HAVA HARP OKULU
HAVACILIK VE UZAY TEKNOLOJİLERİ ENSTİTÜSÜ

DAĞITILMIŞ SİSTEMLERDE SALDIRI TESPİT SİSTEMLERİ

YÜKSEK LİSANS TEZİ

Okan CAN

Bilgisayar Mühendisliği Ana Bilim Dalı Başkanlığı

Siber Güvenlik Programı

HAZİRAN 2015

HAVA HARP OKULU
HAVACILIK VE UZAY TEKNOLOJİLERİ ENSTİTÜSÜ

DAĞITILMIŞ SİSTEMLERDE SALDIRI TESPİT SİSTEMLERİ

YÜKSEK LİSANS TEZİ

Okan CAN
113102

Bilgisayar Mühendisliği Ana Bilim Dalı Başkanlığı

Siber Güvenlik Programı

Tez Danışmanı: Doç. Dr. Özgür Koray ŞAHİNGÖZ

HAZİRAN 2015

Hava Harp Okulu Havacılık ve Uzay Teknolojileri Enstitüsü'nün 113102 numaralı Yüksek Lisans Öğrencisi **Okan CAN**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı **"DAĞITILMIŞ SİSTEMLERDE SALDIRI TESPİT SİSTEMLERİ"** başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı:	**Doç. Dr. Özgür Koray ŞAHİNGÖZ** Hava Harp Okulu	.....
Jüri Üyeleri:		.....
	Hava Harp Okulu	
		.....
	Hava Harp Okulu	

Teslim Tarihi : 15 Mayıs 2015
Savunma Tarihi : Haziran 2015

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

Haziran 2015 Okan CAN

Eşime ve biricik kızıma,

ÖNSÖZ

Tez çalışmamı hazırlama aşamasında bana her konuda yardımcı olan, her soruma sabırla yanıt veren ve tez çalışmasının eksik yönlerinin ortaya çıkarılması ve giderilmesi kapsamında değerli katkılarda bulunan tez danışmanım Doç. Dr. Özgür Koray ŞAHİNGÖZ başta olmak üzere, Bilgisayar Mühendisliği Ana Bilim Dalı Başkanı Doç. Dr. Güray YILMAZ'a, Bilgisayar Mühendisliği Ana Bilim Dalı Başkanlığı öğretim üyelerine, yüksek lisans tez çalışmam esnasında engin bilgilerinden faydalandığım hocam Yrd.Doç.Dr Muhammed Ali AYDIN'a ve çalışmam süresince büyük ilgi ve fedakârlıklarıyla her zaman yanımda olan eşime, tatlı dili ve pırıltılı gözleriyle her zaman mutluluk kaynağım olan güzeller güzeli kızıma, bana bu imkânı sağlayan ülkem ve Türk Silahlı Kuvvetleri'ne, her türlü ihtiyacımda yanımda olan yuvam Hava Harp Okulu'na teşekkürlerimi bir borç bilir, sonsuz minnettarlığımı sunarım.

Haziran 2015 Okan CAN

İÇİNDEKİLER

Sayfa

ÖNSÖZ ..... **ix**
İÇİNDEKİLER ..... **xi**
KISALTMALAR ..... **xiii**
ÇİZELGE LİSTESİ ..... **xv**
ŞEKİL LİSTESİ ..... **xvii**
SEMBOL LİSTESİ ..... **xix**
ÖZET ..... **xxi**
SUMMARY ..... **xxiii**
1. GİRİŞ **1**
1.1 Tezin Amacı 1
2. LİTERATÜR ARAŞTIRMASI **3**
2.1 *KAA*' larda Siber Saldırılar 4
2.1.1 Hizmet engelleme saldırıları 6
2.1.2 Yeniden yönlendirme 6
2.1.3 Seçmeli iletim 7
2.1.4 Karadelik saldırıları 8
2.1.5 Sybil saldırıları 8
2.1.6 Solucan deliği 8
2.1.7 Hello baskını saldırıları 9
2.2 Genel Saldırı Tespit Sistemi Yapısı 9
2.2.1 Anormallik tespiti 11
2.2.2 Kötüye kullanım tespiti 12
2.3 *KAA*' larda *STS* Yaklaşımları 12
2.3.1 *KAA* için anormallik tespiti 13
2.3.2 *KAA* için kötüye kullanım tespiti 14
2.3.3 *KAA* için karma tespit yaklaşımı 14
3. KABLOSUZ ALGILAYICI AĞLARDA GÜVENLİK VE SALDIRI TESPİT SİSTEMLERİ **15**
3.1 *KAA*' ın Sahip Olduğu Kısıtlar Ve Bu Kısıtların Güvenlik Mekanizmasına Olan Etkileri 15
3.1.1 Kısıtlı kaynaklar 16
3.1.2 Güvenli olmayan iletişim ortamı 17
3.1.3 Çevresel etkiler 18
3.2 *KAA* İçin Güvenlik Gereksinimleri 19
3.2.1 Veri gizliliği 19

3.2.2 Veri bütünlüğü 20
3.2.3 Erişilebilirlik 21
3.2.4 Veri güncelliği 21
3.2.5 Kendini yönetebilme 22
3.2.6 Senkronizasyon 22
3.2.7 Güvenli yer bildirimi 22
3.2.8 Parolalaşma 23
3.3 *KAA* İçin Gerçekleşebilecek Saldırılar 23
3.3.1 *DOS* saldırıları ve çeşitleri 25
3.3.2 Sybil saldırıları 26
3.3.3 Trafik analiz saldırıları 26
3.3.4 Fiziksel saldırılar 26
3.3.5 Düğümler ile gerçekleştirilen saldırılar 27
3.3.6 Veri gizliliğini hedef alan saldırılar 27
3.4 *KAA* İçin Savunma Mekanizmaları Ve Saldırı Tespit Sistemleri *(STS)* 28
3.4.1 Saldırı tespit sistemleri *(STS)* 29
3.4.1.1 Ağ tabanlı saldırı tespit sistemleri *(ATSTS)* 30
3.4.1.2 Sunucu tabanlı saldırı tespit sistemleri *(STSTS)* 30
3.4.1.3 Karma saldırı tespit sistemleri 30
3.4.1.4 Anormallik tespiti 31
3.4.1.5 Kötüye kullanım tespiti 31
3.4.1.6 Karma saldırı tespiti 32
3.4.2 *KAA* uygulamalarında saldırı tespiti 32
4. SALDIRI TESPİTİ YAKLAŞIMI OLARAK YAPAY SİNİR AĞLARI 37
4.1 Yapay Sinir Ağı Yapısı Ve Bileşenleri 40
4.1.1 Girdiler 41
4.1.2 Propogasyon fonksiyonu 41
4.1.3 Aktivasyon (Eşikleme) fonksiyonu 41
4.1.4 Çıktılar 43
4.2 Yapay Sinir Ağı Öğrenme Yaklaşımı 43
5. ÖNERİLEN SİSTEM VE DENEYSEL SONUÇLAR 47
5.1 KDD 99 Cup Veri Seti 47
5.2 Akış Şeması 50
5.3 Normalizasyon İşlemleri Ve Öznitelik Seçimi 50
5.4 Gerçekleştirilen İş Akışı 53
5.5 Deneysel Sonuçlar 54
6. SONUÇ VE ÖNERİLER 57
KAYNAKLAR 61
ÖZGEÇMİŞ 65

KISALTMALAR

ATSTS : Ağ Tabanlı Saldırı Tespit Sistemleri
STSTS : Sunucu Tabanlı Saldırı Tespit Sistemleri
CPU : Central Processing Unit
KAA : Kablosuz Algılayıcı Ağlar
STS : Saldırı Tespit Sistemleri
YSA : Yapay Sinir Ağları
KB : Kilo Byte
RAM : Random Access Memory
MB : Mega Byte
DOS : Denial Of Service
MAC : Mesia Access Control
IDS : Intrusion Detection System
Bit : Binary Digit

ÇİZELGE LİSTESİ

Sayfa

Çizelge 3.1: *KAA*' da genel saldırı kavramları. 24
Çizelge 4.1: Bilgisayar ile insan beyninin karşılaştırılması. 37
Çizelge 5.1: KDD 99 Cup tekil bağlantı vektörü öznitelikleri 48
Çizelge 5.2: % 10 KDD' 99 saldırı tip ve sayıları 49
Çizelge 5.3: Seçilen 22 adet öznitelik. 51
Çizelge 5.4: Servis öznitelikleri normalizasyonu. 52
Çizelge 5.5: Protokol özniteliği normalizasyonu. 52
Çizelge 5.6: Bayrak özniteliği normalizasyonu. 52
Çizelge 5.7: Gerçekleştirilen test sonuçları 55
Çizelge 5.8: Gerçekleştirilen testlere bazı örnekler. 56

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Hizmet engelleme saldırı tipleri. 7
Şekil 2.2 : Genel *STS* yapısı. 10
Şekil 2.3 : Anormallik tespiti. 11
Şekil 4.1 : Biyolojik sinir hücresinin yapısı. 38
Şekil 4.2 : Yapay sinir hücresinin yapısı. 40
Şekil 4.3 : Heaviside fonksiyonu. 42
Şekil 4.4 : T parametreli fermi fonksiyonu. 43
Şekil 4.5 : Hiperbolik tanjant fonksiyonu. 44
Şekil 5.1 : KDD Cup 99 veri seti örnek bağlantı vektörü. 47
Şekil 5.2 : Akış şeması. 50

SEMBOL LİSTESİ

α	:	Alpha
β	:	Beta
θ	:	Theta eşik değeri
o	:	Nöron çıktıları
i	:	i nöronu
T	:	Fermi fonksiyonuna eklenen T (Temperature) değişkeni
$net_{i,j}$	:	Propogasyon fonksiyonu sonucu bulunan nöronun net girdisi
$w_{i,j}$	:	i nöronundan j nöronuna olan girdi baplantısının ağırlığı
x	:	Nöron Girdileri

DAĞITILMIŞ SİSTEMLERDE SALDIRI TESPİT SİSTEMLERİ

ÖZET

Günümüz iş, eğitim, sosyal yaşam, askeri ve ekonomik alanlarında bilgisayar ağlarının sürekli artan kullanım faaliyetleri, bilgisayar kullanıcı sayısının her geçen gün sınırları zorlayacak şekilde artış göstermesi, siber saldırı ve bu saldırıları tespit edebilmek ve onlara karşı koyabilmek için yapılan araştırmaları sürekli olarak yoğun çalışılan konu başlıkları olarak karşımıza çıkarmaktadır. Bilgisayar sistemlerinin ve eşyanın tabiatı gereği karşımıza çıkan bilgisayar ağları yaklaşımının teknoloji sahnesinde sahip olduğu önem göz ardı edilemez bir gerçektir. Bilgisayar sistemlerinin birbirinden bağımsız ancak birbirleri ile işbirliği ve koordinasyon halinde çalışmaları prensibi altında oluşan dağıtılmış sistemler yaklaşımı kullanım alanları ile teknoloji dünyasının birçok problemine çare olmuş çok sayıda kullanım alanında kendine yer edinmiştir. Adı geçen sistemlere önemli bir örnek olan Kablosuz Algılayıcı Ağlar *(KAA)* da günümüzde önemli bir role sahiptir. Kablosuz Algılayıcı Ağlar *(KAA)* akıllı evler, askeri güvenlik uygulamaları, izleme ve takip uygulamaları gibi birçok alanda kullanılmaktadır. Bu tez kapsamında gerçekleştirilen çalışma dağıtılmış sistemlerin en önemli örneklerinden biri olan *KAA* yaklaşımı üzerinden oluşturup test edilmiştir.

Siber saldırı, bir bilgisayar sistemini veya bir ağ yapısını ele geçirmek, işlevselliğini azaltmak veya hiç işlemez hale getirmek, sistemi veya kullanıcılarını yanıltarak olması gereken çizginin dışına çıkarmak için gerçekleştirilen ve sistem içinden veya dışından ayrıca bilinçli veya bilinçsiz olarak yapılan her türlü zararlı aktiviteler olarak tanımlanmaktadır.Siber saldırı tipleri her geçen gün kendini yenilemekte ve çeşitliliğini arttırmaktadır.Bu nedenle güçlü, kendini yenileyebilen, güncel bir saldırı tespit sisteminin gerçeklenmesi ve üzerinde sürekli yeni yaklaşımlarla araştırmalar ve geliştirmeler yapılması çok önemlidir.

Literatüre baktığımızda farklı saldırı tespit sistemlerinin kullanıldığı görülmektedir. Bunlar Etmen tabanlı sistemler, dağıtılmış saldırı tespit sistemleri, evrimsel yaklaşımlar, ağ tabanlı sistemler, davranış tabanlı sistemler vb. anabaşlıklar üzerinde yoğunlaşmaktadır. Yapay Sinir ağları da bu alanda kullanılabilen yaklaşımlardan biridir. *KAA* sahip oldukları kısıtlar ile kısıtsız veya kablolu ağ yapılarından farklı özellikler göstermektedir. Bu özellikler dahilinde bakıldığı zaman *KAA* yapısının güvenlik yaklaşımında da farklılıklar olacağı aşikardır çünkü adı geçen kısıtlar *KAA* için farklı siber tehdit mekanizmalarına neden olmaktadır. *KAA* üzerinde gerçekleştirilecek olan saldırı tespiti yaklaşımının *KAA*' ya ait olan kısıtları dikkate alması ve bu kısıtlar çerçevesinde verimliliği yakalaması gerekmektedir. Dikkat edilmesi kısıtlardan literatürde en çok karşılaşılanlar enerji, işlemci ve depolama alanı kısıtlarıdır. *KAA* kısıtları ayrıntılı olarak ilerleyen bölümlerde anlatılacak hangi kısıta nasıl çözüm bulunduğu ve verimliliğin arttırılması için nelerin gerçekleştirildiği

açıklanacaktır. Çalışma kapsamında *YSA* yaklaşımının seçilmesinin en temel nedeni de kısıtları en iyi şekilde ortadan kaldırabilme imkanını sunmasıdır. Bu kapsamda elde edilen test sonuçları da son bölümde kendine yer bulmuş durumdadır.

Günümüzün harekat ortamları siber savaş üzerine geçmektedir. Bazı önlemler saldırı miktarlarını azaltsada tamamen engelleyememektedir. Bu nedenle bir ağın etkili bir saldırı tespit sistemi ile korunması önemlidir. Yapay Sinir ağları bu noktada önemli bir araç olarak karşımıza çıkmaktadır.Bu nedenle bu çalışmamızda yapay sinir ağı ile eğitilmiş bir Saldırı Tespit Sistemi tasarlanması amaçlanmıştır. Geliştirilen sistem KDD99 verileri ile test edilmiştir. KDD 99 Cup veri seti DARPA'98 *STS* hesaplama programından elde edilen veriler üzerine inşa edilmiştir. DARPA'98 herbiri yaklaşık 100 byte olan yaklaşık beş milyon bağlantı kaydıiçeren yaklaşık 4 gigabyte boyutunda 7 haftalık ağ trafiğine ait sıkıştırılmış binary tcpdump verilerinden oluşmaktadır. Söz konusu veri setinin kullanılmasının ana nedeni literatürde en çok kullanılan veri seti olması nedeniyle elde edilen test sonuçlarının yorumlanması kapsamında kolaylık sağlamasıdır. ve önerilen sistemin oldukça yüksek başarı gösterdiği tespit edilmiştir. Bu çalışmada yapay sinir ağları yaklaşımı ile bir saldırı tespit sistemi gerçeklenmesi amaçlanmış ve önerilen sistem altyapısı üzerinde gerçekleştirilen testlerin sonuçları gösterilmiştir. *YSA* yapısının eğitilmesi ve tst işlemleri *MATLAB* üzerinde gerçekleştirilmiş ve elde edilen test sonuçları beşinci bölümde sergilenerek yorumlanmıştır.

INTRUSION DETECTION SYSTEMS FOR DISTRIBUTED SYSTEMS

SUMMARY

Using rates of computer systems, abnormal increasing of number of computer users on social and education life, military and economic areas cause new cyber attack types and techniques so always there are new searching studies to resist new cyber attacks. It is very important that computer and network technologies have a big area in human life. Distributed system is a computer system community and this community includes independent computers working together cooperatively. As a *Distributed System*, Wireless Sensor Networks *(WSNs)* are very important example for this computer and network technologies. They have a big role new technology applications. Wireless Sensor Networks *(WSNs)* are a large scale network having thousands of tiny devices, sensing andcollecting data from physical environment. These tiny and small sized devices have low processing and storage capacity and low cost. It is easy to design and create a *WSN* because of it's cheap price and so it is used for many different fields by various applications. These areas can be lined up as science (exploring oceans, animals, habitats, wildlife and space.g.), health-care (Monitoring in Mass-Casualty Dis-asters), military (Boomerang Sniper Identifying System, Nuclear Bilogic Chemical Attack Identifying), tracking and monitoring applications (monitoring highway traffic, fire alarm systems and home automation systems), agriculture, transportation and many others. There are a lot of IDS design approaches for *WSNs*. As mentioned above *WSN* is a very good example for distributed systems so in this thesis study in order to achieve proposed system and test it, *WSN* is selected.

Intrusion is an unwanted activity in the network and intrusion detection is an important research and development topic with many applications that influencing confidentiality, integrity, availability. A cyber attack aims that capturing whole computer system or a part of it, reducing functionality of computer system, operating the system wrongly by feinting the system or user of it and it is described as activities that being performed by internal and external users or willingly and unwillingly. In current time it is very important that studying for new detection approaches because of incredible evolution of cyber attacks.

In literature there are a lot of approaches for intrusion detection. Such as neural network based, data mining based, mobile agent based, rule based, game theory based, statistical based, genetic algorithm based. In this working, artificial neural network based approach is selected to design an *IDS* to serve a smart system having learn ability. *WSN* has some constraints such as energy, processing and storage. All constraints of *WSN* is described in following section detailed.It is described above that WSNs have limited resources so implementing on a sensor node an anomaly based or a

signature-based IDS is not effective for WSNs. And an additional process consumes energy of sensor. Therefore, as an alternative solution, it is aimed that implementing a neural network based intrusion detection system for wireless sensor networks. In WSN, security threats are more different from wired and non-energy constrained wireless networks. These differences are caused from typical properties of WSN. Energy is the most important constraint for WSN and in addition to three components of security (confidentiality, integrity, and availability), there is a new basic aspect that is energy. To train and test the neural net-work KDD' 99 Cup data-set is selected. KDD'99 Cup data set is adopted for this study because it is widely used intrusion detection data set and so facility of comparison this study' s results with different studies is achieved. KDD'99 Cup data set is created by extracting some features (ip number, port number, initial date) from DARPA 98 and it has about 4.900.000 data vector. KDD' 99 Cup data set includes 80 % attack and 20% normal data.

The aim of IDS implemented for WSN must be isolate the malicious, physical damaged or abnormal node from the network. It is main that IDS for WSN concentrates to find abnormal, not working or working maliciously node. For WSN security, another important aspect is wireless transmission and physical security. Firstly, wireless transmissions can be attacked by signal jamming and eavesdropping. The mechanism implemented for IDS must evaluate the signal jamming and eavesdropping due to it is so important to provide integrity, confidentially, availability of created security system of WSN. Physical security of nodes is important and noticing the damage to any node is one of the goals of IDS. This goal provides integrity and routing of data throughout the network successfully and securely. While serving these aspect low-latency communications must be taken into. It is clear that intrusion detection is so important for a good security policy. There are two main approach for security management these approaches are prevention-based and detection-based. In any security plan, if intrusion prevention (encryption, authorization, authentication) named as the first line of security is passed by attackers, as a second line of defence, intrusion detection comes into prominence. Intrusion detection provides a deterrence for intruder and serves an alarm mechanism for a computer system or a network to manage security plan successfully. An intrusion-detection system (IDS) can be defined as software or hardware tools that monitoring network to detect internal or external cyber attacks. An Intrusion Detection System can observe and investigate system and user activities, recognize patterns of known attacks, identify abnormal network activity. General definition of IDS is about intrusions to network but for WSN it can be added that physical damages to sensor devices. Identifying sensor damage is important in order to serve fault tolerance and reliability.

Anymore, it is more important that protecting, processing, hiding and moving the data because of incredible progressing of information technologies. Spy-wares, mobile threats, growing up of number of attack and attack types have increased importance of cyber security. Because of these increasing, a lot of approach is used to serve a healthy system security. As a information system, WSNs need a protection mechanism to resist against cyber attacks. This counteractive mechanism can be defined as two lined approach. The first line is prevention based approach (encryption, authorization, authentication) and the second line is detection based approach (Intrusion detection). If any attacker passes the first line and then the second line tries to find whether there

is any intrusion or not. Intrusion detection system is a software or hardware that is an alarm component of any network warning the system administrator against unwanted and unauthorized movements. WSNs have some differences and constraints so their IDS approaches are different from wired and non-energy constraint networks. For this study all processes is achieved by ”Matlab nftool”. Test results are got from Matlab all process steps is explained detailed.

