDD'99'a ait rastgele seçilmiş %10'luk veri kümesi kullanılmıştır. Rastgele seçim işlemi, nispeten eğitim başarımı daha zor bir kümeye ulaşım talebi ile birkaç kez gerçekleştirilerek, daha gerçekçi sonuçlara ulaşmak hedeflenmiştir. Burada, YSA yapısı olarak; 42 giriş, 40-40 iki ara katman ve 1 çıkış (42-40-40-1) kullanılmıştır. YSA'lar aktivasyon fonksiyonu olarak hiperbolik tanjant sigmoid fonksiyonu (tansig) kullanılmıştır. Deneylerde hiperbolik tanjant sigmoid fonsiyonunun tercih edilmesinin sebebi,bilinmeyen saldırıların yakalanmasında logaritmik sigmoid fonksiyonu kullananYSA'lardan daha başarılı olmasıdır.ÇKA yapısında KDD'99 veri kümesinin %10'luk kısmında etiketlenmiş veri kullanıldığı için, 42 özelliğin tamamı kullanılmıştır.

Çalışma boyunca karşılaşılan sorunlar ve öneriler aşağıda belirtilmeye çalışılmıştır.

- Çalışmada YSA eğitimi için kullanılacak veri kümesi önemlidir. Veri kümesinde kullanılacak verilerin çok olması ve veri kümelerinin uygun hale getirmek için sayısallaştırılması uzun zaman almaktadır.
- Çalışmada kullanılan KDD'99 veri kümesi normalde yaklaşık 5 milyon satırdan oluşan bir veridir. Bu veri MS Office Excel ile açılamamış bunun yerine Açık kaynak kodlu GPL lisanslı LibreOffice Calc program kullanılmıştır. %10 luk veri kümesinin sayısallaştırılması bu şekilde yapılmıştır.
- Çalışmada oluşturulan YSA'ların eğitim sürelerinin uzun veya kısa olmaları, kullanılan bilgisayarların performansına bağlıdır. İşlemci sayısı ve bellek miktarı fazla olanbilgisayarlar ile yapılan denemelerde deney sürelerinin kısalacağı açıktır.
- Çalışmada örneklemeler seçilirken MATLAB programında işlenecek verinin matris boyutuna dikkat edilmelidirr. Matris boyutunun işlenmesi kullanılan bilgisayara bağlı olabileceğinden, örnekleme yapılırken matris boyutu dikkate alınarak örnekleme sayıları azaltılmış 3000 ve 2781 satırlık veri kümeleri kullanılmıştır.
- Çalışmada kullanılan MATLAB programının NN aracı kurulu olması önemlidir. MATLAB kurulumunda seçilen işlemci bit sayısı 32-64 bit olması önemlidir.

Daha evvel 32 bit MATLAB kullanılarak yapılan örneklerin 64 bit MATLAB programında açılmaması sorunu yaşanabilmektedir.

- Çalışmada YSA'lar aynı eğitim veri kümesi kullanmalarına rağmen, farklı zamanlarda eğitilmeleri halinde başarım oranlarının farklı olduğu gözlenmiştir. Bunun nedeni YSA'ların her yeni eğitimde saldırı kümelerini farklı ağırlıklar ile öğrenmeleridir. Bu yüzden çok sayıda deneme yapılarak başarım oranı yüksek YSA'ların tespit edilerek başarımlarda kullanılması başarımın yüksek olmasını sağlayacaktır.
- STSleri ile ilgili çalışmalarda genel kabul görmüş standart veri kümeleri kullanılmaktadır. Günümüzde güncel saldırıların tespitinde veri setlerinin oluşturulması zor ve maliyetli olduğundanveri setleri gönüllülerin ve ağ güvenlik şirketlerinin oluşturduğu kara listelerden (blacklist) temin edilebilmektedir.

KAYNAKLAR

Anderson, D. , Lunt, T. F. , Javitz, H. , Tamaru, A. , Valdes, A. , "Detecting unusual program behavior using the statistical component of the nextgeneration intrusion detection expert sistem (NIDES)", ***SRI-CSL-95-06, Menlo Park, California***, 1-22 (1995).

Anderson, D. , Frivold, T. , Tamaru, A. , Valdes, A. , "Next Generation Intrusion Detection Expert System (NIDES), SRI-CSL-95-07"***, Menlo Park, California***, 10-74 (1994).

Axelsson, S. , "Intrusion detection systems: A survey and taxonomy", ***Technical Report 99-15, Dept. of Computer Eng. , Chalmers University ofTechnology, Göteborg, Sweden,*** 1-23 (2000).

Bace, R. , Mell, P. , "Intrusion detection systems", ***Technical Report, National Institute of Standards and Technology, NIST SP300-31, Scotts Valley, CA,*** 5-46 (2001).

Barutçugil, İ., "Bilgi Yönetimi", ***Kariyer Yayıncılık***, ISNB:975 815-26-8**,**İstanbul,2002.

Cabrera, B. D. , Cabrera, L. Lewis and R. K. Mehra, "Detection and classification of intrusions and faults using sequence of system calls", ***ACMSIGMOD record***, 30(4): 25-34 (2001).

Canlı, C. "Saldırı Tespit SistemlerininİncelemesiVe Bu Bağlamda BilgiGüvenliği". Yüksek Lisans Tezi,***Marmara Üniversitesi Sosyal Bilimleri Enstitüsü,*** İstanbul, 2009.

Cannady, J. , "Artificial neural networks for misuse detection", ***Proceedings of the 1998 National Information Systems Security Conference (NISSC'98),*** Arlington, VA, 443-456 (1998).

Christoph, G. G. , Jackson, K. A. , Neuman, M. C. , Siciliano, C. L. B. , Simmonds, D. D. , Stallings, C. A. , Thompson, J. L. , "UNICORN: Misuse Detection for UNICOS", ***Proceedings of the 1995 ACM/IEEE conference on Supercomputing,*** San Diego, California, United States, 56-80 (1995).

Denning, D. E. , "An intrusion detection model", ***IEEE Transactions on Software Engineering***, 13(2): 118–131 (1987).