1. GİRİŞ

Kablosuz Algılayıcı Ağlar *(KAA)* çok geniş ölçekli ve on binlerce küçük cihaza sahip, bulunduğu ortamdaki verileri takip eden ve toplayan ağ yapılarıdır. Söz konusu küçük cihazlar kısıtlı işlemci, depolama ve enerji yeteneklerine sahiptir. Maliyetleri çok düşük olduğundan dolayı birçok uygulamada tercih edilirler. Bir *KAA* dizayn etmek oldukça kolay ve masrafsızdır. Bu uygulama alanları şu şekilde sıralanabilir; bilim (okyanus keşifleri, hayvanlar, doğa, vahşi yaşam ve uzay), sağlık (Kas erimesi hastalıklarını izleme), askeri (Bumerang keskin nişancı tanımlama sistemi, KBRN tespit uygulamaları), takip ve izleme uygulamaları (hava yolu trafiğini izleme, yangın alarm sistemi, ev otomasyonu sistemi), tarım, taşımacılık ve daha birçoğu. Tüm bilişim sistemlerinde olduğu gibi *KAA*' ların da güvenlik problemleri bulunmaktadır. Söz konusu güvenlik problemlerini tespit edebilmek önmeli bir konu başlığı olarak karşımıza çıkmaktadır.

1.1 Tezin Amacı

Artık günümüzde bilgi teknolojilerindeki inanılmaz gelişmeler ile birlikte bilgiyi koruma, işleme, gizleme ve taşıma çok daha büyük önem arz etmektedir. Virüsler, hareketli tehditler, saldırı tipi ve sayısındaki aşırı artışlar siber güvenliğin önemini günden güne arttırmaktadır. Bu ortaya çıkan güvenlik problemleri, güvenlik sağlayıcı yaklaşımların sayısını da arttırmaktadır. Bir bilgi sistemi olarak *KAA* da siber güvenlik mekanizmasına ihtiyaç duymaktadır. Söz konusu güvenlik mekanizması iki basamaklı bir şekilde tanımlanabilir. İlk basamak saldırıdan korunma tabanlı (şifreleme, yetkilendirme, parolalaşma) yaklaşımlardır ve ikinci basamak ise tespit tabanlı (saldırı tespiti) yaklaşımlardır. Herhengi bir saldırgan birinci basamağı aşmayı başarırsa ikinci basamak söz konusu saldırganı, niyetini, verdiği zararları veya saldırının sonuçlanıp sonuçlanmadığını tespit etmeye çalışır. STS' ler sistem yöneticilerini ağ üzerindeki yetkisiz ve istenmeyen her türlü eyleme karşı uyaran

bir alarm niteliğindedir. *KAA* bazı kısıtlara ve farklılıklara sahiptir bu nedenle *KAA* için gerçekleştirilen Saldırı Tespit Sistemleri *(STS)* de normal ağ yapısındakilere göre bazı farklılıklar göstermektedir [1] [2]. *KAA* için bir çok Saldırı Tespit Sistemi *STS* gerçekleştirim yaklaşımı bulunmaktadır. Örneğin; yapay sinir ağı tabanlı, veri madenciliği tabanlı, mobil ajan tabanlı, kural tabanlı, oyun teorisi tabanlı, istatistik tabanlı, genetik algoritma tabanlı. Bu çalışma için **öğrenme yeteneği olan geliştirilebilir bir sistem** oluşturmak için **yapay sinir ağı tabanlı yaklaşım** seçilmiş ve kullanılmıştır. Sinir ağını eğitmek ve test etmek için KDD Cup 99 veri seti kullanılmış ve işlemler "Matlab nftool" üzerinde gerçekleştirilmiştir. Sistemi tasarlama aşamasında *KAA* yapısının sahip olduğu kısıtlar göz önünde bulundurulmuş ve bu kısıtları karşılayabilmek hedefiyle çalışmaya yön verilmiştir. Unutulmamalıdır ki *KAA* kablolu veya kısıtsız ağ yapılarına göre önemli farklılıklar göstermektedir bu nedenle de ihtiyaçları farklı rotalar izlemektedir. *KAA* boyutları oldukça küçük ve sayıları onlardan onbinlere dayanan algılayıcı düğümlerden oluşan ağ yapılarıdır ve kullanım amaçlarında göre belirli bir düzene uymadan hata hoşgörülü bir şekilde ve işbirliği ile çalışmaları istenebilir. Bu küçük algılayıcı düğümler sahip oldukları enerji, işlemci ve depolama yeteneklerinin sınırlı olması nedeniyle oluşturulacak güvenlik politikaları kapsamında kablolu ve kısıtsız ağ yapılarından keskin bir şekilde ayrılırlar.

2. LİTERATÜR ARAŞTIRMASI

Çalışma konusu kapsamında gerçekleştirilen literatür araştırması, *"A Survey of Intrusion Detection Systems in Wireless Sensor Networks"* başlığı altında *6th Internetional Cenference on Modeling, Simulation and Applied Optimazition*, İstanbul' da sunulmuştur. Tez çalışmasının ana hatları adı geçen yayın ile oluşturulmuştur.

Kablosuz Algılayıcı Ağlar *(KAA)* bulunduğu ortamdan bilgileri toplayan ve bünyesinde çok sayıda ufak cihaz bulunduran ağ yapılarıdır. Bu cihazlar düşük işlemci, hafıza ve enerji kapasitesine sahiptir. Etkili bir KAA oluşturmak önemli bir çalışma alanıdır. Oluşturulan *KAA* esnek, güvenilir, güvenli ve hata toleranslı olmalıdır. *KAA*' lar düşük maliyetli yapılar olduğundan dolayı sağlıktan askeri güvenliğe kadar birçok alanda kullanılmaktadır. Bu alanlara petrol kuyusu güvenliği, sınır güvenliği, tabiatı ve hayvanları izleme sistemleri, yangın alarm sistemleri, ev otomasyon sistemleri, uzay keşif sistemleri ve çok daha fazlası örnek olarak gösterilebilir [3].

Siber saldırı ağ üzerindeki istenmeyen aktiviteler olarak açıklanabilir ve saldırı tespit sistemleri *(STS)* söz konusu aktiviteleri tespit etmeye yarayan sistemlerdir [4]. Çok açıkça görülmektedir ki iyi bir güvenlik politikası büyük önem arz etmektedir. Güvenlik yönetiminde iki önemli yaklaşım bulunmaktadır. Bunlar; koruma tabnlı ve tespit tabanlı olarak sıralanabilir. Herhangi bir güvenlik planında koruma tabanlı yaklaşım ilk hat ise tespit tabanlı yaklaşım ikinci bir hat olarak karşımıza çıkar. *STS* saldırganlar için önemli bir caydırıcılık etkenidir. *STS*' ler ağ üzerindeki saldırıları tespit etmeye yarayan yazılımlar veya donanımlar olarak adlandırılır. *STS*' ler bulundukları sistemi gözlemler ve kullanıcı aktivitelerini bilinen saldırılara ait imzalarla karşılaştırır veya anormal ddurumalrı takip eder. *STS*' nin genel tanımına *KAA* ile birlikte cihazlara olabilecek fiziksel zararlar ve cihazların enerjilerinin tükenmesini de ekleyebiliriz. Fiziksel zararların tespiti güvenilir ve hata hoşgörülü sistemler için çok önemlidir [5].

Kablolu ağlar için *STS*' ler istenmeyen aktivitelerin (dahili veya harici) tespitinde kullanılırlar ve geniş ölçekli bir güvenlik hizmeti sunmayı hedefelrler. Bu mekanizma yasaklanmış ağ aktivitelerini ve yetkisiz, anormal sistem hareketlerini tespit etmeyi amaçlar.Kablolu ağlar için çeşitli *STS* teknik ve yaklaşımı vardır fakat bu teknikleri birçoğu *KAA*' lar için uygun değildir [6].

KAA güvenliği için bir diğer önemli yaklaşım şudur ki kablosuz taşıma ve fiziksel güvenlik. İlk etapta kablosuz sinyalizasyon karıştırmaya maruz kalabilir.*KAA* için gerçekleştirilecek bir *STS* sinyal karıştırma saldırılarını da hesaba katarak tasarlanmalıdır. Algılayıcı cihazların fiziksel güvenliği de güvenlik planı için önemli bir konudur ve hedeflerden biridir. Bu hedef sistemin bütünlüğünü sağlamaktadır [7] [8].

2.1 *KAA*' larda Siber Saldırılar

KAA' larda güvenlik tehditleri kablolu ağlara göre farklıdır. Bu farklılıklar *KAA*' nın kendine özgü karakteristiğinden kaynaklanmaktadır. Enerji en önemli kısıttır bu nedenle enerji tüketimi yeni bir bakış açısı kazandırmaktadır. Aşağıda *KAA*' larda bulunan dört temel yaklaşım kısaca açıklanmaktadır.

- **Gizlilik :**

  Siber Güvenlik paradigmasında gizlilik en çok bilinen ve dikkat edilen bileşendir. Kullanıcılar gönderdikleri verilerin üçüncü kişiler tarafından görünmesini istemezler. KAA iletişiminde ağdan akan verinin dinlenmesi oldukça kolaydır. Gizlilik için altın kelime "bilgi değerlidir" sözcük bütünüdür ve kullanıcılar bilgilerini yetkisiz, istenmeyen kişilerin eline geçmesini istemezler. Gizliliği korumak için bilgi şifrelenmelidir. Şifreleme işlemleri simetrik veya asimetrik anahtarlama yöntemleri ile yapılabilir. Asimetrik şifreleme yöntemleri daha güçlü pozisyondadır. Fakat enerji tüketimi açısından elverişli değildir. *KAA* kullanıcısı simetrik şifrelemeyi tercih ederse anahtarını çok dikkatli şekilde saklamalıdır. Gizliliği sağlamaya yardımcı olmak için *STS* kullanıcıları ikaz edebilir [9].

- **Bütünlük :**

  Bütünlük ağ üzerinde gönderilen verini değiştirilmesine engel olmak ve değiştirilmediğine emin olmaktır. Bilgi sadece doğru ise bir değere sahiptir. *KAA*' lar bulundukları fiziksel ortamı gözlemler ve bu ortamdan elde ettikleri veriyi toplarlar. Toplanan veriler işe yaraması için değiştirilmediğinden emin olmak gerekir. Kablolu ve enerji kısıtı olmayan ağ yapılarında bütünlik dijital imzalar ile sağlanmaktadır fakat bu yaklaşım *KAA*' lar için uygun değildir çünkü bir cihazdan diğer cihaza gönderilen verilere fazladan bitlerin eklenmesine neden olur. Bir diğer olumsuzluk ise fazladan hesaplama kaynağına ihtiyaç duyulmasına neden olur. Gönderilecek veride yaşanan artış ve fazladan hesaplama yeteneği gereksinimi nedeniyle bu yöntem uygun olmaz fakat enerji etkili bir *STS* ile bütünlük sağlanabilir.

- **Erişilebilirlik :**

  Eğer ki saldırı çeşitliliği incelenirse saldırıların sadece veriye ulaşmaya çalışmadığını, veriye erişimi engellenmesinin de bir saldırı türü olduğu görülebilir. Bu tür saldırışarda amaç ağı işlemez duruma getirmektir. Bilgi tabi ki değerlidir fakat doğruysa (bütünlük) ve zamanında elde edilebiliyorsa (erişilebilirlik). Saldırganlar *KAA*' lara hiçbir çıkışı olmayan yalancı rotalara yönlendirme suretiyle saldırabilir.

- **Enerji :**

  KAA enerji kısıtına sahiptir ve bu durum tüm güvenlik planlarını etkilemektedir. *KAA*algılayıcı kısıtlı hesaplama ve depolama alanlarına sahiptir. *KAA*' lar bulundukları ortamı gözlemler ve ilgili verileri toplayarak değerlendirip kullanıcısına bilgi verir. Kullanıcılar sistemin sahip olduğu cihazların enerji ömürlerinin kısa olmasını doğal olarak istemezler. Bu nedenler gizlilik, bütünlük ve erişilebilirlik yaklaşımlarına ek olarak enerji kısıtı da *KAA*' larda güvenlik temelleri planında yerini almış duurumdadır. Gerekli güvenlik sistemi oluşturulurken enerji kısıtı mutlaka hesaba katılmalıdır.

Geleneksel ağ güvenlik yaklaşımına ek olarak *KAA* güvenliği farklı bileşenler ile ortaya çıkar. Bu farklı bileşenler farkklı tekniklerin ortaya çıkmasında neden olur.

Farklı tehditler ve farklı karşı önlemler. Buradan sonra *KAA*' lar için ortaya çıkan siber saldırı tipleri üzerinde durulacaktır.

2.1.1 Hizmet engelleme saldırıları

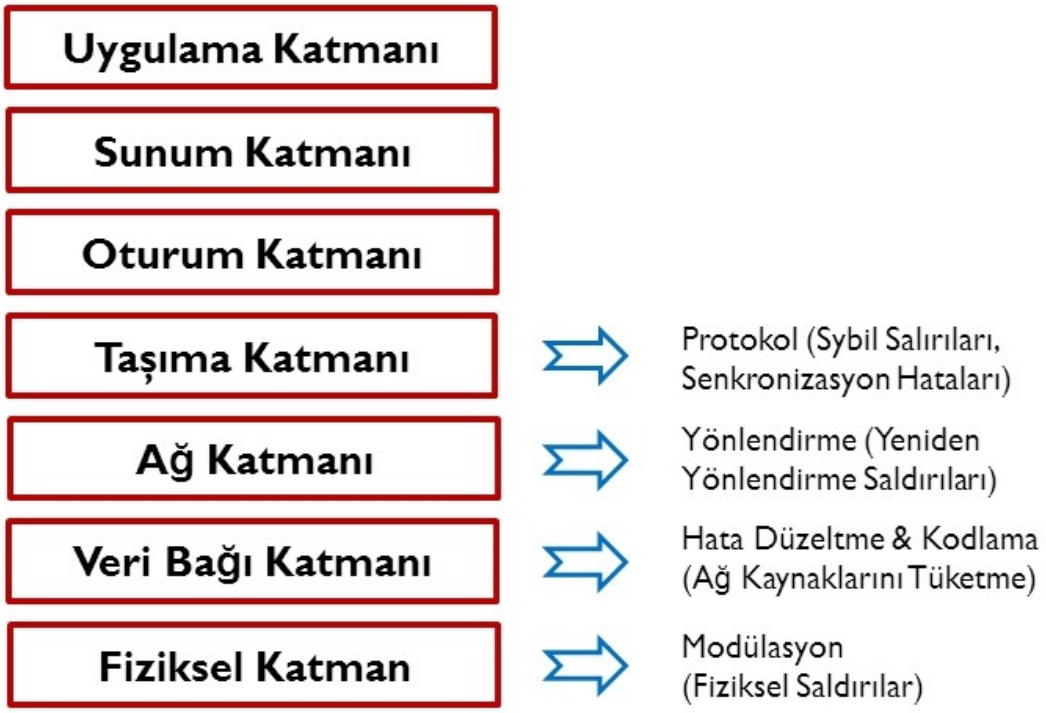
Hizmet engelleme saldırılarının temel amacı sistemi cevap veremez hale getirmektir. KAA' lar özellikleri gereği farklı hizmet engelleme saldırılarına konu olabilir [2].

- Kısıtlı kaynakları kullanılamaz hale getirmek
- Oluşturulan *KAA* konfigurasyonu gizli kalmalı çünkü saldırganlar bunu kullanabilirler.
- *KAA*' larda fiziksel zarrara çok dikkat edilmelidir. Cihazlar açık ve geniş bir alana savunmasız olarak dağılmış olabilirler. Bu nedenle fiziksel zararlara açık hedef haline gelebilirler.

KAA' lar doğası gereği saldırganlar sistemin açıklarından ve kısıtlarında faydalanmak isteyecektir. Örneğin ilgili donanım fazla yoğunlukta çabucak cevap veremez hale gelirler. Fiziksel katmana saldırı yaklaşımında ise saldırganlar kablosuz haberleşmeyi karıştırmaya çalışabilirler. Karıştırmaya önlem amacıyla haritalama protokolu kullanılabilir. Hangi bilginin nereye ne zaman gideceği önceden belli olursa anormal durumlar çabucak tespit edilebilir bu sayede durumsal farkındalık artar. Ortaya çıkacak mesaj karmaşası komşu düğümler arasında bir anormalliğin meydana geldiğine dair karar verme yeteneği sağlayabilir. Fiziksel zarara karşılık ise sensörlerin kamufile edilmesi önlem olarak düşünülebilir. Şekil 2.1 üzerinde katmanlar arası hizmet engelleme saldırı tipleri özet olarak gösterilmiştir [10].

2.1.2 Yeniden yönlendirme

Yeniden yönlendirme saldırılarında *KAA* üzerinde bulunan olması gereke bilgi akış rotası olumsuz yönde değiştirilir. Bu saldırının amacı verinin yanlış rota üzerinden koşmasını sağlamaktır. Şekil 2.1 üzerinde yeniden yönlendirme saldırıları ağ katmanı saldırıları olarak işaret edilmiştir. Ağ üzerinde yeniden yönlendirme olup olmadığını tespit etmek çok önemlidir. *KAA* üzerinde belirli hesaplamalar ve veri



Şekil 2.1: Hizmet engelleme saldırı tipleri.

akış verimliliğini hesaplayarak herhangi bir yeniden yönlendirme olup olmadığı anlaşılabilir. Tespit yöntemleri şu şekilde sıralanabilir :

- Hash edilmiş veri paketlerini kullanmak. Fazladan enerji tüketmez
- Bir parolalaşma mekanizması kullanmak. Bu yöntem düşük performansa neden olur
- Çoklu hoplamalı gizli bir rota kullanmak. Bu yöntem zararlı düğümleri tespiti zorlaştırır.

2.1.3 Seçmeli iletim

Şekil 2.1 üzerinden görüleceği üzere seçmeli iletim saldırısı da bir ağ katmanı saldırısıdır. Bu saldırıda zararlı düğümler normal bir düğümmüş gibi davranırlar ve paketleri iletirler fakat bu işlem esnasında bazı paketleri yok ederler. Bu saldırıyı tespit etmek gerçekten çok zordur [11]. Bu saldırıya karşı önlemler şu şekildedir:

- Onay tabanlı tespit
- Komşuluk bilgisini kullanan tespit

- Çoklu data akışını kullanmak

Bu karşı önlemleri gerçeklemek için belirli gereksinimler bulunmaktadır. Temel olarak başarılı bir güvenlik yönetim mekanizması oluşturmak zorunludur. Düğümler arası gizli ve güvenli bir iletişim oluşturulmalıdır. Verim oranı ve başarılı paket oranları sürekli hesaplanmalıdır.

2.1.4 Karadelik saldırıları

Bu saldırı tipinde saldırgan ağ içinde bir düğümü ele geçirir ve bu düğümü saldırı gerçekleştirmek için kullanır [12] [13]. Ağı sürekli denler ve ana düğüme ulaşmak için en kısa rotayı belirlemeye çalışır. İlgili düğüm bunu başarırsa saldırı gerçekleştirmek için hazır demektir. Bu tip saldırılar veri bağı katmanı saldırısı olarak değerlendirilir [13]. Hedefine ulaşan saldırgan artık ne isterse yapabilir. Kendisine gelen tüm paketleri yok edebilir, değiştirebilir, kendisi paket oluşturabilir. Karşı önlem olarak ise düğümlerin işlemci verimliliklerini incelemek gibi yöntemler bulunmaktadır [12] [13].

2.1.5 Sybil saldırıları

Bahse konu saldırı biçimlerinde ise iki durum da söz konusu değildir. Bir adım gerideki düğüm açısından iletişim gayet normal olarak devam etmektedir. Saldırgan düğümün bir adım ilerisindeki düğüm açısından ise herhangi bir iletişim olmadığı için her şey normaldir. Bu durumda sadece saldırgan düğüm problemin farkındadır ve onunda bu durumu rapor etmek gibi bir niyeti olmayacağı açıktır. İlk akla gelen iki seçenekten birisi, sonraki düğüm seçiminde rastlantısallığı devreye alarak belirli oranda paketin saldırgan düğüme doğru gitmemesini garanti altına almak olabilir ki, bu durum şans faktörünü ön plana çıkaracağından yeterli olmayacaktır. Diğer seçenek ise, iletişimin "sonraki düğüm" olarak seçilmeyen düğümler tarafından sessizce kontrol edilmesi olabilir. Buna benzer çözümler Kara Delik saldırılarına karşı teklif edilmiştir [14] [15] [16].