Erol, M., "Saldırı Tespit Sistemlerinde İstatistiksel Anormallik Belirleme Kullanımı". **İTÜ Bilgisayar Mühendisliği Bölümü**, İstanbul, 2005.

Güven, E. N, "Zeki Saldırı Tespit Sistemlerinin İncelenmesi, Tasarımı ve Gerçekleştirilmesi".Yüksek Lisans Tezi, ***Gazi Üniversitesi Fen Bilimleri Enstitüsü,*** Ankara, 2007.

KAYNAKLAR(Devam Ediyor)

Haykin, S."Neural Networks : A Comprehensive Foundation", ***Macmillan College Publishing Company***, New York,24(1999).

Ilgun, K. , "Ustat: A real-time intrusion detection sistem for Unix", ***Proceedings of the 1993 IEEE Symposium on Research in Security andPrivacy***, Oakland, California, 16-28 (1993).

Ilgun, K. , Kemmerer, R. , Porras, P. , "State Transition Analysis: A RuleBased Intrusion Detection System", ***Software Engineering,*** 21(3): 181-199 (1995).

Jackson, K. A. , "Intrusion detection sistem (IDS) product survey", ***Technical Report LA-UR-99-3883, Los Alamos National Laboratory, Los Alamos, New Mexico***, 1-96 (1999).

Jones, A. K. , "Computer System Intrusion Detection: A Survey", ***Technical Report, Computer Science Dept. , University of Virginia***, Charlottesville, Virginia, 1-21 (2000).

Knowledge Discoveri and Deliveri, "KDD Cup 1999: General Information", http://www. sigkdd. org/kddcup/index. php?section=1999&method =info,12.12.2013.

Kumar, S. , "Classification and detection of computer intrusions", Doktora Tezi, ***Department of Computer Science, Purdue University,*** West Lafayette, IN, USA, 1995.

Lee, W. , Stolfo, S. J. , Chan, P. K. , Eskin, E. , Fan, W. , Miller, M. , Hershkop, S. , Zhang, J. , "Real time data mining-based intrusion detection", ***Second{DARPA} Information Survivability Conference and Exposition (DISCEXII)***, Anaheim, CA, 89-100 (2001).

Lee, S. C. , ve Heinbuch, D. V. , "Training a Neural-Network Based Intrusion Detector to Recognize Novel Attacks", ***IEEE Transactions on sistems, man, and Cybernetics-Part A: Sistems and Humans,*** 31(4): 294-299 (2001).

Lunt, T. F. , "Automated audit trail analysis and intrusion detection: A survey", ***11th National Computer Security Conference,*** Baltimore, MD, 65-73 (1988).

Massachusetts Teknoloji Enstitüsü Lincoln Laboratuarları ,"Off-line intrusion detection evaluation data",http://www. ll. mit. edu/IST/ideval/ ,2007.

MIT(Massachusetts Teknoloji Enstitüsü Lincoln Laboratuarları) ,**"**1998 DARPA Intrusion Detection Evaluation Data Set Overview**",** http://www. ll. mit. edu/IST/ideval/data/1998/1998_data_index. html,12.12.2013.

MIT(Massachusetts Teknoloji Enstitüsü Lincoln Laboratuarları),"1999 DARPA Intrusion Detection Evaluation Data Set Overview", http://www. ll. mit. edu/IST/ideval/data/1999/1999_data_index. html,12.12.2013.

MIT(Massachusetts Teknoloji Enstitüsü Lincoln Laboratuarları) "2000 DARPA Intrusion Detection Evaluation Data Set Overview", http://www. ll. mit. edu/IST/ideval/data/2000/2000_data_index.html,12.12.2013.

McAuliffe, N. , Wolcott, D. , Schaefer, L. , Kelem, N. , Hubbard, B. , Haley, T. , "Is your computer being misused? A survey of current intrusion detection system technology",***Sixth Annual Computer Security ApplicationsConference***, Tucson, AZ, 260-272 (1990).

Marttin V., Pehlivan İ. "*ISO 27001:2005 Bilgi Guvenliği Yonetimi Standardı ve Türkiye'deki Bazı Kamu Kurulusu Uygulamaları Üzerine Bir İnceleme*", **SDÜ Mühendislik Bilimleri ve Tasarım Dergisi,**1(1): 49-56 (2010).

Moradi, M. , Zulkernine, M. , "A neural network based sistem for intrusion detection and classification of attacks",***IEEE International Conference on Advances in Intelligent Sistems - Theory and Applications***, Luxembourg- Kirchberg, Luxembourg, 148:1-6 (2004).

Mukkamala, S. , Janoski, G. , Sung, A. , "Intrusion detection using neural networks and support vector machines", ***IEEE International Joint Conferenceon Neural Networks***, IEEE Computer Society Press, 1702-1707 (2002).

Mukkamala, S. , Sung, A. H. , "Artificial Intelligent Techniques for Intrusion Detection", ***IEEE International Conference on Sistems, Man andCybernetics***, Washington D. C. , USA, 2: 1266 - 1271 (2003).

Mukkamala, S. , Sung, A. H. , "Feature Selection for Intrusion Detection using Neural Networks and Support Vector Machines", ***Journal of theTransportation Research Board of the National Academies***, Transportation Research Record No. 1822, Washington D. C. , USA, 33-39 (2003).

Murali, A. , Rao, M. , "A survey on intrusion detection approaches", ***First International Conference on Information and Communication Technologies***, IEEE Communications Society Press, 233-240 (2005).

Öztemel, E., "Yapay Sinir Ağları", ***Papatya Yayıncılık***, İstanbul, 2006.

Peddabachigari, S. , Abraham, A. , Grosan, C. , Thomas, J. , "Modeling intrusion detection sistem using hybrid intelligent sistems", ***Journal of Network and Computer Applications, Elsevier,*** 30:114–132 (2007).

KAYNAKLAR(Devam Ediyor)

Porras, P. A. , "STAT: A State Transition Analysis Tool for intrusion detection", Yüksek Lisans Tezi, ***Computer Science Department, University of California***, Santa Barbara, 1-150 (1992).

Porras, P. A. , Neumann, P. G. , "EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances", ***20th NIST-NCSC NationalInformation Sistems Security Conference,*** Baltimore, Maryland, 353-365(1997).