2.1.6 Solucan deliği

Solucan deliği (wormhole), birbirinden uzakta konuşlanmış iki saldırgan düğümün aralarında oluşturdukları özel bir iletişim yolu ile (*VPN* benzeri) birine gelen

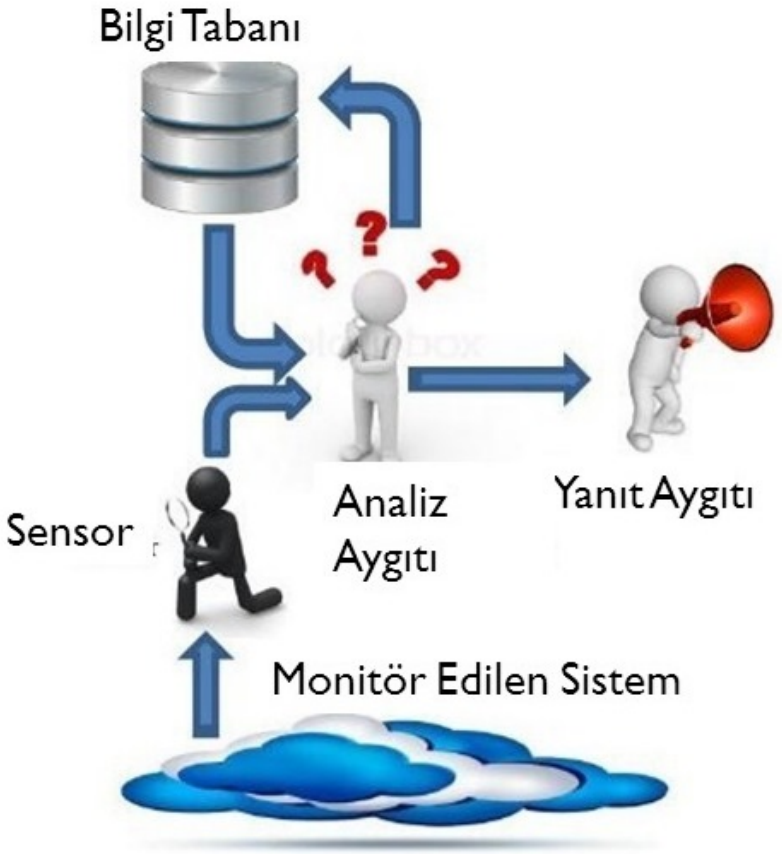
veri paketlerinin doğrudan diğerine aktarılması yoluyla yönlendirmenin bozulmaya çalışıldığı saldırı biçimidir . Ağ üzerindeki diğer düğümler tarafından iki normal düğüm arasındaki normal bir iletişim olarak algılanacağı için tespit edilmesi oldukça zor bir saldırı türüdür. Ancak iki ayrı konumdaki iki ayrı düğüm tarafından kendine özgü kuralları olan bir iletişim gerektirdiğinden gerçekleştirilmesi de aynı oranda güç olan bir saldırı biçimi olduğu değerlendirilmektedir. Ayrıca, ilk saldırgan düğümün komşuları açısından incelendiğinde kara delik saldırılarına çok benzer şekilde iletilen paketlerin aktarılmadığına yönelik belirtiler vereceği için, aynı grup içinde ele alınmalarının uygun olacağı değerlendirilmektedir. Yönlendirme istek ve/veya cevap paketlerinin de solucan deliği üzerinden aktarılması durumunda uyarlamalı yönlendirme kullanan protokoller çok uzak noktalarda düşük sıra numaralı istek paketlerinin yayınlanmasına sebep olabilir [17].

2.1.7 Hello baskını saldırıları

Saldırgan güçlü çıkış gücüne sahip bir verici vasıtasıyla ağa *"hello"* mesajı gönderir. Bu mesajı alan düğümler vericinin kendilerine komşu bir düğüm olduğunu düşünür ve yönlerdirme tablolarını buna göre oluştururlar. Ancak bu düğümlerin çıkış gücü yeterli olmadığından sahte düğüme göndermek isteyecekleri mesajlar kaybolacaktır. Saldırganın baz istasyonuna kaliteli bir yolunun bulunduğunu ilan etmesi ile tüm düğümler mesajlarını bu düğüme iletmeye çalışacak ve böylece ağ trafiği kesilecektir [18].

2.2 Genel Saldırı Tespit Sistemi Yapısı

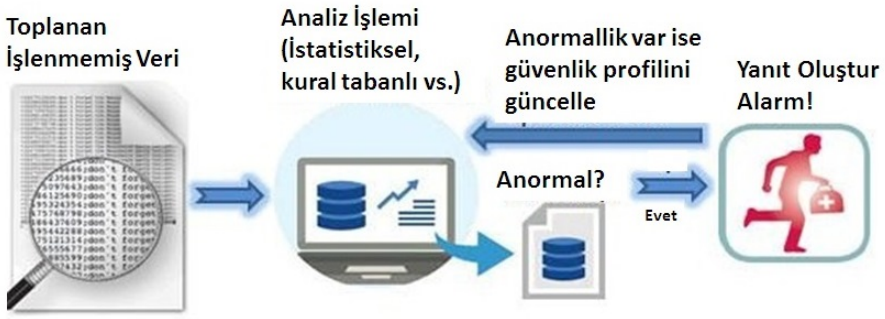
STS' nin öncelikli ve ana hedefi sisteme içeriden veya dışarıdan olabilecek herhangi bir saldırının tespit edilmesidir. Saldırının tespit edilmesi hususu ne kadar ba¸sarılı olursa sisteme karşı yapılabilecek herhangi bir saldırı motivasyonuna karşı caydırıcılık oluşacak ve saldırılar için bir koruma mekanizması oluşacaktır. Bütün bunların yanında oluşabilecek saldırı durumunda ise kanıt toplamak ve saldırının nereden, nasıl, kim tarafından yapıldığının tespit edilmesi de *STS* için önemli bir amaçtır. Saldırı Tespit Sistemi yapısı Şekil 2.2 [19] üzerinden incelendiğinde; bilgi kaynağı, analiz,



Şekil 2.2: Genel *STS* yapısı.

yanıt olmak üzere 3 ana bileşenden oluştuğu görülmektedir. Bu bileşenlerin ayrıntısına inmek gerekirse [20]:

- **Sensör :** Monitör edilen sistemden analiz edilecek verilerin toplanması ile sorumludur
- **Analiz Aygıtı :** Sensörden toplanan verileri, monitör edilen sisteme yapılan herhangi bir saldırı olup olmadığını tespit etmek için analiz eder
- **Bilgi Tabanı :** Sensörler tarafından toplanan bilgileri analiz edilmemi¸s haliyle (yapılmış saldırılar ve saldırılara ait imzalar, filtrelenmiş veri vs.) içerir. Bilgi Tabanı bizim için bir nevi istihbarat bilgisidir ve olabilecek saldırılara karşı ön hazırlık yapmamızı sağlar. Bilgiler genellikle ağ ve güvenlik uzmanlarından temin edilir.



Şekil 2.3: Anormallik tespiti.

- **Yanıt Aygıtı :** Herhangi bir saldırı tespit edildiğinde gerekli aksiyonu yerine getirir. Yanıt otomatik olarak (aktif) veya kullanıcı tarafından (inaktif) olarak gerçekleşebilir

Saldırı Tespit Sistemini genel olarak sistemi izleme yaklaşımına göre ikiye ayırabiliriz: **Ag Tabanlı ve Bilgisayar Tabanlı *STS***. Bu yakla¸sımlardan hangisinin seçilecegi sistemin kullanılacagı organizasyonun yapısına, kullanıcı profiline, ulaşılabilir kaynaklara, organizasyonun sahip oldugu risk oranına göre degişiklik gösterebilir. **Bilgisayar Tabanlı *STS*** belirli bir bilgisayar üzerinde bulunur ve bu bilgisayar üzerinde meydana gelebilecek saldırıları gözlemler. Ağ Tabanlı *STS*, ağ trafigini izleyen dağıtılmış bir sistem üzerinde bulunur. Diğer bilgisayarlar için ag katmanındaki verileri toplar. Ağ Tabanlı *STS* sensörleri network üzerinde herhangi bir yerde olabilir. Ağ Tabanlı ve Sunucu Tabanlı *STS* harmanlanarak oluşturulan yaklaşıma ise Karma Tabanlı *STS* adı verilmektedir [21] [19].

Saldırıları tespit etmek amacıyla yapılan analizler için iki temel yaklaşım bulunmaktadır. Bu yaklaşımlar *"Kötüye Kullanım Tespiti"* ve *"Anormallik Tespiti"* isimleriyle adlandırılmışlardır

2.2.1 Anormallik tespiti

Anormallik Tespitinde kullanıcılar, sunucular ve ağ bağlantılarının normal davranış sergileyen profillerinin analiz edilmesine dayanan kısacası normal şartları inceleyen bir yaklaşımdır. Bu yaklaşım potansiyel tehlikeleri tespit etme oranının yüksekligi ile dikkat çekmesine rağmen çok sayıda gereksiz alarm vermesi olumsuz tarafı olarak gösterilmektedir [21] [19]. Çalışma prensibi Şekil 2.3 üzerinden incelenebilir.

2.2.2 Kötüye kullanım tespiti

Kötüye Kullanım Tespiti bilinen saldırılara ve sitemin zayıf noktalarına ait oluşturulmuş geniş bir bilgi tabanı üzerine kurulan bir yapıdır. Kötüye Kullanım Tespiti yaklaşımı sahip oldugu bilgi tabanı ile bilinen/olası saldırıların tespitini veya sistemin zayıflıklarının bilinmesi ile de bu zayıflıkları kullanması muhtemel saldırıların tespitini amaçlamaktadır. Kötüye Kullanım Tespiti yaklaşımı bilinen saldırıların tespitinde her ne kadar oldukça etkili olsa da yeni ve hakkında bilgisi bulunmayan saldırılar konusunda zayıf kalmaktadır [21].

2.3 *KAA*' larda *STS* Yaklaşımları

KAA' ların sahip oldukları karakteristik özellikleri nedeniyle kablolu ağlardan farklı tehdit kavramlarının olduğundan daha önce bahsedilmişti. Bu nedenle *KAA* için tasarlanan *STS* yapılarında farklı yaklaşımlar kabul edilmektedir. Bu bölümde *KAA* için gerçekleştirilen belirli *STS* çalışmaları üzerinde durulmuştur [2]. Her ne kadar farklı yaklaşımlar ortaya çıksa da gerçekleştirilen *STS* yapısının kaynak noktası klasik yapıdır (Anormallik Tespiti, Kötüye Kullanım Tespiti). Kablolu ağlara göre farklılıklar bu bölümde ortaya çıkarılmıştır.

Sınıflandırma saldırı tipine, saldırgan tipine, tespit tekniğine, elde edilen verinin kaynağına, datanın analiz edildiği yere göre yapılmıştır ve literatürde de bu şekilde açıklanmıştır. Bir ağ yapısında saldırgan tipleri iki gruba ayrılmıştır. Dahili saldırganlar ve harici saldırganlar. Saldırı tipine göre verinin çalınması, yanlış veri üretilmesi, sisteme erişimin engellenmesi, enerji tüketiminin arttırılması. Tespit metedolojisine göre yukarıda anlatıldığı üzere anormallik tespiti, kötüye kullanım tespiti ve literatüre göre ikisinin birleşimi olarak gösterilen karma yöntem. Herhangi bir durumda kural ihlali olursa sistem bir anormallik olduğuna karar verir. Verinin elde edildiği noktaya göre ise ağ tabanlı, sunucu tabanlı ve karma tabanlı olarak gruplandırılır. Verinin işlendiği noktaya göre ise merkezi ve dağıtılmış yaklaşımlar söz konusudur. Nasıl bir yaklaşım belirleneceği sistem gerekliliklerine göre değişebilir [22]. Son yıllarda geliştirilen bazı çalışmalar aşağıda belirtilmiştir.

2.3.1 *KAA* için anormallik tespiti

KAA yapısında ortaya çıkan anormallikler ağ anormallikleri, düğüm anormallikleri, veri anormallikleri ve diğer anormallikler olarak gruplandırılabilir [23].

- Ağ anormallikleri KAA üzerinde oluşan bağlantı problemleri ile ortaya çıkmaktadır. Aniden artan veya azalan sinyal kalitesi anormallik olup olmadığını anlaşılmasına yardımcı olur. Sinyal kaybı, Kesik kesik sinyal edinimi, döngü tespiti ve genel yaylımlar anormallik işaretleridir.
- Düğüm Anormallikleri yazılım ve donanım problemleridir.
- Veri anormallikleri elde edilen verilerde görülen bozulmalar paket boyutlarındaki değişikliklerdir.
- Diğer anormallikler yukarıda sıralanan başlıklar altına doldurulamayan karşılaşılabilecek anormalliklerdir.

KAA anormalliklerine ek olarak, *KAA* anormalliklerini tespit etme yaklaşımları da önemlidir. Bu yaklaşımlar *KAA* için *STS* tasarımında kullanılırlar ve birbirleri ile birleştirilebilirler. İstatistiksel tabanlı, yapay sinir ağı, makine öğrenmesi tabanlı, veri madenciliği tabanlı yaklaşımlar sıralanabilir.

***KAA* İçin Tasarlanmış Oyun Teorisi Tabanlı *STS* - 2014,** çalışmasında sezgisel yaklaşımlar yeine analtik eğilimlerin kullanılması anlatılmıştır. Oyun teorisinin bileşenleri ve kuralları *KAA*' nın öznitelikleri ile oluşturulmuştur [24].

***KAA* yapısında Anormallik tespiti ve yer belirlemesi çalışması - 2013,** incelendiğinde yazarın yüksek bant genişliğinde tasarlanan *KAA*' lara ait *STS* yapılarının incelendiği gözlemlenmiştir [25].

***KAA* için tasarlanan bir *STS* yapısında Veri Madenciliği Yaklaşımının Benimsenmesi - 2013,** konulu çalışmada ise hem anormallik tespiti hem de kötüye kullanım tespiti yapan bir sistem önerilmiş ve ilgili çalışmanın sonuçları gösterilmiştir [26].

2.3.2 *KAA* için kötüye kullanım tespiti

Aynı zamanda imza tabanlı olarak bilinmektedir. Bilinen saldırıların tespitinde en etkili yöntemdir. İlk defa karşılaştığı saldırılar karşısında sıkıntı yaşamaktadır. Tüm saldırıların imzasını taşımak ve bu şekilde saldırı tespiti yapmak özellikle *KAA*' ların kısıtlı kaynakları düşünüldüğünde hiç etkili değildir. Mobil ajan tabanlı yaklaşımlar literatürde görülmüştür [2].

İleri İzlemeli Tabanlı STS ve Seçmeli Algoritma - 2013 isimli çalışmada bir düğümün veri iletirken komşuları ile olan ilişkisi ile anormal düğüm olup olmadığının belirlenmesi amaçlanmıştır [27].

2.3.3 *KAA* için karma tespit yaklaşımı

Bazı saldırı tespit çözümleri benimsedikleri yaklaşımlardan dolay anormallik tespiti veya kötüye kullanım tespiti kavramlarının içine yerleştirilemez. Bu tip yaklaşımlar Karma tespit yöntemi başlığı altında toplanmışlardır. Bu yöntem üç şekilde sınıflandırılmıştır. 1- Dağıtılmış yaklaşım, 2- Önceden Tanımlı Yaklaşım, 3- Karma Sistem Yaklaşımı [2].

Kümelenmiş Bir *KAA* İçin Karma *STS* Tasarımı - 2011, isimli çalışmada yazar anormallik tespiti ve kötüye kullanım tespitini destekçi vektör makinesi ile birleştiren bir yaklaşım kullanmış ve adına karma yaklaşım demiştir [28].

3. KABLOSUZ ALGILAYICI AĞLARDA GÜVENLİK VE SALDIRI TESPİT SİSTEMLERİ

Kablosuz Algılayıcı Ağlar *(KAA)* günümüz dünyasında birçok önemli probleme çözüm olduğu ve bu çözümü ekonomik olarak sağladığı için her geçen gün artan bir kullanım çeşitliliği ve yoğunluğuna sahiptir. Bu yoğunluk kapsamında hem sivil hem de askeri alanlarda uygulanabilir bir yapısı mevcuttur fakat oluşturulan ağ yapısının mutlaka göz önünde bulundurulması ve mevcut ihtiyaçlara göre çözüm için faktör olarak kabul edilmesi gereken bazı kısıtları bulunmaktadır. Bu kısıtların başında depolama, işlemci ve enerji kısıtları gelmektedir. Sahip olduğu farklı özellikler nedeniyle, *KAA* geleneksel güvenlik anlayışının dışında yaklaşımlarla değerlendirilmeli, bu yönde güvenlik çözümleri üretilmelidir. *KAA* için güvenli olmayan iletişim kanalları ve her türlü dış çevresel etkiye açık ortamlarda bulunması savunma mekanizmaları oluşturulmasını daha zor hale getirmektedir. *KAA* için gerekli güvenlik mekanizmalarını incelerken gerekli yaklaşım dört farklı başlık altında toplanabilir. Bu başlıklar *KAA*' ın sahip olduğu kısıtlar ve bu kısıtların güvenliğe etkileri, güvenli bir *KAA* yapısı için ihtiyaç duyulan gereksinimler, saldırılar ve önlemler şeklinde sıralanabilir. Önlemler konusu altında çalışmanın ana teması olan *STS* üzerinde ayrıntılı olarak durulacak ve geleneksel yapısından başlayarak *KAA* için sahip olunan farklılıklardan da bahsedilecektir.

3.1 *KAA*' ın Sahip Olduğu Kısıtlar Ve Bu Kısıtların Güvenlik Mekanizmasına Olan Etkileri

Kablosuz Algılayıcı Ağlar *(KAA)* geleneksel kablolu veya herhangi bir kısıta sahip olmayan ağ yapılarından ayrı olarak değerlendirilmeli ve bu yaklaşım ışığında uygulamalar geliştirilmelidir. Geliştirilen uygulamalar kapsamında oluşturulacak ağ güvenliği mekanizmalarında *KAA*' ın sahip olduğu kısıtlar mutlaka hesapa katılmalıdır. Unutulmamalıdır ki; kısıtları ortadan kaldırırken ve etkili bir güvenlik mekanizması tasarlarken geliştirilen uygulamanın ihtiyaçları değerlendirilmeli ve bu ihtiyaçlara

hangi kısıtların etki ettiği planlanmalıdır. Böylece hangi kısıtın giderileceği konusunda çaba sarfedilecek ve ihtiyaç duyulmayan maliyetlerin altına girilmeyecektir. Sahip olunan gereksinimler, gereksinimleri karşılamak üzere oluşturulan bir uygulama ve gereksinimleri etkileyen kısıtları belirleyerek ilgili kısıtlar ışığında oluşturulan etkili, verimli bir güvenlik mekanizması tasarlanmalı ve geliştirilmelidir [29].

3.1.1 Kısıtlı kaynaklar

Herhangi bir bilgisayar ağ sisteminde gerekli olan güvenlik mekanizması geliştirilirken, geliştiriciler tarafından istedikleri sistemi ortaya çıkarabilmek amacıyla belirli kaynaklara ihtiyaç duyulmaktadır. Basit anlamda, söz konusu gereksinimler sistem kaynakları olarak da adlandırılabilir. Bu kaynaklar, geleneksel anlamda, enerji, işlemci gücü ve depolama alanlarıdır ki bu kaynaklar *KAA* ' ın yapısında bulunan küçük algılayıcılarda çok sınırlı miktarda bulunmaktadır.

- **Sınırlı hafıza ve depolama alanları:** *KAA* dahilinde çalışan küçük boyutlu algılayıcılar kendisinden beklenen işleri programlamak üzere geliştirilen kodları çalıştırmak için gerekli olan hafıza ve işlemci kaynaklarına sınırlı miktarda sahiptirler. *KAA* için etkili ve verimli bir güvenlik mekanizması inşa etmek için geliştirilen kod yapısını mümkün mertebe düşük işlemci ve hafıza kaynağına ihtiyaç duyacak şekilde planlamak gereklidir. Güvenlik için geliştirilen kod yapısı *KAA*' ın kendisinden beklenen asıl işlemlerini yapmasına engel olmamalıdır ve asıl görevleri sekteye uğratmamalıdır. Örneğin SunSpot *KAA* algılayıcısı incelendiğinde sahip olduğu özellikler 32 *Bit CPU* 512 *KB RAM* 4 *MB* anlık hafıza olarak görülebilir. Algılayıcı düğümlerin sahip olduğu bu özellikler nedeniyle oluşturulacak kod oldukça küçük boyutlu olmalı ve sistemi yormadan çalışmalıdır. *KAA*' ın asıl görevleri olan algılama ve uzun ömürlü olma görevlerinin yerine getirilmesi öncelikli hedef olmalıdır. Bunların dışında algılayıcı düğümlere istenilen her durumda müdahele edebilme imkanı bulunuyor ise kaynakların kullanılmasında daha rahat olunabilir [30].

- **Sınırlı enerji kaynakları:** Enerji konusu *KAA* için en önemli kısıtlardan biridir. Basit kullanım alanları dışında *KAA*' ı oluşturan algılayıcı düğümlerin yenisiyle değiştirilmesi veya yeniden şarj edilmesi yüksek operasyonel maliyetler,

bulundukları ortamın müdaheleye izin vermemesi gibi nedenlerle mümkün olamamaktadır. Bu nedenle, algılayıcın enerji kaynakları çok dikkatli kullanılmalı, israf edilmemeli böylece algılayıcıların ömürleri dolayısı ile *KAA*' ın ömrü de mümkün mertebe uzatılmalıdır. Algılayıcı düğüm üzerinde geliştirilen şifreleme, saldırı tespiti gibi programların ne kadar enerji tükettiği, düğümün ömrüne olan etkisi mutlaka hesaba katılmalıdır ve enerjiyi daha az tüketecek yöntemler üzerinde çalışılmalıdır. Güvenliği sağlaması gerekli olan ve bunun için geliştirilen bir güvenlik mekanizmasının *KAA*' ı çalışamaz ve kendisinden istenen görevleri yerine getiremez hale dönüştürmesi başarısız bir durumdur. Genel güvenlik yaklaşımları incelendiği zaman; şifreleme, şifre çözme, imza onaylama gibi işlemler işlemciyi meşgul etmekte, güvenlik mekanizmasının getirdiği fazladan verilerin gönderilmesi için daha çok enerjiye ihtiyaç duyulmaktadır. Ayyrıca, güvenlik parametrelerinin saklanması için de fazladan depolama alanlarına ihtiyaç duyulmaktadır [29].

3.1.2 Güvenli olmayan iletişim ortamı

Güvenli olmayan iletişim ortamı *KAA* yapısında karşılaşılan ve dikkate alınması gereken bir diğer kısıttır. Bu kısıt algılayıcı düğüm güvenliğine önemli ölçüde tehdit arz etmektedir. İletişim ortamında güvenliği sağlamak adına etkili protokollerin belirlenmesi ve uygulanması çok önemlidir [29] [10].

- **Güvenli olmayan transfer:** *KAA*' da paket tabanlı yönlendirme bağlantısızdır ve bu nedenle paket kayıplarına açık durumdadır. Bu açıklık *KAA* iletişiminin güvenli olmayan bir şekilde değerlendirilmesine neden olmaktadır. Paketler, bağlantı hatalarından dolayı kaybolabilir, zarar görebilir. Bu durumun oldukça kalabalık sayılarla ifade edilen algılayıcıların oluşturduğu ağ yapısında iyi bir protokol yapısı oluşturulmadıysa fark edilemeyebilir ve ağın verimli çalışmasını etkileyebilir. Tüm bunların yanında ağ üzerinde geliştirilen güvenlik mekanizmasına ait bir paket de kaybolabilir. Görüldüğü üzere hem ağın kendisinden bekleneni yerine getirebilmesi hem de güvenlik mekanizmasının sağlıklı çalışabilmesi için veri transferi esnasında protokollerin etkili bir şekilde kullanılması gerekmektedir.
- **Algılayıcılar arası tutarsızlık:** Algılayıcı düğümler arasında tutarlı bir kanal oluşturulsa bile bu durum tutarlı, güvenilir bir iletişim gerçekleşeceği anlamına

gelmeyecektir. Bu durum algılayıcı düğümlerin genel yayın yapan yapısından kaynaklanmaktadır. Birbirinden bağımsız olarak yayın yapan algılayıcı düğümlerin gönderdiği paketler iletişim esnasında karşılaşırlarsa bu durum düğümler arası karışıklığa neden olabilir ve bu durum tutarsızlık oluşturur böylece veri transferi başarısız olur. Kalabalık algılayıcı düğüm sayısına sahip bir *KAA* yapısında bu durum ana problemlerden biridir.

- **İletişimdeki gecikmeler:** Çoklu atlamalı yönlendirme, ağ sıkışıklığı ve algılayıcı düğümler üzerinde gerçekleştirilen işlemler ağ yapısı üzerinde gecikmelere neden olabilirler. Bu nedenle algılayıcı düğümler arasında eş zamanlılığın sağlanması zor bir durum olmaktadır. *KAA*' da oluşturulan güvenlik mekanizmasının olay durum raporlarına veya anahtar dağıtımına bağlı olması durumunda ağ üzerinde gerçekleşecek gecikmeler ve eş zamanlılığın sağlanamaması kritik bir konudur.

3.1.3 Çevresel etkiler

KAA' ın askeri ve sivil olarak çok fazla uygulama alanlarının olduğundan daha önce bahsedilmişti. Özellikle askeri açıdan incelendiği zaman *KAA* düşman topraklarında her türlü tehdite açık bir şekilde gelişi güzel bir durumda bulunabilirler. Bu korunmasız şartlar *KAA*' da geleneksel ağ yapısının karşılaşabileceği tehditlerden farklı olarak yeni tehdit anlayışı ortaya çıkarmaktadır. Yeni tehdit anlayışı yeni güvenlik gereksinimlerine neden olmaktadır [30].

- **Fiziksel saldırılar:** *KAA* açık, korunmasız alanlarda çevresel olumsuz etkilere maruz kalabilirler. Bu etkiler basitçe rüzgar, yağmur, toz, kar şeklinde sıralanabilir. Bu etkilere ek olarak düşman olarak nitelenebilecek alanlarda da *KAA*' ın uygulanması gerçekleştirilebilir. Gerçekleştirilen *KAA* yapısının tespit edilmesi durumunda fiziksel olarak bir saldırıya uğraması kaçınılmaz bir durumdur. Ayrıca açık, korunmasız alanlarda bulunması nedeniyle isteyerek gerçekleşecek saldırıların yanında istemsiz olarak da gerçekleşebilecek saldırılara maruz kalınabilir. Bu istemsiz fiziksel saldırılara doğada yaşayan hayvanların hareketleri örnek olarak verilebilir. Açıkça görüldüğü üzere *KAA* kişisel bilgisayar ve ağ bileşenlerinden farklı olarak gelişebilecek tehdit yaklaşımlarına maruz kalabilirler.

- **Uzaktan yönetim:** Geliştirilen uygulama kapsamında kullanılan ağın uzaktan yönetimi özellikle bakım konusunda sıkıntılar yaşatabilmektedir. Bu noktada yine *KAA*' ın düşman topraklarında oluşturulması örneği verilebilir. Bu durumda ağa müdahele için fırsat bulunmayabilir ve bu nedenle bakım veya fiziksel karıştırmaların engellenmesi konusunda sıkıntılar yaşanabilir. Dost kuvvetler ile hiçbir bağı kalmayan *KAA* bulunduğu çevrede yalnızdır ve iyi bir öz güvenlik mekanizmasına muhtaçtır. Gliştirilecek *KAA* yapısında yazılımsal olarak güvenliğin yanı sıra gizleme veya kamufile etmek gibi önlemler de alınabilir [30].

- **Merkezi yönetim noktasından yoksunluk:** Bir algılayıcı ağ dağıtılmmış yapısı ile birlikte merkezi bir yönetim noktasından yoksun durumdadır. Bu nedenle ağ organizasyonun iyi olması hem ağın güvenliği hem de hata hoşgörüsü açısından çok önemlidir.

3.2 *KAA* İçin Güvenlik Gereksinimleri

Geleneksel güvenlik yaklaşımında karşılanması gereken üç ana gereksinim bulunmaktadır. Bu gereksinimler gizlilik, bütünlük ve erişilebilirlik olarak sıralanabilir. Ancak, önceki bölümde bahsedildiği üzere *KAA*' ın gendine özgü özellikleri ve kısıtları bulunmaktadır ve bu kısıtlar ve özellikler nedeniyle farklı gereksinimler de ortaya çıkmaktadır. *KAA* için ihtiyaç duyulan ve karşılanması gereken güvenlik gereksinimleri bu bölümde açıklanmıştır.

3.2.1 Veri gizliliği

Veri gizliliği bilgisayar ve ağ uygulamalarında dikkat edilmesi gereken en önemli konulardan biridir. Geliştirilen her güvenlik uygulamasının odaklandığı ilk konu veri gizliliği konsudur. *KAA* iletişiminde ağda bulunan verinin dinlenmesi oldukça kolaydır. Gizlilik için altın kelime **"bilgi değerlidir"** sözcük bütünüdür ve kullanıcılar bilgilerini yetkisiz, istenmeyen kişilerin eline geçmesini istemezler. Gizliliği korumak için bilgi şifrelenmelidir. Şifreleme işlemleri simetrik veya asimetrik anahtarlama yöntemleri ile yapılabilir. Asimetrik şifreleme yöntemleri daha güçlüdür. Fakat enerji tüketimi açısından elverişli değildir. *KAA* kullanıcısı simetrik şifrelemeyi tercih ederse anahtarını çok dikkatli şekilde saklamalıdır [1].

- Bir *KAA*' da algılayıcı düğümün içerdiği veriler çok kıymetli olabilir. Özellikle askeri uygulamalarda çok hassas verilerle çalışılıyor olunabilir. Bu nedenle dikkat edilmesi gereken bir konudur.

- Algılayıcı düğümlerin içinde bulunan verilerin dışında, düğümler arası gerçekleşen iletişim esnasında da kıymetli veri akışı bulunabilir. Söz konusu kıymetli verilere anahtar dağıtım paketleri örnek olarak verilebilir. Bu nedenle düğümler arası iletişimin güvenli bir kanal aracılığı ile gerçekleşebilmesi önemlidir.

- Ayrıca, düğümler arasındaki iletişimde giden/gelen paketler ilk etapta değerli olarak görünmese bile ağ analizcileri açısından ağın topolojisini anlayabilmek açısından çok kıymetli olabilirler. Genel iletişim paketlerinin de güvenli bir şekilde, üçüncü kişilerin elde edemeyeceği şekilde iletilmesi sağlanmalıdır. Paketler bu amaçla şifrelenebilirler.

3.2.2 Veri bütünlüğü

Gizliliğin sağlanmış olması verinin güvende olduğu anlamına gelmemektedir. Karşı kimse elde edilmek istenen veriyi değiştirebilir. Bu durum *KAA* için düzensizliğe ve yanlışlıklığa neden olur. Örneğin zararlı bir düğüm olmaması gereken bir veriyi üretip *KAA* içinde iletebilir. Bu durum ağın yanlış bir şekilde yönlendirilmesine neden olur. Fazladan, yanlış veri üretiminin yanında verinin aybolması veya zarar görmesi de *KAA* yapısını manipule edebilir. Bütünlük ağ üzerinde gönderilen verini değiştirilmesine engel olmak ve değiştirilmediğine emin olmaktır. **Bilgi sadece doğru ise bir değere sahiptir**. *KAA* bulundukları fiziksel ortamı gözlemler ve bu ortamdan elde ettikleri veriyi toplarlar. Toplanan verilerin işe yaraması için değiştirilmediğinden emin olunması gerekir. Kablolu ve enerji kısıtı olmayan ağ yapılarında bütünlik dijital imzalar ile sağlanmaktadır fakat bu yaklaşım *KAA* için uygun değildir çünkü bir cihazdan diğer cihaza gönderilen verilere fazladan bitlerin eklenmesine neden olur. Bir diğer olumsuzluk ise fazladan hesaplama kaynağına ihtiyaç duyulmasına neden olur. Gönderilecek veride yaşanan artış ve fazladan hesaplama yeteneği gereksinimi nedeniyle bu yöntem uygun olmaz fakat enerji etkili bir *STS* ile bütünlük sağlanabilir.

3.2.3 Erişilebilirlik

Genel olarak bilinen şifreleme algoritmalarının *KAA* üzerinde gerçeklemesi uygun değildir. *KAA* üzerinde gerçekleştirilecek her türlü fazladan işlemin karşılığında bir maliyeti bulunmaktadır. Bazı yaklaşımlar güvenlik mekanizmsını sağlayan kodun mümkün olduğu kadar küçülmesini sağlamak için modifiye etmeyi amaçlamaktadır. Bazı uygulamalar ise aynı hedefe gerçekleşen iletişim sayılarını azaltmaya çalışarak erişmeyi amaçlamaktadır. Gizliliği sağlanarak değerli kılınan verinin bütünlüğü ile kullanılabilir hale getirilmesi önemlidir fakat eğer ki veriye ihtiyaç duyulduğu anda erişilemiyorsa önceden gerçekleştirilen başarılı gizlilik ve bütünlük işlemlerinin bir önemi kalmamaktadır. Bu nedenle erişilebilirlik için gerçekleşen işlemler aşağıdaki nedenlerden dolayı yetersiz kalabilmektedir [30]:

- Fazladan gerçekleşen işlemler fazladan enerji tüketimine neden olacaktır. Bu tüketim algılayıcı düğümün enerjisinin çabuk tükenmesine dolayısı ile zamanla artık erişilemez hale gelmesine neden olacaktır.
- Fazladan gerçekleşen iletişim sayıları da fazladan enerji tüketimine neden olacaktır. Algılayıcı düğüm üzerinde fazladan gerçekleşen her işlem o düğümün ömrünü biraz daha kısaltacaktır.

3.2.4 Veri güncelliği

Bazı *KAA* uygulamalarında bulunulan ortamdan elde edilen verinin geçerli olabilmesi için mutlaka güncel olması gerekebilir. Bu güncelliğin veriyi değerli kıldığı durumlarda gizlilik, bütünlük, erişilebilirlik şartlarının ardından muhakkak elde edilmesi gereken sonuç verinin tazeliğidir. Anlık karar verilmesi gereken durumlarda belki bir dakika öncesinden bile elde edilen verinin hiçbir kıymeti bulunmamaktadır. Verinin tazeliği ve güncelliği konusuna, kendini tekrar eden verilerin iletilmesi de dahil edilebilir. Kendini tekrar eden bir verinin iletilmesi değişen durumların tespit edilebilmesi konusunda problem çıkaracağı apaçık bir gerçektir.

3.2.5 Kendini yönetebilme

Temelde bakıldığı zaman *KAA* birbirinden bağımsız ama birbirleri ile işbirliği ile çalışması gereken algılayıcı düğümler bütünüdür. Bu kapsamda düğümler arası esnekliğin sağlanması ve yeni durumlara uyumun gerçeklnebilmesi önemlidir. Hata hoşgörüsünü sağlayan bu durum ağın kendisini yönetebilemsini sağlayacak ve ağ gerçekleştirmesi beklenen amaçlar için çalışmaya her durumda devam edebilecektir. *KAA*' da sabit olarak geliştirilmiş bir yönetim altyapısı bulunmamaktadır. *KAA* aktif olarak değişen her durumda kendini yönetebilecek ve değişikliklere adapte olabilecek yeteneklere sahip olmalıdır. Bu değişkenlik durumu *KAA* için tasarlanacak olan güvenlik yönetim mekanizması için de önemli bir zorluk olarak karşımıza çıkmaktadır. Eğer bir *KAA* uygulaması kednini yönetebilir yeteneğine sahip olmazsa veya gerekli esnekliği sağlamayıp hata hoşgörülü bir sistem olmazsa karşılaşması muhtemel bir saldırı sonucunda tamamen işlemez hale gelebilir ki bu da *KAA*' ı saldırganlar için çok basit bir hedef haline getirir.

3.2.6 Senkronizasyon

Birçok *KAA* uygulaması çalışmalarını zaman senkronizasyonu üzerinden yürütmektedir. Bu kapsamda enerji tasarrufunu sağlayabilmek amacıyla algılayıcı düğümler sadece çalışması gereken zamanlarda uyandırılıp çalıştırılabilir ki bunu başarabilmek için mutlaka zaman senkronizasyonuna ihtiyaç duyulmaktadır. Belirli zaman aralıklarında kapatılan algılayıcı düğümün enerjisi çok daha uzun süre yetecek duruma gelmektedir. Senkronizasyon için düğümler arası veri iletim zamanı da kullanılabilir.

3.2.7 Güvenli yer bildirimi

KAA uygulamalarının birçok tehlikeye açık, yalnız, koruma veya bakım amaçlı müdahelelerin gerçekleşemeyeceği alanlarda oluşturulabileceğinden üçüncü bölümde bahsedilmişti. Farklı saldırı türlerine konu olabilecek *KAA* için [29]' de güvenli yer bildirimi konusu ile ilgili ayrıntılı bilgiler bulunmaktadır ve bu konuda ortaya çıkan çalışmaları, gerçekleşen algoritmaları detaylı olarak anlatmaktadır. Genel anlamda incelendiğinde ise *KAA* bünyesinde özellikle meydana gelen hataların yerlerinin belirlenebilmesi çok önemlidir ancak bu belirleme işlemini saldırganlar karşı silah

olarak kullanmak istemektedir. Bu nedenle yer belirleme işleminin gizli, güvenli bir şekilde gerçekleştirilmesi gerekmektedir [29].

3.2.8 Parolalaşma

KAA'a saldıran bir saldırgan sadece ağ üzerindeki paketlerde değişiklik yapmayı istememektedir. Bunun yanında ağ üzerindeki tüm paket akışını yeni ve yanlış paketler de üreterek tamamiyle değiştirmek isteyebilir. Bu nedenle alıcı düğüm kendisine gelen paketin doğruluğundan emin olmalıdır ve bu doğruluğu kanıtlayabilmek için bir karar mekanizması çalıştırılmalıdır. Diğer yandan ağın yönetilmesi ile ilgili işlemlerde de parolalaşma ihtiyacı duyulmaktadır. Bir algılayıcı düğüm, parolalaşma mekanizması ile kendisine gelen paketin doğru olup olmadığını anlayabilir bu şekilde bir karar mekanizması çalıştırabilir [29].

3.3 *KAA* İçin Gerçekleşebilecek Saldırılar

KAA birçok saldırı türüne karşı zayıflıklar taşımaktadır. Birçok farklı yol izlenerek gerçekleştirilebilecek bu saldırılar çoğunlukla *DOS* saldırıları olarak meydana gelse de trafik analizleri yapan, gizliliği hedef alan veya fiziksel olarak zarar vermeyi amaçlayan saldırılar da gerçekleşmektedir. *DOS* saldırıları *KAA* algılayıcı düğümleri arasındaki iletişimi karıştırmaktan, 802.11 *MAC* protokolünü bozmaya kadar geniş bir yelpazede gerçekleştirilebilir. *KAA*' da gerçekleştirilebilecek saldırıları sadece *DOS* yaklaşımı ile sınırlamak doğru değildir. Bu kapsamda *KAA* uygulamalarına karşı algılayıcı düğümleri ele geçirmek, yönlendirme protokollerini değiştirmek, yanlış veri iletimi sağlamak, ağ topolojisini elde etmek gibi saldırı tiplerinden söz edilebilir. *KAA* saldırılarının genel anlamdaki açıklamaları kısaca çizelge 3.1 üzerinden incelenebilir. Literatür incelendiğinde gerçekleştirilen çalışmaların daha çok *DOS* saldırıları üzerinde yoğunlaştığı görülmüştür. Saldırı tekniklerinin birçoğu *KAA* uygulamasını cevap veremez hale getirmek için çalıştığı ve bu yönde daha çok algılayıcı düğümlerin kısıtlı kaynaklarının hedef olarak seçildiği veya sistemin zayıf noktası olarak kullanıldığı görülmüştür. Bu kapsamda algılayıcı düğümlerden daha güçlü olanından yayınlar yapılarak *KAA* iletişimi baskılanmaya çalışılmıştır. Ayrıca *KAA* üzerinde olması gereken veri akışını değiştirecek ve olumsuz

yönde etkileyecek saldırılar da karşılaşılabilecek saldırı tipleri arasındadır. Kendisi üzerinden geçen paketleri silen ve sistemden atan düğümlerin ağın gereklerini yerine getirmesini engellemesi de saldırı tiplerine örnek olarak gösterilebilir. Saldırı çeşitliliği incelendiğinde iyi planlanmış bir güvenlik mekanizmasının ve yönetiminin ne kadar önemli olduğu ortaya çıkmaktadır. Bu bölümde *KAA* için gerçekleşebilecek saldırı tipleri açıklanacak ve karşılaşılması muhtemel tehditler, karşı yöntemler incelenecektir [11] [12] [13] [18].

Çizelge 3.1: *KAA*' da genel saldırı kavramları.

Saldırı Tipleri	
Saldırı Tipleri	**Açıklamalar**
Hizmet Engelleme Saldırıları	Hizmet engelleme saldırılarının temel amacı sistemi cevap veremez hale getirmektir.
Yeniden Yönlendirme	Yeniden yönlendirme saldırılarında KAA üzerinde bulunan olması gereken bilgi akış rotası olumsuz yönde değiştirilir.
Seçmeli iletim	Bu saldırıda zararlı düğümler normal bir düğümmüş gibi davranırlar ve paketleri iletirler fakat bu işlem esnasında bazı paketleri yok ederler.
Karadelik Saldırıları	Bu saldırı tipinde saldırgan ağ içinde bir düğümü ele geçirir ve bu düğümü saldırı gerçekleştirmek için kullanır. Ağı sürekli denler ve ana düğüme ulaşmak için en kısa rotayı belirlemeye çalışır
Sybil Saldırıları	Bir adım gerideki düğüm açısından iletişim gayet normal olarak devam etmektedir. Saldırgan düğümün bir adım ilerisindeki düğüm açısından ise herhangi bir iletişim olmadığı için her şey normaldir. Bu durumda sadece saldırgan düğüm problemin farkındadır ve onunda bu durumu rapor etmek gibi bir niyeti olmayacağı açıktır.
Solucan Deliği	Solucan deliği (wormhole), birbirinden uzakta bulunan iki saldırgan düğümün aralarında oluşturdukları özel bir iletişim yolu ile (VPN benzeri) birine gelen veri paketlerinin doğrudan diğerine aktarılması yoluyla yönlendirmenin bozulmaya çalışıldığı saldırı biçimidir .
Hello Baskını Saldırıları	Saldırgan güçlü çıkış gücüne sahip bir verici vasıtasıyla ağa "hello" mesajı gönderir. Bu mesajı alan düğümler vericinin kendilerine komşu bir düğüm olduğunu düşünür ve yönlerdirme tablolarını buna göre oluştururlar.

3.3.1 *DOS* saldırıları ve çeşitleri

Genel olarak basit anlamda *(KAA)* için gerçekleştirilen *(DOS)* saldırıları bir veya birden fazla algılayıcı düğümün karıştırılması ve görev yapamaz hale getirilmesi yaklaşımı üzerine inşa edilmektedir. Karıştırma, *(KAA)* tarafından kullanılan iletişim kanallarının baskılanması ve kullanılmaz hale getirilmesidir. Bir ağ üzerinde karıştırma iki türlü olmaktadır:

- **Sabit (sürekli) karıştırma** olduğu durunlarda *(KAA)* sürekli olarak karıştırılır ve hiçbir şekilde algılayıcı düğümlerin ne alıcı ne de verici olarak iletişim kurmasına izin verilmez. Ağ işlemez hale getirilmiş olur.
- **Atlamalı (aralıklı) karıştırma** olduğu durumlarda ise karıştırma belirli aralıklarla gerçekleştirilir ve ağ yalnızca belirli zamanlarda kesintiye uğratılır. Sürekli bir işlemezlik yoktur ancak bu kesintili çalışma dolayısı ile de ağın yerine getirmesi gereken görevi mevcudiyetini sağlayamaz.