Patcha, A. , Park, J. M. , "An overview of anomaly detection techniques: Existing solutions and latest technological trends", ***Computer Networks,*** 51(12): 3448-3470 (2007).

Paxson, V. , "Bro: A sistem for detecting network intruders in real-time", ***7th USENIX Security Symposium,*** San Antonio, TX, 32-53 (1998).

Ryan, J. , Lin, M. J. , Mikkulainen, R. , "Intrusion Detection with Neural Networks", ***Advances in Neural Information Processing sistems 10***, Cambridge, MA, MIT Press, 1-7 (1998).

Sağıroğlu, Ş. , Beşdok, E. , Erler, M. , "Mühendislikte yapay zeka uygulamaları-1: Yapay sinir ağları", ***Ufuk Kitabevi,*** Kayseri, 10-100 (2003).

Smaha S. E. , "Haystack: An intrusion detection sistem", ***In Fourth Aerospace Computer Security Applications Conference,*** Tracor Applied Science Inc. Austin, Texas, 37-44 (1988).

Snapp, S. R. , Brentano, J. , Dias, G. V. , Goan, T. L. , Heberlein, L. T. , Ho, C. , Levitt, K. N. , Mukherjee, B. , Smaha, S. E. , Grance, T. , Teal, D. M. , Mansur, D. , "DIDS (Distributed Intrusion Detection Sistem) - Motivation, Architecture, and an Early Prototype", ***14th National Computer Security Conference,*** California,167- 176 (1991).

Snort - open source network intrusion prevention and detection sistem, "Snort Downloads", http://www. snort. org/dl/ ,01.05.2014.

SRI International's Computer Science Lab, "IDES history", http://www. csl. sri. com/programs/intrusion/history. html#IDES,01.05.2014.

Sağıroğlu, Ş. , Beşdok, E. , Erler, M. , "Mühendislikte yapay zeka uygulamaları-1: Yapay sinir ağları", ***Ufuk Kitabevi,*** Kayseri, 10-100 (2003).

KAYNAKLAR(Devam Ediyor)

Sundaram, A. , "An introduction to intrusion detection", ***Crossroads: The ACM Student Magazine,*** New York, USA, 2(4): 3-7 (1996).

Şeker H.,Uçar O., "Güvenlik Riskleri ve Saldırı Yöntemleri", www. cehturkiye. com, 12.12.2013,

Tanrıkulu, H, "Saldırı Tespit Sistemlerinde Yapay Sinir Ağların Kullanılması", Yüksek Lisans Tezi***, Ankara Üniversitesi Fen Bilimleri Enstitüsü,*** Ankara,2009.

Yıldırımoğlu M., "Her Yönüyle İnternetin Altyapısı TCP/IP", ***Pusula Yayıncılık***, İstanbul, 2012.

Yurtoğlu, H. "Yapay sinir ağları metodolojisi ile öngörü modellemesi: bazı makroekonomik değişkenler için Türkiye örneği". ***Uzmanlık Tezi, DevletPlanlama Teşkilatı, Ekonomik Modeller ve Stratejik Araştırmalar GenelMüdürlüğü,*** Ankara, 3-43.(2005).

Vaccarro, H. S. ve Liepins, G. E. "Detection of Anomalous Computer Session Activity", ***IEEE Symposium on Research in Security and Privacy***, Oakland, California, 280-289 (1989).

EKLER

EK 1. Bilinen Atakların Bulunması için Kullanılan Eğitim Veri Kümeleri

atak3000.txtsaldırı dosyası (3000*42)
(örnekleme)

1. 0 0 0 4 239 486 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 19 19 1.00 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0
2. 0 0 0 4 235 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 29 29 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
3. 0 0 0 4 219 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 39 39 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
4. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 49 49 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
5. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 59 59 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
6. 0 0 0 4 212 1940 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 2 0.00 0.00 0.00 0.00 1.00 0.00 1.00 1 69 1.00 0.00 1.00 0.04 0.00 0.00 0.00 0.00 0
7. 0 0 0 4 159 4087 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 5 5 0.00 0.00 0.00 0.00 1.00 0.00 0.00 11 79 1.00 0.00 0.09 0.04 0.00 0.00 0.00 0.00 0
8. 0 0 0 4 210 151 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 8 89 1.00 0.00 0.12 0.04 0.00 0.00 0.00 0.00 0
9. 0 0 0 4 212 786 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 8 99 1.00 0.00 0.12 0.05 0.00 0.00 0.00 0.00 0

.....

.....

.....

2990. 0 0 13 4 334 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 10 1.00 0.00 1.00 0.30 0.00 0.00 0.00 0.00 21
2991. 0 0 13 4 334 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 3 11 1.00 0.00 1.00 0.27 0.00 0.00 0.00 0.00 21
2992. 0 0 13 4 334 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 4 12 1.00 0.00 1.00 0.25 0.00 0.00 0.00 0.00 21
2993. 0 0 6 4 36 197 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 255 1 0.00 0.05 0.00 0.00 0.39 0.00 0.05 0.00 22
2994. 156 0 6 4 950 2551 0 0 0 18 0 1 0 0 0 0 0 21 0 0 0 0 0 1 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 218 1 0.00 0.03 0.00 0.00 0.01 0.00 0.07 0.00 22
2995. 0 0 13 4 0 7181 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 1 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2996. 0 0 13 4 0 848 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 2 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2997. 9 0 13 4 0 5153460 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 0.00 0.00 0.00 0.00 1.00 0.00 0.00 3 3 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2998. 10 0 13 4 0 5155468 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 4 4 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2999. 10 0 13 4 0 5151385 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 5 5 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
3000. 9 0 13 4 0 5150836 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 6 6 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22

ornek200.txtsaldırı dosyası (200*42)
(örnekleme)

1. 0 0 0 4 239 486 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 19 19 1.00 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0
2. 0 0 0 4 235 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 29 29 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
3. 0 0 0 4 219 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 39 39 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
4. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 49 49 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
5. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 59 59 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0.

.....

.....

.....

191. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 3 0.00 0.00 0.00 0.00 1.00 0.00 0.67 1 11 1.00 0.00 1.00 0.55 0.00 0.00 0.00 0.00 14
192. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 4 0.00 0.00 0.00 0.00 1.00 0.00 0.50 2 12 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14
193. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 13 1.00 0.00 1.00 0.54 0.00 0.00 0.00 0.00 14
194. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 14 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14
195. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 3 0.00 0.00 0.00 0.00 1.00 0.00 0.67 1 15 1.00 0.00 1.00 0.53 0.00 0.00 0.00 0.00 14
196. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 4 0.00 0.00 0.00 0.00 1.00 0.00 0.50 2 16 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14
197. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 17 1.00 0.00 1.00 0.53 0.00 0.00 0.00 0.00 14
198. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 18 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14
199. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 19 1.00 0.00 1.00 0.53 0.00 0.00 0.00 0.00 14
200. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 20 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14

EK 2. Bilinmeyen Atakların Bulunması için Kullanılan Eğitim Veri Kümeleri

atak2781.txt saldırı dosyası (2781*42)
(örnekleme)

1. 0 0 0 4 239 486 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 19 19 1.00 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0
2. 0 0 0 4 235 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 29 29 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
3. 0 0 0 4 219 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 39 39 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
4. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 49 49 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
5. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 59 59 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
6. 0 0 0 4 212 1940 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 2 0.00 0.00 0.00 0.00 1.00 0.00 1.00 1 69 1.00 0.00 1.00 0.04 0.00 0.00 0.00 0.00 0
7. 0 0 0 4 159 4087 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 5 5 0.00 0.00 0.00 0.00 1.00 0.00 0.00 11 79 1.00 0.00 0.09 0.04 0.00 0.00 0.00 0.00 0
8. 0 0 0 4 210 151 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 8 89 1.00 0.00 0.12 0.04 0.00 0.00 0.00 0.00 0
9. 0 0 0 4 212 786 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 8 99 1.00 0.00 0.12 0.05 0.00 0.00 0.00 0.00 0
10. 0 0 0 4 210 624 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 18 18 0.00 0.00 0.00 0.00 1.00 0.00 0.00 18 109 1.00 0.00 0.06 0.05 0.00 0.00 0.00 0.00 0

.....

.....

.....

2771. 0 0 13 4 334 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 10 1.00 0.00 1.00 0.30 0.00 0.00 0.00 0.00 21
2772. 0 0 13 4 334 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 3 11 1.00 0.00 1.00 0.27 0.00 0.00 0.00 0.00 21
2773. 0 0 13 4 334 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 4 12 1.00 0.00 1.00 0.25 0.00 0.00 0.00 0.00 21
2774. 0 0 6 4 36 197 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 255 1 0.00 0.05 0.00 0.00 0.39 0.00 0.05 0.00 22
2775. 156 0 6 4 950 2551 0 0 0 18 0 1 0 0 0 0 21 0 0 0 0 1 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 218 1 0.00 0.03 0.00 0.00 0.01 0.00 0.07 0.00 22
2776. 0 0 13 4 0 7181 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 1 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2777. 0 0 13 4 0 848 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 2 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2778. 9 0 13 4 0 5153460 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 0.00 0.00 0.00 0.00 1.00 0.00 0.00 3 3 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2779. 10 0 13 4 0 5155468 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 4 4 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2780. 10 0 13 4 0 5151385 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 5 5 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22
2781. 9 0 13 4 0 5150836 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 6 6 1.00 0.00 1.00 0.00 0.00 0.00 0.00 0.00 22

ornek 222.txt saldırı dosyası (222*42)
(örnekleme)

1. 0 0 0 4 239 486 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 19 19 1.00 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0
2. 0 0 0 4 235 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 29 29 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
3. 0 0 0 4 219 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 39 39 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0
4. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 49 49 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
5. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 59 59 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0
6. 0 0 0 4 212 1940 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 2 0.00 0.00 0.00 0.00 1.00 0.00 1.00 1 69 1.00 0.00 1.00 0.04 0.00 0.00 0.00 0.00 0
7. 0 0 0 4 159 4087 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 5 5 0.00 0.00 0.00 0.00 1.00 0.00 0.00 11 79 1.00 0.00 0.09 0.04 0.00 0.00 0.00 0.00 0.

.....

.....

.....

216. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 154 1 0.01 0.94 0.95 0.00 0.95 1.00 0.00 0.00 11
217. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 164 1 0.01 0.94 0.95 0.00 0.95 1.00 0.00 0.00 11
218. 0 0 27 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 174 1 0.01 0.94 0.95 0.00 0.95 1.00 0.00 0.00 11
219. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 184 1 0.01 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11
220. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 194 1 0.01 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11
221. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 204 1 0.00 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11
222. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 214 1 0.00 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11

EK 3. Bilinen Atakların Bulunması için Kullanılan Ayrı Ayrı Eğitim Veri Kümeleri

normal0.txtsaldırı dosyası (260*42)
(örnekleme)

1. 0 0 0 4 239 486 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 19 19 1.00 0.00 0.05 0.00 0.00 0.00 0.00 0.00 0.00 0
2. 0 0 0 4 235 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 8 8 0.00 0.00 0.00 0.00 1.00 0.00 0.00 29 29 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0.00 0
3. 0 0 0 4 219 1337 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 39 39 1.00 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0.00 0
4. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 49 49 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0
5. 0 0 0 4 217 2032 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 59 59 1.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0
6. 0 0 0 4 212 1940 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 2 0.00 0.00 0.00 0.00 1.00 0.00 1.00 1 69 1.00 0.00 1.00 0.04 0.00 0.00 0.00 0.00 0.00 0
7. 0 0 0 4 159 4087 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 5 5 0.00 0.00 0.00 0.00 1.00 0.00 0.00 11 79 1.00 0.00 0.09 0.04 0.00 0.00 0.00 0.00 0.00 0

.....

.....

.....