DOS kapsamındaki saldırılar veri bağı katmanı seviyesinde de gerçekleşebilir. Bu durumda ağın kullandığı protokol üzerinden saldırılar gerçekleştirilerek karşı tarafın paketi alıp almadığının belli olmaması için saldırı gerçekleştirilebilir. Böylece ağ üzerinde algılayıcı düğümler tarafından tekrar tekrar gönderilmiş çok fazla paket olacak ve bu durum ağın olumsuz çalışmasına neden olacaktır. Ayrıca sürekli paketler göndermek zorunda kalan algılayıcı düğümlerin enerjileri de çok çabuk tükenecektir.

Ağ katmanı seviyesinde yeniden yönlendirma saldırıları şeklinde saldırılar meydana gelebilmektedir. Saldırgan gelen paketleri istenilen düğümden bir başkasına yönlendirebilir ve bu durumda ağ üzerinde gecikmeler meydana gelir. Bu gecikmelerin boyutu ve sayısı arttıkça ağın çalışma performansı düşer ve ağ verimsizleşir.

Taşıma katmanı seviyesinde de farklı saldırı tipleri ile karşılaşılabilir. Bu tip saldırılarda veri akışı kullanılır. Yani, bir algılayıcı düğüme çok fazla iletişim isteği gönderilir ve ilgili düğüm bu isteklere cevap vermeye çalışmaktan artık kendi görevini yapamaz hale gelir. Algılayıcı düğümün sahip olduğu zaten kısıtlı olan kaynaklar bu bağlantı isteklerine cevap verebilmek için tahsis edilir ve bu yönde kullanılır.

3.3.2 Sybil saldırıları

Sybil saldırıları zararlı bir aygıtın birden fazla kimlik üstlenmesi ile gerçekleştirilen eylemler olarak tanımlanmaktadır. Aynı zamanda Sybil Saldırıları ağ üzerindeki yönlendirme algoritmaları, veri toplama yaklaşımları, kısıtlı kaynak tashis edilmesi yöntemleri üzerinde etkilidir. *KAA* dahilinde bulunan algılayıcı düğümler arasında zararlı bir şekilde çoklu kimlik yeteneğine sahip olan bir düğüm bulunduğu durumlarda ağın yönlendirme mekanizması kolaylıkla olumsuz yönde etkilenebilir.

3.3.3 Trafik analiz saldırıları

KAA baskın ve daha güçlü olan ana terminal ile iletişim halinde olan düşük güce sahip birçok algılayıcı düğümden oluşmaktadır. Ağı oluşturan düğümlerin temel görevi bir araya getirdikleri verileri ana terminale ulaştırmaktır. Bazı durumlarda, *KAA* uygulamasını çalışmaz hale getirmek isteyen bir saldırgan basit anlamda ana terminali olumsuz yönde etkileyerek ağı çalışmaz hale getirmek isteyebilir. Bu kapsamda paketler şifrelenmiş olsa bile paketlerin içeriği ile ilgilenmeden gerçekleştirilebilecek saldırı yaklaşımları olduğundan dolayı sadece paket şifreleme tek başına işe yaramayabilir. *KAA* yapısında bulunan düğümlerden ana terrminale daha yakın olan düğümler üzerinden daha fazla sayıda paket geçecektir böylece paketin içeriği görülmese bile paket sayısının takip edilmesi ile ana terminale ulaşılmak istenebilir. Bunun yanında belirlenen bir algılayıcı düğüm üzerinden ana terminale yönlendirilmek üzere yola çıkan bir paket takip edilmek istenilebilir. Bunu yapmak için de paketin geçtiği algılayıcı düğümler üzerinde fiziksel değişiklikler (düğüm üzerinde ışık yakmak gibi) yapılmak istenebilir.

3.3.4 Fiziksel saldırılar

KAA' ı geleneksel ağ yapılarından ayıran en önemli özelliklerden biri açık alanlarda fiziksel müdahelelere karşı savunmasız olmasıdır ve bu özellik hakkında ayrıntılı açıklamalar önceki bölümlerde yapılmıştır. Bu bölümde ise fiziksel ortamda savunmasız olan bir *KAA* uygulamasının karşı karşıya kalabileceği saldırılar üzerinde durulacaktır. Öncelikler algılayıcı düğümlerin bulunduğu ortamları müdahele edilebilir ve müdahele edilemez olarak ikiye ayırmak gerekir. Müdahele edilebilir

ortamlarda (ev, ofis v.b.) fiziksel güvenliğin sağlanması daha kolay olabilecektir. En azından doğa şartlarından kaynaklanan (yağmur, toz, kar, rüzgar, hayvanlar v.b.) fiziksel zararların önüne kolayce geçilecek önlemler alınabilecektir. İkinci olarak ise düşman alanlarında bulunabilecek algılayıcı düğümler için müdahele, bakım gibi işlemler mümkün olmayabilir ve doğa şartları dışında da düşman güçleri tarafından saldırılara maruz kalabilir. Düşman güçleri tarafından tespit edilen ve ele geçirilen bir *KAA* uygulamasında:

- Algılayıcı düğüm fiziksel olarak zarara uğratılıp, tamamen yok edilebilir.
- Algılayıcı düğüm dahilinde bulunan kriptografik veya herhangi bir değere sahip bilgi düşman güçleri tarafından ele geçirilebilir.
- Düşman güçleri algılayıcı düğümü yeniden programlayıp karşı saldırı olarak kullanabilir.

3.3.5 Düğümler ile gerçekleştirilen saldırılar

Bu saldırı tipinde ise saldırgan basit anlamda *KAA* uygulamasında bulunan algılayıcı düğümlere içlerinden birisinin kimliğini kullanarak yeni bir düğüm eklemeye çalışması ve bu işlemi başardıktan sonra uygulama içindeki iletilen paketleri yeniden yönlendirerek, paketleri yok ederek, sahte paketler üreterek çalışmasını engellemeyi amaçlamasıdır. Saldırgan algılayıcı düğümlerden birini fiziksel olarak ele geçirirse kriptografik anahtarları da ele geçirip kendi düğümünü bu anahtarlara sahip bir şekilde kopyalayım ağın önemli noktalarına kendi düğümünü yerleştirebiilr. Bu sayede ağı manupile edebilir veya ağın o stratejik nokta ile iletişimini kesebilir.

3.3.6 Veri gizliliğini hedef alan saldırılar

KAA uygulamalrının kullanımı ile birçok alanda çok basit ve çok başarılı çözümler üretilmiş, küçük ve ucuz algılayıcı düğümler ile veri toplama ve değerlendirme süreçleri çok kolay hale getirilmiştir. Bu önemli faydalarının yanında *KAA* uygulamalarının karşılaştığı problemlerin başında verilerin gizliliğinin korunması gelmektedir. Zaman zaman istihbarat amaçlı kullanılabilecek *KAA* uygulamalarının bir araya getirdiği veri kümelerinin deşifre edilmemesi ve başkalarının ulaşmasının

engellenmesi önemlidir. Algılalayıcı düğümler arasında iletilen sayısız paketin alınacak kriptografik önlemlerle bir araya getirilip anlamlandırılması zor bir durumdur ancak imkansız değildir. Düğümlerin sahip olduğu gizliliğe gerçekleştirilebilecek saldırılar şu şekilde sıralanabilir.

- ***KAA*'ı dinlemek:** Veri gizliliğine karşı gerçekleştirilebilecek en etkili saldırı türüdür. *KAA* içindeki iletişim dinlenerek veriler bir araya getirilmeye ve anlamlandırılmaya çalışılır.
- ***KAA*' da trafik analizi yapmak:** *KAA* uygulaması içindeki paket akışının hem dinlenip hem de takibinin yapılması ile ortaya çıkan saldırı türüdür. Paket rafiğinin analizi yapılarak özel görevleri olan algılayıcı düğümler veya ana terminal tespit edilmeye çalışılır. Bu sayede ağın kritik noktaları tespit edilir ve daha az saldırı ile daha fazla etki elde edilebilir.
- ***KAA* içine zararlı düğüm gizlemek:** Saldırgan kendi algılayıcı düğümünü *KAA* algılayıcı düğümlerinin arasına gizleyebilir ve bu düğümler normal bir düğüm gibi davranabilirler. Bu sayede iletilen paketleri yeniden yönlendirebilir, gizliliği ihlal edecek şekilde diğer düğümleri dinleyebilir.

Veri gizliliğini sağlamak için sistem yöneticileri farklı yaklaşımlar ile yeni yöntemler üzerinde çalışmalıdır. Güvenlik mekanizmaları için yazılan programların yanı sıra saldırganları yanıltacak aldatıcı yöntemler geliştirilmelidir. Örneğin düşman arazisi içine yerleştirilecek bir *KAA* algılayıcı düğümü bulunduğu ortama uyum sağlayacak şekilde kamufile edilebilir ve fark edilmesi veya tespit edilmesi zorlaştırılabilir.

3.4 *KAA* İçin Savunma Mekanizmaları Ve Saldırı Tespit Sistemleri *(STS)*

Önceki bölümlerde *KAA* uygulamalarının sahip olduğu kısıtlı kaynaklar, bu kısıtlı kaynakların neden olduğu zaafiyetler ve *KAA* uygulamalarının karşılaşabileceği güvenlik tehditleri açıklandı ve tehditlere karşı oluşturulabilecek mekanizmaların nasıl planlanması gerektiği ile ilgili bilgiler verildi. Bu bölümde *KAA* uygulamaları için uygulanabilecek güvenlik mekanizmaları örneklenecek ve bu mekanizmalardan

Saldırı Tespit Sistemleri *(STS)* ayrıntılı olarak anlatılacaktır. *KAA* uygulamalarında oluşturulabilecek güvenlik mekanizmaları aşağıdaki gibi sıralanabilirler [31].

3.4.1 Saldırı tespit sistemleri *(STS)*

İlk saldırı tespit sistemi yaklaşımının (STS) 1980' li yıllarda bilgisayar sistemlerine yetkisiz erişimleri tespit edebilmek amacıyla Doroty Denning tarafından ortaya atılması ile birlikte günümüze kadar STS üzerinde birçok araştırma yapılmış ve yeni ve farklı yaklaşımlar gerçekleştirilmiştir [1]. Ağ ve bilgisyar güvenliği yönetiminde iki önemli başlık karşımıza çıkmaktadır. Bu başlıkları saldırı önleme yaklaşımı ve saldırı tespit yaklaşımı olarak sıralayabiliriz. Uygulanmakta olan bir bilgisayar ve ağ güvenliği planında, saldırı önleme (şifreleme, yetkilendirme, parola vs.) yaklaşımını ilk güvenlik hattı olarak adlandırırsak, bu hattın ihlal edilmesi durumunda gerekli tedbirlerin alınmasını sağlayacak, söz konusu saldırıyı en az zararla veya zararsız olarak karşılayacak ayrıca özellikle sistem içinden olabilecek saldırılara caydırıcılık getirecek saldırı tespit yakklaşımı ise söz konusu güvenlik planının ikinci güvenlik hattı olarak adlandırılabilir [30]. STS, bilgisayar veya ağ sistemini izleyerek dahili veya harici saldırıları rapor eden yazılım veya donanım olarak adlandırılabilir. STS' nin çalışma prensibi ise mevcut bilgisayar, ağ sistemini ve sistemin kullanıcılarını gözlemleyerek sistem üzerindeki işlemleri bilinen "saldırı imzalarıyla" eşleştirerek veya önceden normal olarak belirlenen sistem yapısına aykırı düştüğünü tespit edip anormal olarak etiketleyerek saldırıları yakalayabilmek şeklindedir. STS, üzerinde bulunduğu sistemi izleme yaklaşımına göre **"Ağ Tabanlı Saldırı Tespit Sistemi (ATSTS)"** ve **"Sunucu Tabanlı Saldırı Tespit Sistemi (STSTS)"** olarak ikiye ayrılmaktadır. ATSTS yaklaşımında kullanılınan STS üzerinde bulunduğu bilgisayar ağı üzerinde gerçekleşmesi muhtemel saldırıları tespit etmeyi amaçlarken, STSTS yaklaşımında ise STS üzerinde bulunduğu belirli bir bilgisayar üzerinde gerçekleşebilecek saldırıları tespit etmeyi amaçlamaktadır. Ayrıca; gerçeklenen STS (Ağ tabanlı veya Sunucu tabanlı) saldırıları tespit etme tekniğine göre de sistem üzerindeki işlemleri bilinen siber saldırıların imzalarıyla karşılaştırarak çalışan **Kötüye Kullanım** ve normal olmayan herşey anormaldir yaklaşımıyla önceden belirlenmiş normal tanımına aykırı olan tüm işlemleri anormal olarak etiketleyerek çalışan

Anormallik Tespiti olarak ikiye ayrılmaktadır. Hangi yaklaşımın veya tekniğin benimseneceği, sistemin organizasyonel yapısına, kullanıcı profiline, erişilebilir kaynaklara, organizasyonun sahip olduğu risk yapısına ve sistem altyapısına göre değişmektedir.

Saldırı Tespit Sistemleri *(STS)* konumlandırıldıkları ortama göre temelde üç sınıf altında değerlendirilmektedir.

3.4.1.1 Ağ tabanlı saldırı tespit sistemleri *(ATSTS)*

Ağ Tabanlı Saldırı Tespitini akıllı ağ donanımı olarak düşünebiliriz. Bu donanımlar ağ trafiğini monitör eder ve bazı işaretlere göre bu trafiği analiz eder. Olay kaynağı olarak ağ üzerinden akan paketler kullanılır. Ağ üzerindeki kritik noktalara yerleştirilmiş ve algılayıcılar ile ağ trafiğini izleyen sistemlerdir. ATSTS aygıtının bulunduğu ağın kritik noktalarına konumlandırılması önemlidir [30].

3.4.1.2 Sunucu tabanlı saldırı tespit sistemleri *(STSTS)*

Sunucu Tabanlı Saldırı Tespit Sistemleri *(STSTS)* bulunduğu sistem üzerinde bir yazılım olarak karşımıza çıkar. Saldırı Tespit Sisteminin gerçeklenebileceği ortamlar; bilgisayarlar, bazı uygulamalar, tarayıcılar, harici aygıtlar (yazıcılar v.s.), sunucular, ağ bileşenleri olabilir. Sunucu tabanlı STS'nin her iki yaklaşımında da sistem dosyaları üzerinde yapılmak istenen değişiklikleri kontrol etmek üzere oluşturulan bir yapı bulunmaktadır. Bu kontrol dosya üzerinde MD5 gibi hash fonksiyonu kullanan kriptografik bir checksum tarafından gerçekleştirilmektedir. Eğer ki checksum değerleri örtüşmezse Sunucu Tabanlı STS dosyanın değiştirildiğini anlar ve bu bilgiyi rapor eder [20].

3.4.1.3 Karma saldırı tespit sistemleri

Bazı sistemlerde, *STS* konumlandırıldıkları ağ yapısında hem Ağ Tabanlı Saldırı Tespit Sistemleri *(ATSTS)* yaklaşımını hem de Sunucu Tabanlı Saldırı Tespit Sistemleri *(STSTS)* yaklaşımını benisenebilir. Bu tip yaklaşıma en etkin yöntem Mobil Ajanlar (Mobile Agents) olarak gösterilebilir. Literatürde etmen tabanlı sistemler olarak da adlandırılan Mobil Ajanlar sahip oldukları otonomi ve hareketsellik özellikleri

sayesinde ağ yapısı üzerinde hem ağ tabanlı hem de sunucu tabanlı yaklaşımların uygulanabilmesine başarı ile olanak sağlarlar [30].

Saldırı Tespit Sistemleri *(STS)* kullandıkları saldırı tespit tekniğine göre temelde üç sınıf altında değerlendirilmektedir.

3.4.1.4 Anormallik tespiti

Anormalik Tespiti yöntemini kullanan bir *STS* yaklaşımında sistem davranışları incelenerek normal davranış durumlarının dışına çıkılması halinde alarm üretilir. Sistem davranışlarının incelenmesine ve bu davranışların normal veya anormal olarak etiketlenmesine dayanan bir çalışma prensibi bulunmaktadır. Bir ağ yapısı dahilindeki bir kullanıcının çalışma saatleri sabitse ve bu kullanıcı mesai saatleri dışında sisteme giriş yaptıysa bu durum anormal olarak etiketlenerek sistem yöneticisine alarm üretebilir veya yine bir kullanıcı yetkisi olmadığı halde sistem dosyalarına erişmeye veya değişiklik yapmaya çalışıyorsa, bir program kernel erişimi istiyorsa bu durumlar da anormal olarak etiketlenebilir. Bu bakımdan Anormallik Tespiti karşılaşacağı yeni saldırılar karşısında başarı sağlayabilirler. Bu başarıyı sağlayabilmek için normal durumların çok iyi planlanması gerekecektir [19].

Anormallik Tespiti yaklaşımının zaafiyet gösterdiği nokta ise çok fazla yanlış alarm üretmesidir. Günümüz dinamik dünyasında karşılaşılabilecek her durumun önceden belirlenerek normal olarak tanımlanabilmesi mümkün olamamaktadır. Muhakkak ki çok defa ilk defa karşılaşılan veya plansız şekilde gerçekleştirilmesi gereken durumlar olabilmektedir. Anormallik Tespiti yaklaşımı bu nedenle zararlı olmasa dahi birçok durum için anormal etiketlemesi yapmakta ve bu durum çok sayıda yanlış alarm üretilmesine neden olmaktadır. Yukarıda verilen örnek paralelinde düşünülürse mesai saatleri dışındaki kullanıcı aktivitelerini anormal olarak niteleyen ve anormallik tespiti yapan bir *STS* yaklaşımında zararlı aktivitede bulunmayan bir kullanıcı da anormal yani zararlı olarak nitelendirilebilir.

3.4.1.5 Kötüye kullanım tespiti

Kötüye Kullanım Tespiti mevcut durumu önceden bilinen saldırı durumları ile karşılaştırarak saldırı tespitini gerçekleştirmeye çalışmaktadır. Kullandığı teknik

itibariyle **İmza Tabanlı Yöntem** olarak da adlandırılmaktadır. Söz konusu yöntem mevcut ağ hareketlerini önceden bilinen saldırılar ile karşılaştırdığından dolayı bilinen saldırı tiplerine karşı çok başarılıdır ancak yeni bir saldır türü ile karşılaştığı durumlarda aynı başarıyı gösterememektedir. Bu yöntem uygulanması en kolay yöntemdir çünkü sistem dahilinde bilinen saldırılara göre kurallar oluşturulur ve sistem saldırı olup olmadığına bu kurallara göre karar verir. Sistemin daha önceden karşılaştığı bir e-posta türünün zararlı olduğu sistem tarafından tecrübe edilmişse söz konusu ile ilgili sistem dahilinde bir kural oluşturulur ve aynı e-postanın sisteme tekrar gelmesi durumunda oluşturulan kurala takılarak sisteme zarar vermesi engellenir. Sistem basit anlamda bu şekilde çalışmaktadır [19].

Kötüye Kullanım Tespiti' nin yeni karşılaştığı saldırılara karşı zaafiyetinin bulunduğundan bahsedilmişti. Bu durumda *STS* kendisine daha önceden zarar vermiş e-posta ile ilgili olarak gerekli kuralı oluşturmuş bile olsa, bu e-posta üzerinde gerçekleştirilecek bir kaç değişiklik ile oluşturulan kuralın atlatılması sağlanabilecektir.

3.4.1.6 Karma saldırı tespiti

Daha önce Anormallik Tespiti ve Kötüye Kullanım Tespitinin avantaj ve dezavantajlarından bahsedilmişti. Bu kapsamda Kötüye Kullanım Tespiti yaklaşımı bilinen saldırı türlerine karşı başarılı olmasına rağmen yeni karşılaştığı saldırılara karşı başarısız olmaktadır. Ayrıca Anormallik Tespiti yaklaşımında ise yeni saldırılara karşı başarı sağlanırken gereğinden fazla yanlış alarm üretilmektedir. Bu iki yöntemin birleştirilmesi ile hem bilinen saldırılara karşı yüksek başarı hemde yeni karşılaşılan saldırı türlerine karşı tespit edebilme yeteneği kazanılmıştır [30].