250. 0 0 0 4 267 9317 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 43 255 1.00 0.00 0.02 0.05 0.00 0.00 0.00 0.00 0
251. 0 0 0 4 156 4027 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 3 17 0.00 0.00 0.00 0.00 1.00 0.00 0.12 53 255 1.00 0.00 0.02 0.05 0.00 0.00 0.00 0.00 0
252. 0 0 0 4 218 437 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 63 255 1.00 0.00 0.02 0.04 0.00 0.00 0.00 0.00 0
253. 0 0 0 4 298 9442 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 4 4 0.00 0.00 0.00 0.00 1.00 0.00 0.00 73 255 1.00 0.00 0.01 0.04 0.00 0.00 0.00 0.00 0
254. 0 0 0 4 257 4027 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 7 7 0.00 0.00 0.00 0.00 1.00 0.00 0.00 83 255 1.00 0.00 0.01 0.04 0.00 0.00 0.00 0.00 0
255. 0 0 0 4 141 4027 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 93 255 1.00 0.00 0.01 0.04 0.00 0.00 0.00 0.00 0
256. 0 0 0 4 213 261 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 103 255 1.00 0.00 0.01 0.04 0.00 0.00 0.00 0.00 0
257. 0 0 0 4 284 43129 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 4 4 0.00 0.00 0.00 0.00 1.00 0.00 0.00 113 255 1.00 0.00 0.01 0.04 0.00 0.00 0.00 0.00 0
258. 0 0 0 4 183 8654 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 4 4 0.25 0.25 0.00 0.00 1.00 0.00 0.00 123 255 1.00 0.00 0.01 0.04 0.01 0.00 0.00 0.00 0
259. 0 0 0 4 191 3727 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 133 255 1.00 0.00 0.01 0.03 0.01 0.00 0.00 0.00 0
260. 0 0 0 4 183 8654 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 4 4 0.00 0.00 0.00 0.00 1.00 0.00 0.00 143 255 1.00 0.00 0.01 0.03 0.01 0.00 0.00 0.00 0

imap5.txt saldırı dosyası (20*42)

1. 31 0 25 4 1345 10036 0 0 0 0 0 1 16 0 0 16 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 255 1 0.00 0.01 0.00 0.00 0.00 0.00 0.03 0.00 5
2. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 160 1.00 0.58 0.00 0.03 1.00 0.00 0.28 1 1 1.00 0.00 1.00 0.00 1.00 1.00 0.00 0.00 5
3. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 161 1.00 0.58 0.00 0.03 1.00 0.00 0.28 2 2 1.00 0.00 1.00 0.00 1.00 1.00 0.00 0.00 5
4. 0 0 25 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 162 0.67 0.58 0.00 0.03 1.00 0.00 0.28 3 3 1.00 0.00 1.00 0.00 0.67 0.67 0.00 0.00 5
5. 0 0 25 4 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 4 163 0.50 0.58 0.00 0.03 1.00 0.00 0.28 4 4 1.00 0.00 1.00 0.00 0.50 0.50 0.00 0.00 5
6. 41 0 25 4 1334 162 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 5 5 1.00 0.00 0.20 0.00 0.40 0.40 0.00 0.00 5
7. 0 0 25 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 6 6 1.00 0.00 0.17 0.00 0.50 0.50 0.00 0.00 5
8. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 122 1.00 0.76 0.00 0.04 1.00 0.00 0.37 7 7 1.00 0.00 0.71 0.00 0.57 0.57 0.00 0.00 5
9. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 123 1.00 0.76 0.00 0.04 1.00 0.00 0.37 8 8 1.00 0.00 0.75 0.00 0.62 0.62 0.00 0.00 5
10. 0 0 25 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 124 0.67 0.76 0.00 0.04 1.00 0.00 0.36 9 9 1.00 0.00 0.78 0.00 0.56 0.56 0.00 0.00 5
11. 31 0 25 4 1345 10036 0 0 0 0 0 1 16 0 0 16 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 255 1 0.00 0.01 0.00 0.00 0.00 0.00 0.03 0.00 5
12. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 160 1.00 0.58 0.00 0.03 1.00 0.00 0.28 1 1 1.00 0.00 1.00 0.00 1.00 1.00 0.00 0.00 5
13. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 161 1.00 0.58 0.00 0.03 1.00 0.00 0.28 2 2 1.00 0.00 1.00 0.00 1.00 1.00 0.00 0.00 5
14. 0 0 25 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 162 0.67 0.58 0.00 0.03 1.00 0.00 0.28 3 3 1.00 0.00 1.00 0.00 0.67 0.67 0.00 0.00 5
15. 0 0 25 4 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 4 163 0.50 0.58 0.00 0.03 1.00 0.00 0.28 4 4 1.00 0.00 1.00 0.00 0.50 0.50 0.00 0.00 5
16. 41 0 25 4 1334 162 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 5 5 1.00 0.00 0.20 0.00 0.40 0.40 0.00 0.00 5
17. 0 0 25 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 6 6 1.00 0.00 0.17 0.00 0.50 0.50 0.00 0.00 5
18. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 122 1.00 0.76 0.00 0.04 1.00 0.00 0.37 7 7 1.00 0.00 0.71 0.00 0.57 0.57 0.00 0.00 5
19. 0 0 25 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 123 1.00 0.76 0.00 0.04 1.00 0.00 0.37 8 8 1.00 0.00 0.75 0.00 0.62 0.62 0.00 0.00 5
20. 0 0 25 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 124 0.67 0.76 0.00 0.04 1.00 0.00 0.36 9 9 1.00 0.00 0.78 0.00 0.56 0.56 0.00 0.00 5

EK 3. Bilinen Atakların Bulunması için Kullanılan Ayrı Ayrı Eğitim Veri Kümeleri (Devam Ediyor)

Pod14.txt saldırı dosyası (120*42)
(örnekleme)

1. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 85 1 0.01 0.05 0.01 0.00 0.00 0.00 0.00 0.00 14
2. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 86 2 0.02 0.05 0.02 0.00 0.00 0.00 0.00 0.00 14
3. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 0.00 0.00 0.00 0.00 1.00 0.00 0.00 87 3 0.03 0.05 0.03 0.00 0.00 0.00 0.00 0.00 14
4. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4 4 0.00 0.00 0.00 0.00 1.00 0.00 0.00 88 4 0.05 0.05 0.05 0.00 0.00 0.00 0.00 0.00 14
5. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 5 0.00 0.00 0.00 0.00 1.00 0.00 0.00 89 5 0.06 0.04 0.06 0.00 0.00 0.00 0.00 0.00 14
6. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 6 6 0.00 0.00 0.00 0.00 1.00 0.00 0.00 90 6 0.07 0.04 0.07 0.00 0.00 0.00 0.00 0.00 14
7. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 7 0.00 0.00 0.00 0.00 1.00 0.00 0.00 91 7 0.08 0.04 0.08 0.00 0.00 0.00 0.00 0.00 14.

.....

.....

.....

115. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 3 0.00 0.00 0.00 0.00 1.00 0.00 0.67 1 15 1.00 0.00 1.00 0.53 0.00 0.00 0.00 0.00 14
116. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 4 0.00 0.00 0.00 0.00 1.00 0.00 0.50 2 16 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14
117. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 17 1.00 0.00 1.00 0.53 0.00 0.00 0.00 0.00 14
118. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 18 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14
119. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 1 19 1.00 0.00 1.00 0.53 0.00 0.00 0.00 0.00 14
120. 0 2 9 4 1480 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 0.00 0.00 0.00 0.00 1.00 0.00 0.00 2 20 1.00 0.00 1.00 0.50 0.00 0.00 0.00 0.00 14

nmap11.txt saldırı dosyası (22*42)

1. 0 2 7 4 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0.00 0.00 0.00 0.00 1.00 0.00 0.00 4 2 0.50 0.75 0.50 0.00 0.00 0.00 0.00 0.00 11
2. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 1 1.00 1.00 0.00 0.00 0.50 1.00 0.00 14 1 0.07 0.71 0.50 0.00 0.50 1.00 0.00 0.00 11
3. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 24 1 0.04 0.75 0.67 0.00 0.67 1.00 0.00 0.00 11
4. 0 0 5 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 34 1 0.03 0.74 0.76 0.00 0.76 1.00 0.00 0.00 11
5. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 44 1 0.02 0.77 0.82 0.00 0.82 1.00 0.00 0.00 11
6. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 54 1 0.02 0.81 0.85 0.00 0.85 1.00 0.00 0.00 11
7. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 64 1 0.02 0.84 0.88 0.00 0.88 1.00 0.00 0.00 11
8. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 74 1 0.01 0.86 0.89 0.00 0.89 1.00 0.00 0.00 11
9. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 84 1 0.01 0.88 0.90 0.00 0.90 1.00 0.00 0.00 11
10. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 94 1 0.01 0.89 0.91 0.00 0.91 1.00 0.00 0.00 11
11. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 104 1 0.01 0.90 0.92 0.00 0.92 1.00 0.00 0.00 11
12. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 114 1 0.01 0.91 0.93 0.00 0.93 1.00 0.00 0.00 11
13. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 124 1 0.01 0.92 0.94 0.00 0.94 1.00 0.00 0.00 11
14. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 134 1 0.01 0.93 0.94 0.00 0.94 1.00 0.00 0.00 11
15. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 144 1 0.01 0.93 0.94 0.00 0.94 1.00 0.00 0.00 11
16. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 154 1 0.01 0.94 0.95 0.00 0.95 1.00 0.00 0.00 11
17. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 164 1 0.01 0.94 0.95 0.00 0.95 1.00 0.00 0.00 11
18. 0 0 27 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 174 1 0.01 0.94 0.95 0.00 0.95 1.00 0.00 0.00 11
19. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 184 1 0.01 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11
20. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 194 1 0.01 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11
21. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 204 1 0.00 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11
22. 0 0 11 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1.00 1.00 0.00 0.00 1.00 0.00 0.00 214 1 0.00 0.95 0.96 0.00 0.96 1.00 0.00 0.00 11

EK 4. Kullanılan MATLAB Kodları

Bilinen Saldırı-Matlab Kodları

```
%Bilinen ataklarin bulunmasi
load ornek200. txt; % Test Seti
load atak3000. txt; % Normal Imap ve Pod Ataklarinin Egitim Seti
P1=atak3000(1:3000, 1:42);
T1=atak3000(1:3000, 42);
a1=ornek200(1:200, 1:42);
s1=ornek200(1:200, 42);
net1= newff(minmax(P1'), [42 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'});
net1. trainParam. epochs=50;
net1. trainParam. goal = 1e-6;
net1=train (net1, P1', T1');
y1=sim(net1, a1');
figure;
plot(y1, '*');
hold
plot(s1', 'r');
title('[42 40 40 1] Bilinen Ataklarin Tespit Edilmesi (Normal(0), Imap(5), Pod(14)');
hold
```

Bilinmeyen Saldırı-Matlab Kodları

```
%Bilinmeyen ataklarin bulunmasi
load ornek222. txt; % Test Seti
load atak2781. txt; % Nmap hariç tüm atakların olduğu veri seti
P1=atak2781(1:2781, 1:42);
T1=atak2781(1:2781, 42);
a1=ornek222(1:222, 1:42);
s1=ornek222(1:222, 42);
net1= newff(minmax(P1'), [42 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'});
net1. trainParam. epochs=50;
net1. trainParam. goal = 1e-6;
net1=train (net1, P1', T1');
y1=sim(net1, a1');
figure;
plot(y1, '*');
hold
plot(s1', 'r');
title('[42 40 40 1] Bilinen Ataklarin Tespit Edilmesi (Normal(0), Imap(5), Pod(14)');
hold
```

EK 4. Kullanılan MATLAB Kodları (Devam Ediyor)