3.4.2 *KAA* uygulamalarında saldırı tespiti

KAA' ların belli başlı özelliklerinden dolay *STS* tasarımı kablolu ve kısıtsız ağ yapısından farklıdır. *KAA* için bir *STS* dizayn etmek belirli problemlere sahiptir. Söz konusu problemler mutlaka hesaba katılmalıdır. *KAA* binlerce düğümden oluşabilir ve bu durum *KAA* için bir yönetim zorluğu meydana getirmektedir. Çok sayıda algılayıcı olan bir *KAA* yapısında hata hoşgörü mekanizmasını işletmek çok zor bir

işlemdir ancak çözülmesi gereken hayati bir konudur. Algılayıcı düğümler harici bir enerji kaynağına sahip değildirler. Tüm enerjilerini kendi dahili kaynaklarından sağlarlar.Ağ işletim zamanını arttırmak için enerji tüketimini minimumlarda tutmak gereklidir. Sistem yöneticileri sistemde fazladan işlem gerçekleşmesini istemezler. Bu nedenle enerji tüketimini minimumda tutacak uygulamalar için çalışmalar devam etmektedir. Zaman kısıtı da oldukça önemlidir. Bazı uygulamalar için çalışmalarında zaman kriteri bulunmaktadır. Belirli uygulamalarda ver eğer ki gerekli zamanda elde edilemezse sistem için herhangi birdeğeri bulunmamaktadır. Bu nedenle algılayıcı düğümün fazladan yapacağı işlemler sistemi zaman konusunda sıkıntıya düşürebilmektedir ve bu konuda mutlaka dikkate alınması gereken başlıklar arasındadır. *KAA*' lar farklı topolojilere sahip olabilirler bu nedenle sabit bir *STS* uygulaması gerçekleştirmek mümkün olmayabilir. Oluşturulan *STS* yapısı kendi organizasyonel yapısını kurabilmelidir. *KAA* dahilinde bulunan algılayıcı düğümler geniş bir alan üzerinde düzensiz olarak dağılmış olabilirler hatta açık bir alan üzerinde korumasız olarak rakip ortamında bile bulunabilir. Bu problem de yine diğerleri gibi mutlaka göz önünde bulundurulmalıdır [32]. KAA' ların sahip olduğu kısıtlar nedeniyle klasik yaklaşımlarda kullanılan *STS*' le uygun olmamaktadır. Örneğin; imza tabanlı bir *STS* yapısında saldırı imzalarının bir bilgi tabanı üzerinde tutulması üzerinde tutulması gerekmektedir. Ancak, söz konusu imza bilgi tabanının herhangi bir algılayıcı düğüm üzerinde tahsis etmek imkansızdır. Düğümlerin sahip oldukları kısıtlı depolama ve işlem kapasiteleri bu şekilde ağır bir yükü kaldıramazlar. Bu yaklaşım sistemin asıl kullanım amacına ters düşmektedir. En önemli konulardan biri de algılayıcı düğümlerin tek başlarına güvenilir olmaması ve sıklıkla hatalar göstermesidir. Ancak oluşturulan sistemin genel yapısı ile sistem güvenilir ve hata hoşgörülü bir yapı kazandırılmalıdır. Kablolu ağlara göre farklılıklar bu bölümde ortaya çıkarılmıştır. Sınıflandırma saldırı tipine, saldırgan tipine, tespit tekniğine, elde edilen verinin kaynağına, datanın analiz edildiği yere göre yapılmıştır ve literatürde de bu şekilde açıklanmıştır [33].

- **Saldırgan tipi :** Bir ağ yapısında saldırgan tipleri iki gruba ayrılmıştır. Dahili saldırganlar ve harici saldırganlar.

- **Saldırı tipi :**Saldırı tipine göre verinin çalınması, yanlış veri üretilmesi, sisteme erişimin engellenmesi, enerji tüketiminin arttırılması.

- **Tespit yöntemi :** Tespit metedolojisine göre yukarıda anlatıldığı üzere anormallik tespiti, kötüye kullanım tespiti ve literatüre göre ikisinin birleşimi olarak gösterilen karma yöntem. Herhangi bir durumda kural ihlali olursa sistem bir anormallik olduğuna karar verir [33].

- **Ham verinin toplanma yeri :** Verinin elde edildiği noktaya göre ise ağ tabanlı, sunucu tabanlı ve karma tabanlı olarak gruplandırılır.

- **Ağ dizaynı :** KAA ağ dizaynı hiyerarşik ve kümelenmiş olarak gerçekleştirilebilir. Bu durumda *STS* yaklaşımları da farklı olacaktır [34] [35].

- **Kullanım periyodu :** Sistemin sürekli veya periyodik olarak çalıştırılmasını ifade etmektedir.

- **Toplanan verinin analiz edilme yeri :** Verinin işlendiği noktaya göre ise merkezi ve dağıtılmış yaklaşımlar söz konusudur [36].

KAA' lar için geliştirilen *STS* mekanizmalarının öznitelikleri sistemin sahip olduğu ihtiyaçlara göre değişiklik gösterebilirler. Örneğin *KAA* kontrollü bir bölgede gerçekleştirilmiş ve sistem yöneticisi istediği her durumda ağa müdahele edebilecek ve gerekli değişiklikleri yapabilecek durumda ise bazı kısıtlar gözardı edilebilir. Tüm bunların yanında sistemin ana hedefi kısıtlı enerji, işlemci ve depolama kaynaklarını verimli kullanmak ve gerçek zamanlı veri akışını sağlayabilmektir.

KAA uygulamalarında saldırıları tespit etme tekniğine göre incelendiğinde Anormallik Tespiti ve Kötüye Kullanım Tespiti yaklaşımları şu şekilde incelenebilir:

- ***KAA* uygulamalarında kötüye kullanım tespiti**

  Kötüye Kullanım Tespiti yaklaşımı için bilinen saldırılar ışığında kurallar oluşturulduğundan ve bu kurallar ile saldırıların tespit edilmeye çalışıldığından daha önce bahsedilmişti. Bu yaklaşım *KAA* açısından düşünüldüğünde çok verimli olmayacağı rahatlıkla görülebilir çünkü *KAA* algılayıcı düğümlerinin sahip olduğu sınırlı kaynakların söz konusu kuralları saklayabileceği alanları bulunmamaktadır,

eldeki kaynakların da bu kuralları saklamak için harcanması *KAA* yaklaşımının doğasına ters düşmektedir. Bu bakımdan *KAA* uygulaması için tahsis edilecek bir *STS* mekanizmasında Kötüye Kullanım Tespiti tekniğinin kurallarının algılayıcı düğüm üzerinde saklanması yoluyla kullanılması uygun olmayacaktır.

- ***KAA* uygulamalarında anormallik tespiti**

  KAA için planlanan güvenlik mekanizmaları için gerçekleştirilen çalışmalarda daha çok Anormallik Tespiti üzerinde yoğunlaşılmaktadır. Ancak yine Kötüye Kullanım Tespitinde olduğu gibi Anormallik Tespiti yaklaşımında da normal durumların belirlenmesi ve saklanması konusunda kaynak kısıtları bulunmaktadır. Önemli olan kullanılan tekniğin kısıtlı kaynakları optimum seviyenin üzerine çıkmadan, sistem amaçlarını etkilemeden çalışabilmesidir. *KAA* uygulamaları için takip edilen sistem anormallikleri aşağıdaki gibidir [25]:

  - **Ağ anormallikleri:** KAA üzerinde oluşan bağlantı problemleri ile ortaya çıkmaktadır. Aniden artan veya azalan sinyal kalitesi anormallik olup olmadığını anlaşılmasına yardımcı olur. Sinyal kaybı, Kesik kesik sinyal edinimi, döngü tespiti ve genel yaylımlar anormallik işaretleridir.
  - **Algılayıcı düğüm anormallikleri:** Yazılım ve donanım problemleridir. Düğümün enerjisinin tükenmesi de bu kategoriye dahil edilebilir.
  - **Veri anormallikleri:** Elde edilen verilerde görülen bozulmalar paket boyutlarındaki değişikliklerdir.
  - **Diğer anormallikler:** Yukarıda sıralanan başlıklar altına doldurulamayan karşılaşılabilecek anormalliklerdir.

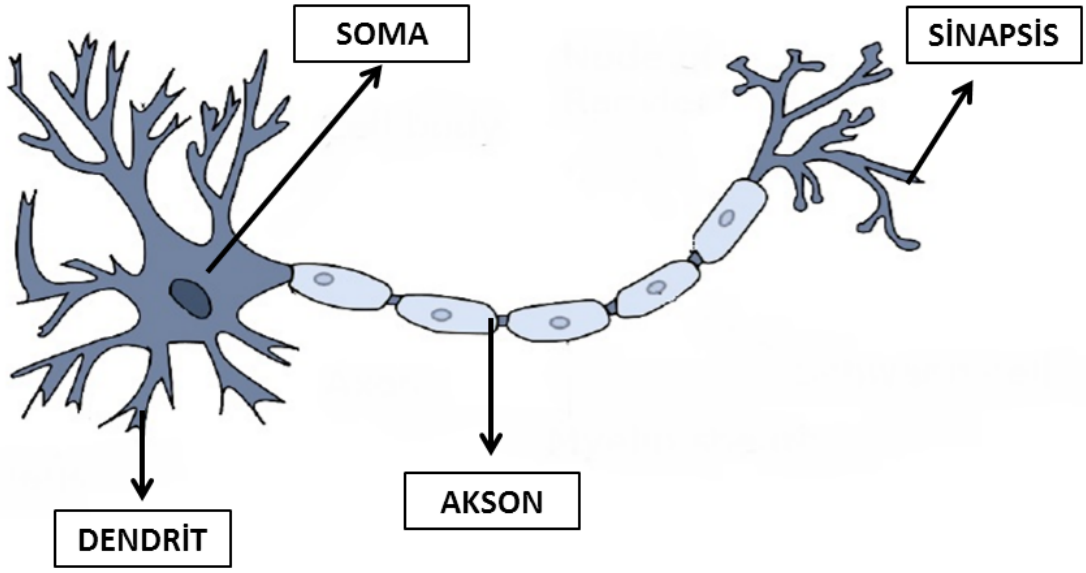
4. SALDIRI TESPİTİ YAKLAŞIMI OLARAK YAPAY SİNİR AĞLARI

Yapay sinir ağları bir algoritma ile formülize edilemeyen problemlerin çözümünde kullanılmak üzere ortaya atılan bir yaklaşımdır. Bu noktada bir soru ortaya çıkmaktadır. "İnsanlar problem çözmeyi nasıl öğrenir?". İnsanlar öğrenme işlemini beyinleri ile gerçekleştirirler ve bilgisayarlar da işlemcilere ve depolama aygıtlarına sahiptirler. Yapay sinir ağları ile insanların öğrenme yeteneklerini bilgisayar ortamında simule etmek amaçlanmaktadır [37] [31]. Yapay sinir ağı konsepti yazılım olarak hayata geçirilebileceği gibi donanım olarak da geliştirilebilir. Yazılım şekliyle daha kolay bir şekilde geliştirilen yapay sinir ağı yaklaşımı donanım olarak gerçeklendiğinde daha hızlı çalışmaktadır. Yapay sinir ağı yaklaşımının temelinde bilgisayarların da insanlar gibi öğrenebilme yeteneklerine sahip olabilmesi amaçlanmış bu yönde yola çıkılmıştır. Bilgisayarlar çok büyük sayyılarla çok fazla işlemi hızlıca yapabilirler ancak uyum sağlama yani öğrenebilme yetenekleri yoktur. Kullanıcılarının kendilerine verdikleri ölçüsünde işlemlerine devam edebilirler. Bilgisayarlara yeni durumlara adapte olabilme yeteneğinin kazandırılması yapay sinir ağları konseptininin temelini oluşturmaktadır. bilgisayar ile insan beyni karşılaştırılması Çizelge 4.1 üzerinde güzel bir şekilde gösterilmiştir.

Çizelge 4.1: Bilgisayar ile insan beyninin karşılaştırılması.

	Beyin	**Bilgisayar**
İşlemci Birimleri Sayısı	$\approx 10^{11}$	$\approx 10^{9}$
İşlemci Birimi Tipi	Nöron	Transistör
Hesaplama Tipi	Kitlesel Olarak Paralel	Genellikle Seri
Veri Depolama	Çağrışımlı	Adres Tabanlı
Akım Geçiş Zamanı	$\approx 10^{-3}s$	$\approx 10^{-9}s$
Mümkün Olan Akım Geçişi İşlemi	$\approx 10^{13}\frac{1}{s}$	$\approx 10^{18}\frac{1}{s}$
Gerçek Olan Akım Geçişi İşlemi	$\approx 10^{12}\frac{1}{s}$	$\approx 10^{10}\frac{1}{s}$

Teorik olarak incelendiğinde bilgisayar insan beyninden daha güçlü görünmektedir. Bilgisayarlar 10^9 transistöre sahiptirler ve bu transistörler 10^{-9} saniyelik akım geçiş zamanına sahiptir. İnsan beyni ise 10^9 tane nörona sahiptir fakat bu



Şekil 4.1: Biyolojik sinir hücresinin yapısı.

nöronlar sadece 10^{-3} saniyelik akım geçiş zamanına sahiptir. Ancak teorinin dışına çıkıldığında görülecektir ki insan beyninin büyük bir bölümü sürekli olarak çalışırken bilgisayarların büyük bir bölümü depolama alanları olarak kullanılmaktadır. Beyin paralel olarak çalışır ve bu sayede kendi teorik maksimumuna yaklaşarak çalışabilir ancak aynı durum bilgisayarlar için geçerli değildir. Bu durumun rakamlarla ifade edilen hali Çizelge 4.1 üzerinden görülebilir. Tüm bunlara ek olarak bilgisayarlar statiktir fakat insan beyni biyolojik sinir ağı sayesinde ömrü boyunca karşılaştığı yeni durumları tanımlayabilir ve dolayısı ile ***"öğrenebilir."*** Bilgisyar sistemlerinin insan beyni gib çalışabilmesini sağlamak için biyolojik gerçeklerden aşağıdaki durumların transfer edilebilmesi amaçlanmıştır.

- Öz organizasyon ve öğrenebilme yeteneği
- Genelleştirebilme yeteneği
- Hata hoşgörü yeteneği

Yapay sinir ağı yaklaşımını daha iyi anlayabilmek için biyolojik sinir ağının da incelenmesi gereklidir. Bu sayede biyolojik olgudan bilgisayar ortamına transfer edilmesi hedeflenen özellikler daha iyi anlaşılabilecektir.

Şekil 4.1 üzerinden de görülebileceği üzere biyolojik sinir ağının temel yapısı olan sinir hücresi belirli bölümlerden oluşmaktadır. Bu bölümler şu şekilde açıklanabilir [37].

- **Dendrit:** Dendritler sinir hücresi bünyesinde bir önceki nörondan gelen elektromanyetik uyarının yine sinir hücresinin bir parçası olan somaya ileten dal benzeri yapılardır. Sinir hücresinin özelliğine göre sayıları değişmektedir. Nöronların algı işlevlerini yerine getiren bölgeler olarak da adlandırılmaktadırlar. Dentritlerin dağılımına göre kendisine gelen elektriksel uyarılardan zayıf olanları iletmediği görülümüştür.

- **Soma:** Kendisine bağlı dendritler aracılığı ile gelen verilerin toplanması ile görevli hücre çekirdeğidir. Tüm verileri bir araya getirerek sonraki nörona iletebilmek amacıyla verileri topladıktan sonra aksona iletir. Sinir hücresinde veri akışı tek yönlü ve somadan aksona doğru gerçekleştirlmektedir.

- **Akson:** Sinir hücrelerinin temel kolu olarak nitelenirler. Dendritlerden gelen ve soma tarafından bir araya getirilen verileri ön işemden geçirerek sinapsislere iletmekten görevlidir.

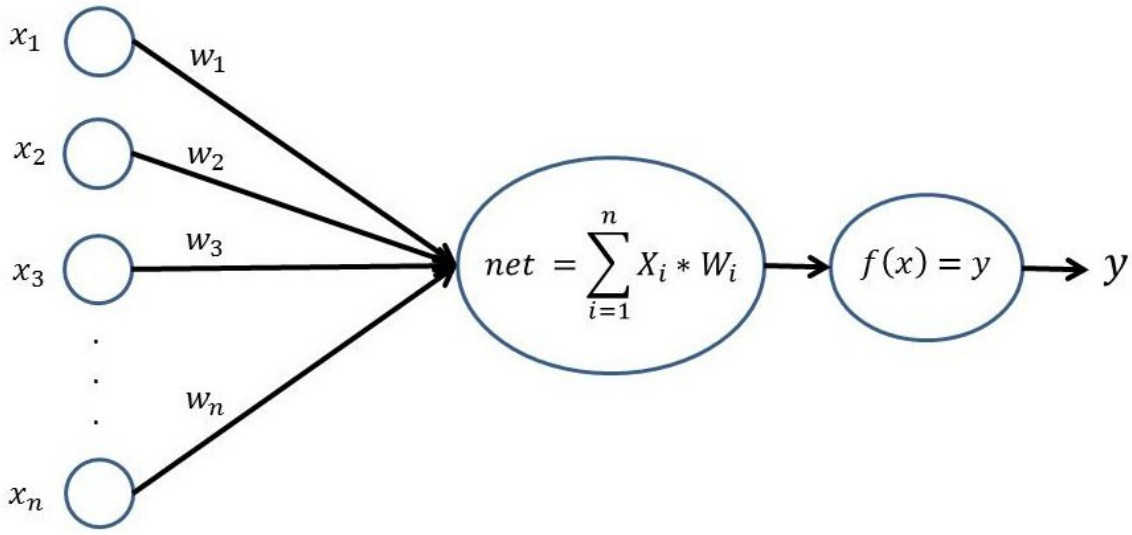
- **Sinaps:** Sinapslar sinir hücrelerinin birbirleriyle veya diğer hücreler ile iletişimini sağlayan ara boşluklardır. Sinapslar sayesinde nöronlar birbirleri ile bir ağ yapısı oluşturabilirler. Bu noktada elektrik sinyalleri kimyasal sinyaller haline dönüşür ki bu bir önişlem olarak nitelenebilir.

Bir organizmanın karmaşıklık ve nöron sayısı arasındaki bağlantı şu şekilde sıralanabilir:

- Biyoloji dünyasında çok iyi bilinen nematod solucanı için 302 nöron yeterlidir.

- 10^4 nöron karıncaların sahip olduğu sayıdır ki karıncaların sahip olduğu sosyal davranışlar çok iyi bilinmekte ve en iyileme problemlerinde çözüm olarak karşımıza çıkmaktadır.

- Bir sineğin sahip olduğu nöron sayısı 10^5 tir. Sinekler çok iyi kaçınma manevraları yapabilen öz savunmaları bu yönde gelişmiş canlılardır.



Şekil 4.2: Yapay sinir hücresinin yapısı.

- Kedi gibi zarif, akıllı bir hayvanın sinir sistemini oluşturmak için 3×10^8 tane nöron gereklidir.
- Sinir sistemindeki 6×10^9 adet nöronla insanlara en yakın olan canlı maymunlardır ve öğrenme, taklit, sosyal ilişki açısından insana en yakın canlılardır.
- İnsanların sinir sisteminde ise toplamda 10^{11} nöron bulunmaktadır.

4.1 Yapay Sinir Ağı Yapısı Ve Bileşenleri

Biyolojik sinir ağı hücresinin yapısı Şekil 4.1 üzerinde görülebilir. Bu biyolojik gerçekten ilham alarak modellenen yapay sinir ağı yapısı ise Şekil 4.2 üzerinden görülebilir. Yapay sinir ağı üzerinde bulunan $x_1, x_2, x_3 ... x_n$ şeklinde gösterilen ve $w_1, w_2, w_3 ... w_n$ şeklinde ağırlıklara sahip girdiler dendritleri temsil ederken tüm bu girdilerden gelen verileri bir araya getiren fonksiyon soma (çekirdek) görevini üstlenmektedir. Toplanarak tek bir değer halini alan veriler Aktivasyon Fonksiyonu (Eşikleme Fonksiyonu) ile işlenir ve nöronun mevcut durumunun değişip değişmeyeceğine karar verilir.

Teknik olarak bir Yapay Sinir Ağı nöronlardan ve bu nöronlarını birbirine bağlayan ağırlıklaştırılmış direk bağlantılardan oluşmaktadır. Şekil 4.2 ile gösterilen yapay sinir hücrelerinin bir araya getirdikleri yapı ise Yapay Sinir Ağı' nı oluşturmaktadır. Ağı eğitirken verilen girdiler ile her iterasyonda değerler hesaplanır ve bu değerler

durağanlaşında ve çıkan değerler ile olması gereken sonuçlar arasındaki hata farkı belirli bir değerin altına düştüğü zaman öğrenmenin olduğu kabul edilir ve eğitim sonlandırılır.

4.1.1 Girdiler

Yapay Sinir Ağı konseptinde bulunan yapay sinir hücrelerinden bir önceki nöronun çıktısı bir sonraki nöron için girdi konumundadır. Girdiler Şekil 4.2 üzerinde $x_1, x_2, x_3 ... x_n$ şeklinde gösterilmiştir.