Bilinmeyen Saldırı-Matlab Kodları

```
%Ayri Ayri Egitim

load ornek200. txt; %% Test Seti
load normal0. txt; %% Normal Trafik Eğitim Seti
load imap. txt; %% Imap Atağı
load pod. txt; %% Pod Atağı
P1=normal0(1:260, 1:42);
T1=normal0(1:260, 42);
P2=imap5(1:20, 1:42);
T2=imap5(1:20, 42);
P3=pod14(1:120, 1:42);
T3=pod14(1:120, 42);
a1=ornek200(1:200, 1:42);
s1=ornek200(1:200, 42);
net1= newff(minmax(P1'), [41 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'}); %% Normal
net2= newff(minmax(P2'), [41 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'}); %% Imap
net3= newff(minmax(P3'), [41 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'}); %% Pod
net1. trainParam. epochs=50;
net1. trainParam. goal = 1e-5;
net1=train (net1, P1', T1');
net2. trainParam. epochs=50;
net2. trainParam. goal = 1e-5;
net2=train (net2, P2', T2');
net3. trainParam. epochs=50;
net3. trainParam. goal = 1e-5;
net3=train (net3, P3', T3');
y1=sim(net1, a1');
y2=sim(net2, a1');
y3=sim(net3, a1');
figure;plot(y1, '*');
hold
plot(s1', 'r');title(' NORMAL [41 40 40 1] '); figure; plot(y2, '*');
hold
plot(s1', 'r');title(' IMAP [41 40 40 1] '); figure; plot(y3, '*');
hold
plot(s1', 'r');title(' POD [41 40 40 1]');
hold
```

EK 4. Kullanılan MATLAB Kodları (Devam Ediyor)

Bilinmeyen Saldırı-Matlab Kodları

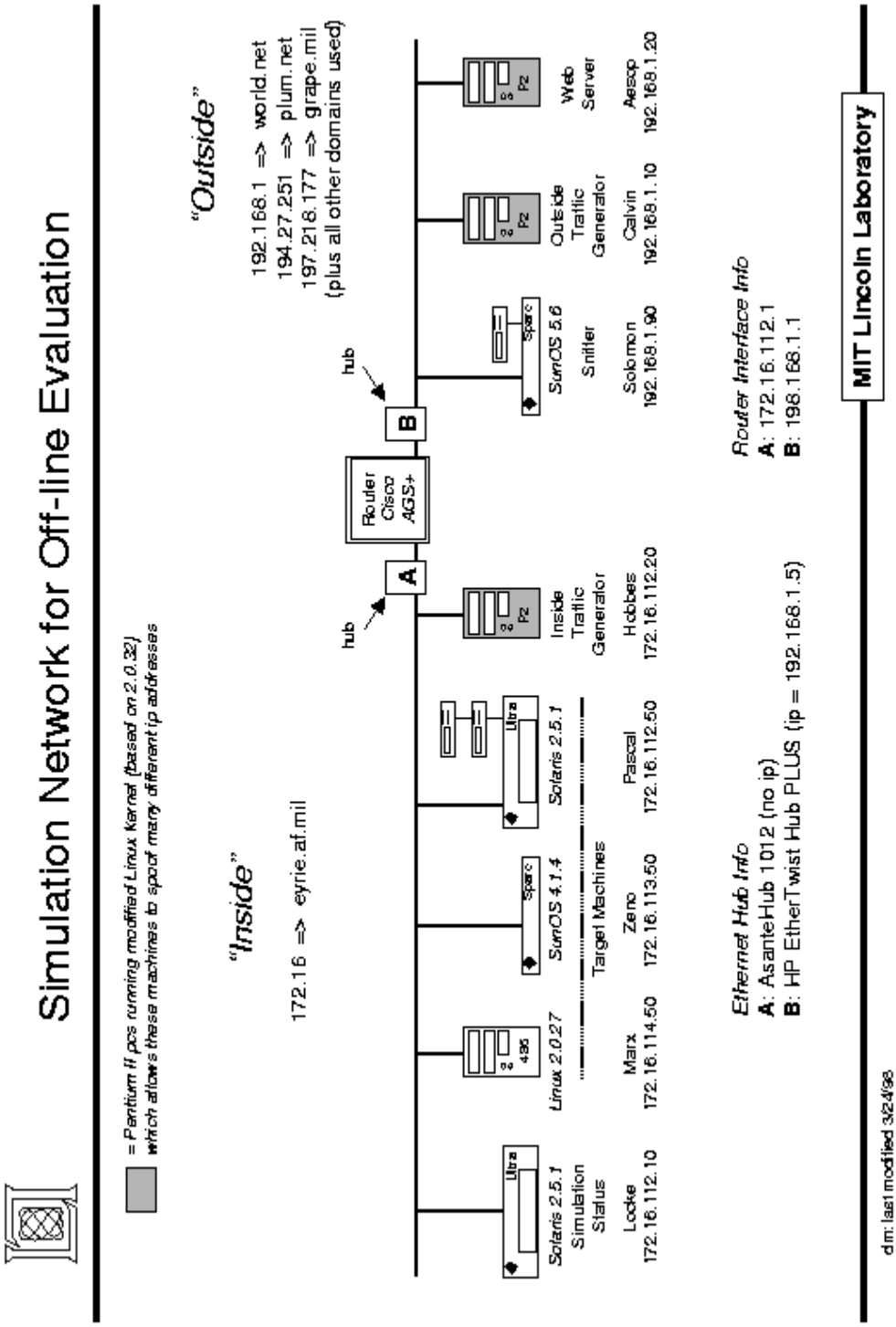
```
% Ayri Ayri Egitim-Bilinmeyen Atak
load ornek222. txt; %% Test Seti
load normal0. txt; %Normal Trafik Eğitim Seti
load imap. txt; %% Imap Atağı
load pod. txt; %% Pod Atağı
P1=normal0(1:260, 1:42);
T1=normal0(1:260, 42);
P2=imap5(1:20, 1:42);
T2=imap5(1:20, 42);
P3=pod14(1:120, 1:42);
T3=pod14(1:120, 42);
a1=ornek222(1:222, 1:42);
s1=ornek222(1:222, 42);
net1= newff(minmax(P1'), [42 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'}); %% Normal
net2= newff(minmax(P2'), [42 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'}); %% Imap
net3= newff(minmax(P3'), [42 40 40 1], {'tansig' 'tansig' 'tansig' 'purelin'}); %% Pod
net1. trainParam. epochs=50;
net1. trainParam. goal = 1e-5;
net1=train (net1, P1', T1');
net2. trainParam. epochs=50;
net2. trainParam. goal = 1e-5;
net2=train (net2, P2', T2');
net3. trainParam. epochs=50;
net3. trainParam. goal = 1e-5;
net3=train (net3, P3', T3');
y1=sim(net1, a1');
y2=sim(net2, a1');
y3=sim(net3, a1');
figure;plot(y1, '*');
hold
plot(s1', 'r');title(' NORMAL [42 40 40 1] '); figure; plot(y2, '*');
hold
plot(s1', 'r');title(' IMAP [42 40 40 1] '); figure; plot(y3, '*');
hold
plot(s1', 'r');title(' POD [42 40 40 1]');
hold
```