4.1.2 Propogasyon fonksiyonu

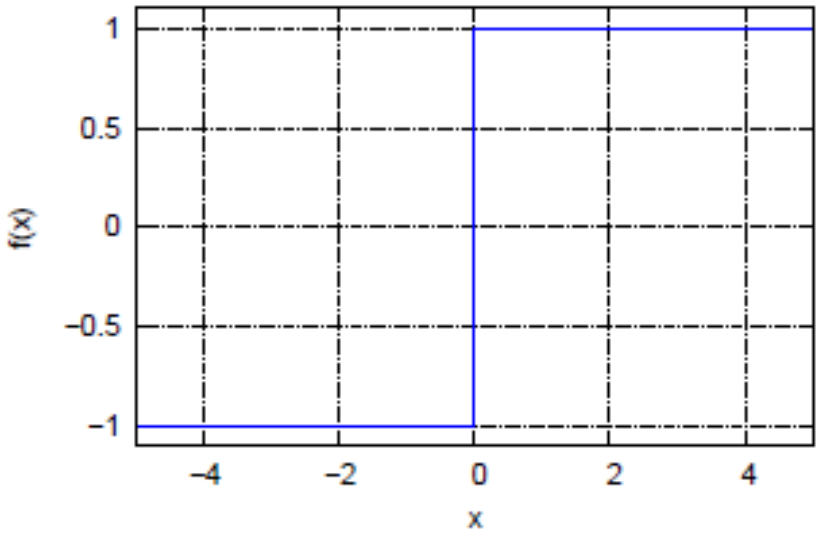
j nöronundaki bir *Propogasyon Fonksiyonu* kendisinden önceki i nöronlarının $(i_1, i_2, i_3 ... i_n)$ o çıktılarını $(O_1, O_2, O_3 ... x_n)$ x girdisi $(x_1, x_2, x_3 ... x_n)$ olarak alır ve bu *(x)* girdilerini w ağırlıklarıyla ilişkilendirerek bulunduğu nöronun *Aktivasyon Fonksiyonu* için net girdiyi *net* hesaplar. Yapay sinir hücresi için asıl girdi Eşitlik **4.1** formülü ile gösterilen *Propogasyon Fonksiyonu* ile hesaplanır. Propogasyon fonksiyonunu işleme koymak noktasında en yoğun olarak Eşitlik **4.2** ile gösterilen *Ağırlıklaştırılmış Toplam* formülü kullanılmaktadır.

$$net_j = f_{prop}(O_{i_1}, ..., O_{i_n}, w_{i_1,j}, ... w_{i_n,j}) \tag{4.1}$$

$$net_J = \Sigma(O_i . w_{i,j}) \tag{4.2}$$

4.1.3 Aktivasyon (Eşikleme) fonksiyonu

Aktivasyon bir nöronun değişme (adapte olma) durumunu göstermektedir. Nöronların kendisine gelen girdilere karşı göstereceği reaksiyonlar mevcut aktivasyon durumuna bağlıdır. j nöronu için aktivasyon durumu aj Aktivasyon Fonksiyonunun sonucunda ortaya çıkan durumdur ve nöronun ne yapacağını gösterir [37]. Nöronun aktivasyon durumuna karar verme aşımasında nöronun ***Eşik Değeri*** kullanılır. Eşik değeri θ ile gösterilmektedir. Bu durumda *Aktivasyon Fonksiyonu* Eşitlik **4.3** formülü ile tanımlanmaktadır.



Şekil 4.3: Heaviside fonksiyonu.

$$a_j(t) = f_{act}(net_j(t), a_j(t-1)\Theta_j \tag{4.3}$$

Bir nöronun aktivasyon durumu bir önceki aktivasyon durumuna $a_j(t-1)$ ve o anki girdilere $net_j(t)$ bağlıdır. Eşik değeri olan θ' nın da önemi Eşitlik **4.3** üzerinden görülebilir. Bu eşitlikte *(t)* nöronun o anki durumunu ve *(t-1)* ise önceki durumunu temsil etmektedir. *Aktivasyon Fonksiyonu* bazı kaynaklarda ***Transfer Fonksiyonu*** olarak da adlandırılmaktadır. Genel Aktivasyon Fonksiyonları şunlardır.

- **Heaviside fonksiyonu**

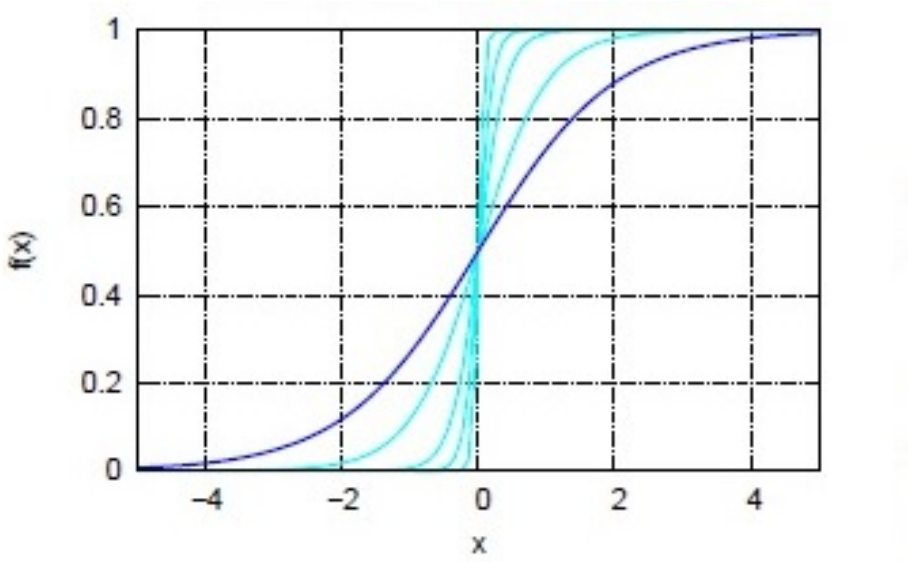
  Girdi eşik değerinin üstünde ise Heaviside Fonksiyonu değeri değiştirir aksi taldirde sabit tutar. Bu şu anlama gelir ki Heaviside Fonksiyonu eşik değerinde türevi alınabilir değildir ve türevinden kalan 0' dır. Heaviside Fonksiyonu Şekil 4.3 üzerinden görülebilir.

- **Fermi fonksiyonu & Lojistik fonksiyonu**

  Fermi fonksiyonu (Lojistik fonksiyonu) uygulandığı değerleri (0,1) arasına indirger. Fermi fonksiyonu Eşitlik **4.4** ile gösterilmiştir.

$$\frac{1}{1+e^{-x}} \tag{4.4}$$

  Fermi fonksiyonunda T (Temperature) değişkeni eklendiğinde ise fonksiyon Eşitlik **4.5** 'da gösterildiği gibi olur



Şekil 4.4: T parametreli fermi fonksiyonu.

$$\frac{1}{1+e^{-x/T}} \tag{4.5}$$

Şekil 4.4 üzerinde Fermi Fonksiyonuna T (Temperature) değişkeninin eklenmiş hali görülebilir. Grafikte orjinal Fermi Fonksiyonu koyu mavi olarak ve sırasıyla $\frac{1}{2}$, $\frac{1}{5}$, $\frac{1}{10}$,$\frac{1}{25}$ değerlerini almış T değişkeni ile oluşan şekiller ise açık mavi olarak gösterilmiştir.

- **Hiperbolik tanjant fonksiyonu**

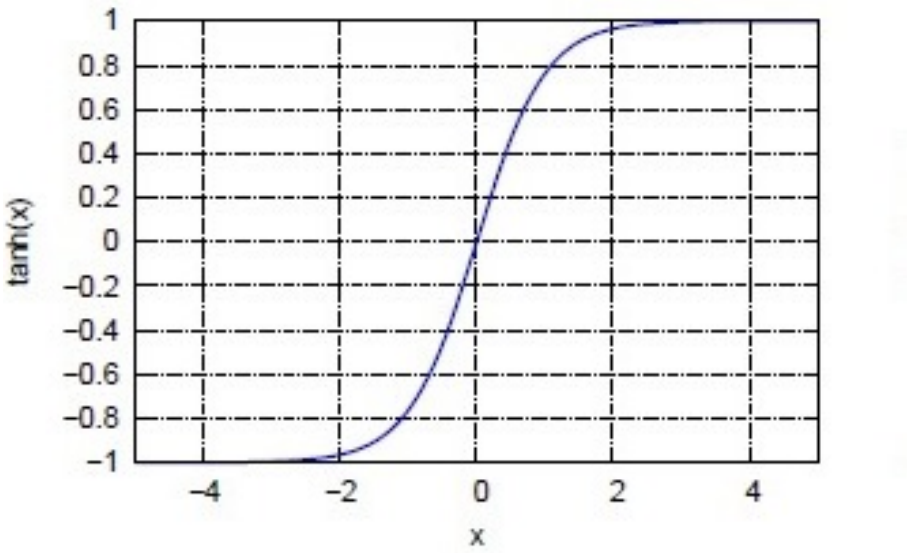
  Hiperbolik Tanjant Fonksiyonu Şekil 4.5 ile gösterilmiştir. Hiperbolik Tanjant Fonksiyonu uygulandığı değerleri (-1,1) arasına indirger.

4.1.4 Çıktılar

Çıktılar yapay nörondan elde edilen *(o)* değerdir. Yapay sinir ağının nihai sonucu değil ise kendisinden sonraki nöron için girdi *(x)* niteliğindedir.

4.2 Yapay Sinir Ağı Öğrenme Yaklaşımı

Öğrenme işlemi adapte olma değişiklik yapma işlemleri ile açıklanabilir. Örneğin sürekli öğrenme aşamasında olan bir bebeğin sıcaklık ile ilgili olan öğrenmesi o



Şekil 4.5: Hiperbolik tanjant fonksiyonu.

sıcaklık ile karşılaşması, zarar görmesi ve daha sonraki zamanlarda davranışlarında veya tehlike anlayışında değişiklik yapması ile gerçekleşir. Tehlike anlayışındaki değişiklik bebeğin artık ateşten uzak durma güdüsü ile karşılanır. Yapay sinir ağları yaklaşımında da benzer şekilde öğrenmenin tanımlanması gerekir. Bu kapsamda öğrenmenin ilk adımı aktivasyon olarak nitelenebilir. Bir yapay nörona gelen girdilerin değerleri o hücreyi aktif hale getirebilirse öğrenme olgusu basitçe tanımlanmış olur. Eşik değeri ile aktif olma veya olmamama durumlarına geçebilen bir nöron aslında *"karar verme"* yeteneğini kazanmış sayılır.

Öğrenme kapsamlı bir işlemdir. Öğrenen bir sistem öğrenme işlemini kendi adaptasyonu ve çevresel değişiklikler gibi etkenler sayesinde gerçekleştirmektedir. Temelde sinir ağı sağlamak istediği öğrenmeyi kendi bileşenlerini değiştirerek sağlamaktadır. Yapay sinir ağı bileşenlerinden daha önce bahsedilmişti ve bu olası değişiklikler şu şekilde sıralanabilir [37]:

- Yeni bağlantılar kurarak.
- Mevcut bağlantılardan bazılarını silerek.
- Bağlantı ağırlıklarını değiştirerek.
- Nöronların eşik değerlerini değiştirerek.
- Yeni nöronlar ekleyerek.

- Mevcut nöronlardan bazılarını silerek
- Nöron içinde kullanılan Propogasyon ve Aktivasyon Fonksiyonlarını değiştirerek.

Değişiklik yapılarak öğrenmeyi sağlayacak yukarıdaki işlemlerden en yoğun olarak kullanılanı bağlantı ağırlıklarını değiştirmektir. Bu öğrenme yaklaşımı şu şekilde örneklenebilir.

$i_1 = 6, i_2 = 10, i_3 = 14, i_4 = 4, i_5 = 16$

Başlangıç ağırlıkları şu şekilde olsun:

$w_1 = 0.3, w_2 = 0.3, w_3 = 0.3, w_4 = 0.3, w_5 = 0.3$

Bir sonraki iterasyonda girdiler şu şekilde olur:

$i_1 = 7, i_2 = 17, i_3 = 5, i_4 = 8, i_5 = 16$

Girdilerde gerçekleşen değişimlere göre ağırlıklar güncellenir.

$w_1 = 0.35, w_2 = 0.45, w_3 = 0.25, w_4 = 0.45, w_5 = 0.3$

Her iki iterasyonda da aktivasyon fonksiyonunun sonucunda eşik değeri geçilirse sistem ağırlıkları kendilerini bu değerlere göre adaptasyonlarını sağlarlar. Bu sayede öğrenme gerçekleşmiş olur.

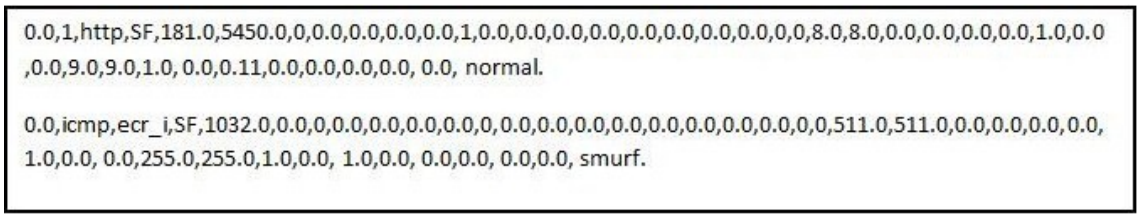
Yapay sinir ağları öğrenme stratejileri açısından danışmanlı öğrenme, danışmansız öğrenme ve destekleyici öğrenme olarak üçe ayrılmaktadır. Ayrıca YSA' lar öğrenme zamanlarına göre de online öğrenme ve offline öğrenme yapısıyla karşımıza çıkarlar. Askeri savunma teknolojileri, finans, sağlık, üretim, otomotiv, sivil güvenlik, telekominikasyon teknolojileri YSA' ların kullanıldığı alanlardan birkaçıdır

5. ÖNERİLEN SİSTEM VE DENEYSEL SONUÇLAR

Tasarlanan sistem ile KDD Cup 99 veri seti ile eğitilen bir Yapay Sinir Ağı ile *KAA* üzerinde bir Saldırı Tespit Sistemi *STS* gerçeklenmesi amaçlanmıştır. Bu bölümde söz konusu çalışma ile ilgili olarak neler yapıldığı ve nihayetinde nasıl deneysel sonuçların elde edildiği anlatılacaktır.

5.1 KDD 99 Cup Veri Seti

KDD Cup 99 tespiti için en çok kullanılan veri setidir. Konu ile ilgili olan birço çalışma KDD 99 Cup veri seti ile gerçekleştirilmiştir ve bu sayede deneysel sonuçların karşılaştırılabilmesi yeteneği kazanılmak istenmiştir. Bu nedenle KDD 99 Cup veri seti tercih edilmiştir. Bu veri seti DARPA'98 *STS* hesaplama programından elde edilen veriler üzerine inşa edilmiştir. DARPA'98 herbiri yaklaşık 100 byte olan yaklaşık beş milyon bağlantı kaydıiçeren yaklaşık 4 gigabyte boyutunda 7 haftalık ağ trafiğine ait sıkıştırılmış binary tcpdump verilerinden oluşmaktadır. KDD eğitim seti 41 tane farklı özniteliği bulunan ve herbiri "saldırı" veya "normal" olarak etiketlenmiş 4,900,000 adet tekil bağlantı vektöründen oluşmaktadır. İki adet tekil bağlantı vektörü örneği Şekil 4.5 ile gösterilmiştir. Görüldüğü üzere ilk vektör "normal" ikinci vektör ise "smurf" olarak etiketlenmiştir.



Şekil 5.1: KDD Cup 99 veri seti örnek bağlantı vektörü.

Her bir bağlantı vektörü ağ üzerinde gerçekleştirilen istemci / sağlayıcı ilişkisini göstermekte ve her bir bağlantı isteğinde bulunan paketin boyutu, kaçıncı paket olduğu, kaynak adresi ve hedef adresi, kulanılacak port numarası, istek sayısı, hatalı istek sayısı

gibi özniteliklerini sırası ile sergilemektedir. Bu öznitelikler yorumlanarak ve bağlantı vektörünün etiketi ile birleştirerek kullanılması sağlanmıştır.

KDD Cup 99 veri setinin her bir tekil bağlantı vektörüne ait 41 adet öznitelik Çizelge 5.1 ile gösterilmiştir.

Çizelge 5.1: KDD 99 Cup tekil bağlantı vektörü öznitelikleri

	Öznitelik		Öznitelik
1	duration	22	is_guest_login
2	protocol_type	23	count
3	service	24	srv_count
4	flag	25	serror_rate
5	src_bytes	26	srv _serror _rate
6	dst_bytes	27	rerror _rate
7	land	28	srv _rerror _rate
8	wrong_fragment	29	same _srv _rate
9	urgent	30	diff _srv _rate
10	hot	31	srv _diff _host _rate
11	num _failed _logins	32	dst _host _count
12	logged _in	33	dst _host _srv _count
13	num _compromised	34	dst _host _same _srv
14	root_shell	35	dst_host _diff _srv
15	su_attemted	36	dst _host _same _src_port
16	num _root	37	dst _host _srv _diff_host
17	num _file _creations	38	dst _host _serror
18	num _shells	39	dst _host _srv _serror
19	num_access _file	40	dst _host _rerror
20	num _outbound _cmds	41	dst _host _srv _rerror
21	is _host _login	42	attack _type

Kullanılan veri setinin simule ettiği saldırı tipleri dört ana başlık altında toplanabilir ve bu başlıklar şu şekilde sıralanabilir:

- **Hizmet Engelleme (DOS):** Hizmet engelleme saldırılarının temel amacı sistemi cevapveremez hale getirmektir. KAA' lar özellikleri gereği farklı hizmet engelleme saldırılarına konu olabilir.
- **Yönetici Hesabı ile Yerel Oturum Açma (Remote toLocal - R2L):** Kullanıcı haklarına sahip olunmadığı durumda misafir ya da başka bir kullanıcı olarak izinsiz erişim yapılmasıdır.

- **Kullanıcı Hesabının Yönetici Hesabına Yükseltilmesi(User to root - U2R):** Bu tip saldırılarda sisteme girme izni olan fakat yönetici olmayan bir kullanıcının yönetici izni gerektirecek işler yapmaya çalışmasıdır.
- **Bilgi Tarama (Probe ya da scan):** Belirli bir portu sürekli taranması (ipsweep) veya bir sunucu üzerindeki hizmetleri bulmak için tüm portların taranması (portsweep).

Bu çalışmada KDD'99 veri setinin % 10 luk kısmı olan ve 494.000 tane kayıt içeren **"kddcup.data_10_percent.gz"**kullanılmıştır. Literatürde gerçekleştirilen çalışmaların bu alt veri setini kullandıkları görülmüştür. Başarı oranlarının karşılaştırılabilmesi amacıyla aynı alt küme kullanılmıştır. Kullanılan veri setinin sahip olduğu saldırı tipleri ve her birinin kayıt örnek sayıları Çizelge 5.2 üzerinde gösterilmiştir.

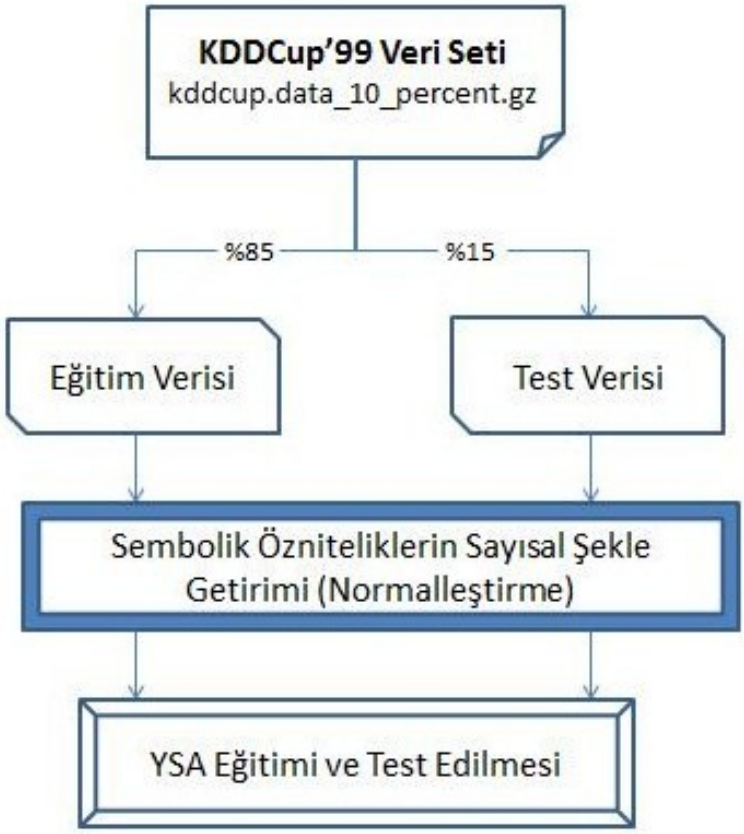
Çizelge 5.2: % 10 KDD' 99 saldırı tip ve sayıları

Saldırı	Örnek	Kategori	Normalleştirme
Smurf	280790	DOS	1
Neptune	107201	DOS	2
Back	2203	DOS	3
Teardrop	979	DOS	4
Pod	264	DOS	5
Land	21	DOS	6
Normal	97277	NORMAL	7
Satan	1589	PROBE	8
IpSweep	1247	PROBE	9
PortSweep	1040	PROBE	10
Nmap	231	PROBE	11
WarezClient	1020	R2L	12
Guess_Password	53	R2L	13
WarezMaster	20	R2L	14
Imap	12	R2L	15
FTP _Write	8	R2L	16
Multihop	7	R2L	17
Phf	4	R2L	18
Spy	2	R2L	19
Buffer _Overflow	30	U2R	20
RootKit	10	U2R	21
LoadModule	9	U2R	22
Perl	2	U2R	23

5.2 Akış Şeması

Yaklaşık 494.000 adet kayıt içeren kddcup.data _10 _percent.gz veri setinin %85' lik kısmı YSA' nın eğitiminde geriye kalan %15' lik kısım ise eğitilen Yapay Sinir Ağı' nın test edilmesinde kullanılmıştır. Oluşturulan yapı Şekil 5.2 üzerinde gösterilmiştir.

KDDCup'99 veri setinin sayısal özniteliklerinin yanında sembolik öznitelikleri de (Service, Protocol, Flag, Label) bulunmaktadır. Bu özniteliklerin YSA eğitimi ve testi işlemlerinde kullanılabilmeleri için sayısal bir formata kavuşturulmaları gerekmektedir. Gerekli format için Şekil 5.2 üzerinden görüleceği üzere normalizasyon işlemleri gerçekleştirilmiştir.