EK 4. Kullanılan MATLAB Kodları (Devam Ediyor)

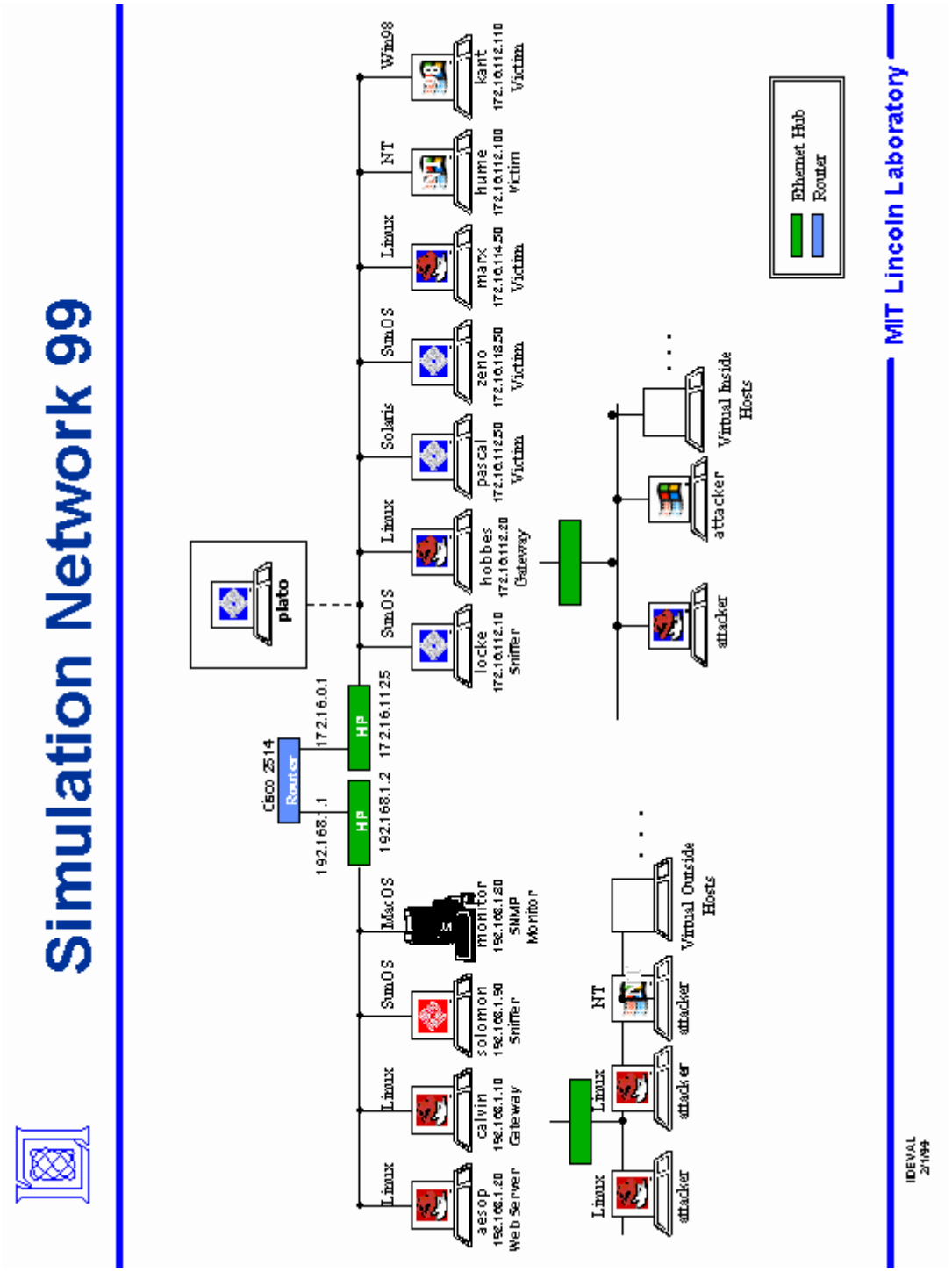
```
%Test-1-Egitim seti-Bilinen ornek200 için Imap Basarimi
imapy=0
for nimap=131:140
imapy= imapy + y1(nimap);
end
ortimap=imapy/(nimap-130)
yuzde_imap_basarim=((5-abs(5-ortimap))/5)*100
```

```
Test-1-Egitim seti-Bilinen ornek200
normaly=0
for nnormal=1:130
normaly= normaly + y1(nnormal);
end
ortnormaly= normaly/(nnormal-0)
yuzde_normal_basarim=(abs(1-abs(ortnormaly)))*100
```

EK 5. DARPA Veri Kümesi İçin Oluşturulan Senaryo



EK 6. KDD'99 Veri Kümesi İçin Oluşturulan Senaryo



ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı : Vedat MARTTİN
Doğum Yeri ve Tarihi : Eskişehir, 04. 01. 1982

Eğitim Durumu

Lisans Öğrenimi :Kocaeli Üniversitesi, Teknik Eğitim Fakültesi,
Elektronik Öğretmenliği (2001-2004)
Sakarya Üniversitesi, Teknik Eğitim Fakültesi,
Elektronik ve Bilgisayar Eğitimi
Elektronik Öğretmenliği (2004-2007)

İş Deneyimi

Çalıştığı Kurumlar : 1) Sakarya Üniversitesi (2004-2008)
2) Eskişehir Osmangazi Üniversitesi (2008-2010)
3) Bilecik Şeyh Edebali Üniversitesi, (2010-)

İletişim

Adres : Bilecik Şeyh Edebali Üniversitesi, Bilgi İşlem Dairesi Başkanlığı, Gülümbe Kampüsü, BİLECİK
Tel: : 0228. 2141982
E-Posta Adresi : vedat.marttin@bilecik.edu.tr

Tarih: / /

İmza