Şekil 5.2: Akış şeması.

5.3 Normalizasyon İşlemleri Ve Öznitelik Seçimi

Başarılı bir sistem için öznitelik seçimi çok önemlidir. KDD içinde bazı öznitelikler KAA için uygun değildir ve etkinliği yoktur. Bu öznitelikleri çıkarılması işlemi öznitelik seçimidir. Bu çalışmada Shirazi [38] tarafından kullanılan seçim yapılmış

ve öznitelikler 25' e indirilmiştir. Seçilen öznitelikler Çizelge 5.3 ile gösterilmiştir. Böylelikle yerel minimumlardan sakınmak amaçlanmıştır [38] [39].

Çizelge 5.3: Seçilen 22 adet öznitelik.

Seçilen Öznitelikler		
Sıra No	Seçilen Öznitelik No	Seçilen Önitelik Adı
1	1	duration
2	2	protocol_type
3	3	service
4	4	flag
5	5	src_bytes
6	6	dst_bytes
7	10	hot
8	12	logged _in
9	14	root_shell
10	17	num _file
11	22	is_guest_login
12	23	count
13	24	srv_count
14	27	rerror _rate
15	29	same _srv _rate
16	30	diff _srv _rate
17	32	dst _host _count
18	33	dst _host _srv _count
19	35	dst_host _diff _srv
20	36	dst _host _same _src_port
21	37	dst _host _srv _diff_host
22	41	dst _host _srv _rerror

KDD eğitim seti' nin 41 tane farklı özniteliği bulunan ve herbiri "saldırı" veya "normal" olarak etiketlenmiş 4,900,000 adet tekil bağlantı vektöründen oluştuğundan ve bu 41 öznitelik ve bir adet etiket karşılığının tamamının sayısal değer olmadığından bahsedilmişti. Karşılığı sayısal olmayan bu öznitelikler saldırı tipi, protokol servis ve bayrak özniteliklerinin sayısal forma dönüştürülmesi gerekmektedir [39]. Saldırı tipinin dönüştürülmesi Çizelge 5.2 ile, protokol özniteliğinin dönüştürülmesi Çizelge 5.5 ile, servis özniteliğinin dönüştürülmesi Çizelge 5.4 ile ve bayrak özniteliğinin dönüştürülmesi Çizelge 5.6ile gösterilmiştir. Gerçekleştirilen seçim işlemleri ile birlikte *YSA* eğitiminde ve matematiksel hesaplamalarda kolaylık sağlanması amaçlanmıştır.

Çizelge 5.4: Servis öznitelikleri normalizasyonu.

Servis Özniteliği Normalizasyonu					
Servis	Num.	Servis	Num.	Servis	Num.
AUTH	1	irc	22	printer	43
BGP	2	imap4	23	private	44
courier	3	iso_tsap	24	red_i	45
csnet_ns	4	klogin	25	remote_job	46
ctf	5	kshell	26	rje	47
daytime	6	ldap	27	shell	48
discard	7	link	28	smtp	49
domain	8	login	29	sql_net	50
domain_u	9	mtp	30	ssh	51
echo	10	name	31	sunrpc	52
eco_i	11	netbios_dgm	32	supdup	53
ecr_i	12	netbios_ns	33	systat	54
efs	13	netbios_ssn	34	tftp_u	55
exec	14	netstat	35	telnet	56
finger	15	nnsp	36	tim_i	57
ftp	16	nntp	37	time	58
ftp_data	17	ntp _u	38	urh _i	59
gopher	18	other	39	urp _i	60
hostnames	19	pm _dump	40	uucp	61
http	20	pop _2	41	whois	62
http_443	21	pop_3	42	vmnet	63

Çizelge 5.5: Protokol özniteliği normalizasyonu.

Protocol Öznitelikleri	
TCP	1
UDP	2
ICMP	3

Çizelge 5.6: Bayrak özniteliği normalizasyonu.

Bayrak Öznitelikleri			
Bayrak	Num.	Bayrak	Num
OTH	1	S1	7
REJ	2	S2	8
RSTO	3	S3	9
RSTOS0	4	SF	10
RSTR	5	SH	11
S0	6		

5.4 Gerçekleştirilen İş Akışı

KDD Cup 99 veri setinin % 10 luk kısmı olan ve 494.000 tane kayıt içeren **"kddcup.data_10_percent.gz"** dahilindeki tüm tekil bağlantı vektörlerine ait özniteliklerin sayısal forma dönüştürülmesi ve *KAA* yaklaşımına göre öznitelik seçiminin yapılmasından sonra bu bölümde *YSA* eğitim ve test işlemleri anlatılacaktır. Eğitim ve test aşamalarında MATLAB *"nftool"* kullanılmıştır.

İlk olarak YSA eğitimini daha başarılı gerçekleştirebilmek için Eşitlik **5.1** ile gösterilen "Normalizasyon Formülü" kullanılmıştır ve bu sayede veri setindeki değerlerin tamamı (0,1) aralığında pozitif olarak elde edilmiştir.

$$V_N = 0.8 \times [\frac{V_r - V_{min}}{V_{max} - Vmin}] + 0.1 \quad (\mathbf{5.1})$$

- V_N Normalize edilecek veri
- V_{max} Verinin minimum değeri
- V_{min} Verinin maksimum değeri.

MATLAB *"nftool"* ile *YSA*' nın eğitilmesi ve test edilmesi işlemlerinde gerçekleştirilen basamaklar şu şekildedir:

1. **Girdi/Çıktı Katmanları Sayısı:**25 / 1.
2. **Ağ Tipi :** İleri Beslemeli.
3. **Veri Seçimi :** Rasstgele.
4. **Eğitim Fonksiyonu:** TRAINLM
5. **Adaptasyon Öğrenme Fonksiyonu:** LEARNGDM
6. **Performans Fonksiyonu:** MSE.
7. **Katman Sayısı:** 2.
8. **Nöron Sayısı:** 20.

9. **Transfer Fonksiyonu:** LOGSIG.

10. **Iterasyon:** 100.

11. **Minimum Hata:** 1e-010.

Çalışma üç grup altında incelenebilir. Bu gruplar öznitelik seçimi ve ön çalışma, yapay sinir ağı eğitimi, yapay sinir ağı testi

1. **Grup 1 Öznitelik Seçimi Ve Önişlemler**

   - Adım 1: "kddcup.data_10_percent.gz". Kullan
   - Adım 2: Öznitelik seçimini yap.
   - Adım 3: Normalizasyon işlemlerini yap.

2. **Grup 2 Yapay Sinir Ağı Eğitimi**

   - Adım 4: MATLAB *"nftool"* ile ağı oluştur.
   - Adım 5: Veri setini %85 eğitim, %15 test olarak rastgele böl.
   - Adım 6:%85 eğitim veri setciği ile ağı eğit.

3. **Grup 3 Yapay Sinir Ağı Testi**

   - Adım 7:%15 test veri setciği ile ağı test et.

5.5 Deneysel Sonuçlar

Yapay Sinir Ağı eğitildikten sonra yine "nftool" tarafından gerçekeştirilen test sonuçları Çizege 5.7 ile gösterilmiştir. Başarı Oranları hesaplanırken Eşitlik **5.2**, **5.3**, **5.4** ile gösterilen formüller kullanılmıştır.

$$TespitOrani = \frac{SaldiriTespitSayisi.}{SaldiriSayisi} \times 100\% \quad (\mathbf{5.2})$$

$$YanlisTespit = \frac{YanlisTespitSayisi}{NormalDurumSayisi} \times 100\% \quad (\mathbf{5.3})$$

$$Basari = \frac{DogruSiniflandirma}{BaglantiSayisi} \times 100\% \quad (\mathbf{5.4})$$

Geliştirilen *YSA* ile gerçekleştirilen test sonuçlarında sistemin saldırıları **91.64%** oranında tespit ettiği ve bu oran ile literatürde gerçekleştirilen çalışmalar ortalamasında bir değer yakaladığı görülmüştür. Bunun yanında **81.17%** oranında gerçekleştirilen saldırıların türünü de tahmin etmeyi başardığı göslemlenmiştir. Yakalanan 81.17% oranı düşük görülebilir ancak bunun nedeni 20 tane saldırı türüne ait eğitim verisinin sayısının az olmasıdır. Sonuçlardan da görüleceği üzere örnek sayısı fazla olan Smurf, Neptune saldırıları ve Normal durum verilerinin tahmin edilme oranları çok daha yüksektir.

Çizelge 5.7: Gerçekleştirilen test sonuçları

Saldırı	Test S.	Tespit Oranı	Başarı	Yanlış Tespit
Smurf	12553	12553(100%)	10394(83%)	12553 (0%)
Neptune	1250	1247 (99.76%)	931 (84.48%)	3 (0.24%)
Back	100	80 (80%)	61 (61%)	20 (20%)
Teardrop	100	80 (80%)	52 (52%)	20(20%)
Pod	50	41 (82%)	19 (38%)	9 (18 %)
Land	10	9 (90%)	3 (30%)	1 (10%)
Normal	52821	48813 (92.41%)	48813 (92.41%)	4008 (7.39%)
Satan	100	97 (97%)	61 (61%)	3 (3%)
IpSweep	100	100 (100%)	79 (79%)	0 (0%)
PortSweep	100	97 (97%)	97 (97%)	3 (3%)
Nmap	100	99 (99%)	88(88%)	1 (1%)
WarezClient	100	97 (97%)	77 (77%)	3 (3%)
Guess_Password	10	9 (90%)	4 (40%)	1 (10%)
WarezMaster	10	9 (90%)	3 (30%)	1 (10%)
Imap	10	8 (80%)	7 (70%)	2 (20%)
FTP _Write	4	3 (75%)	0 (0%)	1 (25%)
Multihop	4	3 (75%)	0 (0%)	1 (25%)
Phf	2	2 (100%)	2 (100%)	0 (0%)
Spy	2	2 (100%)	(0%)	0 (0%)
Buffer _Overflow	10	10 (100%)	10 (100%)	0 (0%)
RootKit	4	3 (75%)	1 (25 %)	1 (25%)
LoadModule	4	4 (100%)	1 (25%)	0 (0%)
Perl	2	1 (50%)	0(0%)	1 (50%)
Toplam	67500	67476 (91.64%)	63246 (81.17%)	24 (8.36%)

Çizelge 5.7 ile gösterilen test sonuçlarına ulaşılırken takip edilen yöntem şu şekilde açıklanabilir: veri seti dahilinde bulunan her bir bağlantı vektörünün saldırı (saldırı tipi olarak) veya normal olarak etiketlendiğinden daha önce bahsedilmişti. Bu rakamsal olmayan etiketleme işlemi *YSA* yapısında kullanılmak üzere sayısal formata dönüştürüldü ve daha iyi sonuç alabilmek adına normalizasyon işlemine

tabi tutuldu (ilgili değerler 0 ile 1 arasına indirgendi). Test aşamasında ise *MATLAB* üzerinden sonuçlar nomalleştirme değerleri üzerinden elde edildi ve tekrar karşılaştırılmak üzere gerçek değerine dönüştürüldü (normalleştirme işleminin tersi uygulandı). Bir örnek üzerinden açıklamak gerekirse; *"normal"* olarak belirtilen bir bağlantı vektörü etiketinin sayısal hale dönüştürülme işleminde karşılığı Çizelge 5.2 üzerinden görüleceği üzere "7" olarak gerçekleşmiştir. *"Normal"* duruma karşılık gelen "7" rakamının normalleştirme işlemi ile elde edilen değeri "0,318182" olarak gerçekleşmiştir. Örneğin, Test aşamasına geçersek, *"normal"* olarak etiketlenen ve eğitim aşamasında kullanılmayan bir bağlantı vektörü test edildiğinde "0,319121" gibi bir değer elde edilmiştir. Elde edilen bu değere normalleştirme işleminin tersi uygulandığında ise çıkan sonuç "7,12" olarak gerçekleştirmiştir. "7" değeri için elde edilen sonuç "6,5-7,5" aralığında olduğundan dolayı "7" olarak kabul edilmiş, elde edilen sonucun doğru olduğu ve sistemin *"normal"* durumu tespit ettiği kayıtlara geçmiştir. Bu örnekten yola çıkarak, elde edilen sonuçlara bazı örnekler Çizelge 5.8 üzerinde görülebilir. Yanlış tespit olarak sonuçlanan testler koyu renk ile gösterilmiştir.

Çizelge 5.8: Gerçekleştirilen testlere bazı örnekler.

Sıra No	Test Sonucu	Hedeflenen	Test Sonucu	Hedeflenen
1	0,40145	0,427272727	9,84785	10
2	0,49894	0,5	11,97085	12
3	0,50013	0,5	12,003575	12
4	0,87428	0,863636364	22,2927	22
5	**0,22313**	**0,136363636**	**4,386075**	**2**
6	0,46181	0,463636364	10,949775	11
7	0,62903	0,645454545	15,548325	16
8	0,75434	0,754545455	18,99435	19
9	**0,75742**	**0,281818182**	**19,07905**	**6**
10	0,72862	0,718181818	18,28705	18

6. SONUÇ VE ÖNERİLER

Birbirinden bağımsız birçok bilgisayar sisteminin işbirliği içinde tek bir bilgisayar sistemiymiş gibi çalışabildiği dağıtılmış sistemler yapısının güvenlik mekanizmasının oluşturulmasında faktör olan birçok durum bulunmaktadır. Dağıtılmış sistemler için güvenlik mekanizması oluşturulurken sistem kaynaklarını erişilebilirliği, sistemin dağıtılmış olmasının gizlenebilirliği, sistemin açık ve genişletebilir olması gerekliliği gibi kısıtlar bulunmaktadır. Bu ksıtılar dahilinde dağıtılmış sistemlerin otonomus olarak çalışan birbirinden bağımsız ancak birbiri ile koordineli bileşenlerinin güvenliğe olan etkileri mutlaka hesaba katılmalı ve tüm sistemi kapsayacak bir güvenlik planı oluşturulmalıdır.

Bu çalışmada güvenlik mekanizmalarının ikinci basamağı olan Saldırı Tespit Sistemleri *(STS)* üzerinde çalışılmıştır. Dağıtılmış sistemlerde Saldırı Tespit Sistemlerinin nasıl tasarlanabileceği konusu ile bir sistem geliştirilmiş ve geliştirilen sistemden deneysel sonuçlar elde edilmiştir. Kablosuz Algılayıcı Ağlar *(KAA)* dağıtılmış sistemlere en belirgin örneklerden biridir ve bu tez çalışmasında *(KAA)* üzerinde gerçekleştirilecek bir Saldırı Tespit Sistemi *(STS)* üzerinde çalışmalar gerçekleştirilmiştir. Kablosuz Algılayıcı Ağlar *(KAA)* amaçları topladıkları verileri ana terminale göndermek olan homojen algılayıcı düğümlerden oluşan bir dağıtılmış sistem yapısıdır. Kablosuz Algılayıcı Ağlar doğaları gereği belirli bir düzen içinde olmadıkları bir uygulama dahilinde görevlerini yerine getirebilirler. Bu kapsamda geliştirilecek olan sisteme uygulanabilirliği ve kaynak verimliliği açısından başarılı olmalıdır.

Çalışmanın ana hedeflerinden biri geliştirilecek olan sistemin Kablosuz Algılayıcı Ağların sahip olduğu enerji, işlemci ve depolama kaynaklarının azami dikkatle kullanılması ve verimli bir sistemin oluşturulmasıdır. Bu nedenle sistemin Yapay Sinir Ağları konsepti ile geliştirilmesi planlanmıştır. Yapay Sinir Ağları yaklaşımının sistemimize sağladığı avantajlar şu şekilde sıralanabilir:

- Bilgisayar sistemlerinde hesaplamalar merkezi bir yaklaşımdayken, Yapay Sinir Ağı *(YSA)* yaklaşımında her nöron hesaplama yapar ve işlemler dağıtılmıştır.
- Hafıza paketlenmiş biçimdeyken (bir dizi içinde olabilir) Yapay Sinir Ağı *(YSA)* yaklaşımında dağıtılmıştır ve söz konusu dağıtılmış hafızayı bağlantı ağırlıkları ifade etmektedir.
- Bilgisayar işlemlerinde hata hoşgörüsü yoktur fakat Yapay Sinir Ağı *(YSA)* yaklaşımında hata hoşgörüsü vardır.
- Bilgisayarda bağlantı statiktir, Yapay Sinir Ağı *(YSA)* yaklaşımında dinamiktir.
- Bilgisayarlar tam değerle çalışır fakat Yapay Sinir Ağı *(YSA)* çıkarsama yapabilir.

Yukarıdaki özellikler incelendiğinde saldırı tespit başarısının yanında asıl hedef olan kaynak tüketiminideki verimliliğin Yapay Sinir Ağı *(YSA)* ile sağlanabileceği öngörülmüştür ve bu öngörü ile şu sonuç çıkarılmıştır ki; saldırı tespitinde kullanılan tekniklerden *"Kötüye Kullanım Tespiti"* ve *"Anormallik Tespiti"* ile *KAA*' da verimli bir sistem gerçeklenmesi zordur fakat *YSA* bu verimliliğe yardımcı olabilir. Bunun nedeni şöyle açıklanabilir hem *"Kötüye Kullanım Tespiti"* hem de *"Anormallik Tespiti"* gerçeklenirken algılayıcı düğümlerde saldırıları karşılaştırmak üzere gerekli bilgi tabanı ve kuralları takip etmek için işlemci gibi kaynaklara ihtiyaç vardır ve bu kaynakların kullanımı da fazladan enerji ihtiyacı doğurmaktadır. Yapay Sinir Ağı *(YSA)* yaklaşımında ise kullanılacak ağ bilgisayar ortamında eğitildikten sonra algılayıcı düğümler üzerinde sadece ağırlıklar ile ilgili basit matematiksel işlemler gerçekleştirilebilir. Deneysel sonuçlardan da görülebileceği üzere bilgisayar algılayıcı düğüm üzerinde çalışabilecek bir Yapay Sinir Ağı *(YSA)* bilgisayar ortamında eğitilmiş ve sonucunda başarılı deneysel sonuçlar elde edilmiştir.

Bundan sonraki çalışmalarda ise eğitilen Yapay Sinir Ağı *(YSA)*' dan alınacak değerlerin gerçek bir Kablosuz Algılayıcı Ağ *KAA* uygulamasında gerçeklenmesi ve bunu gerçeklerken kaynakların kullanımındaki verimliliğin arttırılması amacıyla saldırı tespit işleminin bütün algılayıcı düğümlerde değil sadece belirli düğümler sorumluluğunda gerçekleştirilmesi planlanmaktadır. Bu kapsamda saldırı tespit

oranını düşürmeden kaynak kullanım verimliliğini arttıracak mekanizmaları oluşturacak çalışmalar planlanmaktadır.

KAYNAKLAR

[1] **Okan Can, Ozgur Koray Sahingoz, E.K.**, 2014. The Architecture of Mobile Agent Based Intrusion Detection System (MABDIDS), Proceedings of the World Congress on Engineering, cilt 1, http://www.iaeng.org/publication/WCE2014/WCE2014_pp432-437.pdf, s.432–437.

[2] **Abduvaliyev, A.**, **Pathan, A.S.K.**, **Zhou, J.**, **Roman, R. ve Wong, W.C.**, 2013. On the vital areas of intrusion detection systems in wireless sensor networks, *Communications Surveys & Tutorials, IEEE*, **15(3)**, 1223–1237.

[3] **Khedo, K.K.**, **Perseedoss, R.**, **Mungur, A.** *ve diğerleri*, 2010. A wireless sensor network air pollution monitoring system, *arXiv preprint arXiv:1005.1737*.

[4] **Oktay, U. ve Sahingoz, O.K.**, 2013. Proxy Network Intrusion Detection System for Cloud Computing, Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE), 2013 International Conference on, IEEE, s.98–104.

[5] **Hu, F. ve Cao, X.**, 2010. Wireless sensor networks: principles and practice, CRC Press.

[6] **Rassam, M.A.**, **Maarof, M.**, **Zainal, A.** *ve diğerleri*, 2012. A survey of intrusion detection schemes in wireless sensor networks, *American Journal of Applied Sciences*, **9(10)**, 1636.

[7] **Kachirski, O. ve Guha, R.**, 2003. Effective intrusion detection using multiple sensors in wireless ad hoc networks, System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on, IEEE, s.8–pp.

[8] **Muraleedharan, R. ve Osadciw, L.A.**, 2006. Jamming attack detection and countermeasures in wireless sensor network using ant system, Defense and Security Symposium, International Society for Optics and Photonics, s.62480G–62480G.

[9] **Sahingoz, O.K.**, 2013. Large scale wireless sensor networks with multi-level dynamic key management scheme, *Journal of Systems Architecture*, **59(9)**, 801–807.

[10] **Rao, S.**, **Deepak, S. ve Pradeep, P.**, 2013. Parametric analysis of impact of jamming in wireless sensor networks, Wireless and Optical Communications Networks (WOCN), 2013 Tenth International Conference on, IEEE, s.1–5.

[11] **Bysani, L.K. ve Turuk, A.K.**, 2011. A survey on selective forwarding attack in wireless sensor networks, Devices and Communications (ICDeCom), 2011 International Conference on, IEEE, s.1–5.

[12] **Krontiris, I.**, **Dimitriou, T.**, **Giannetsos, T. ve Mpasoukos, M.**, 2008. Intrusion detection of sinkhole attacks in wireless sensor networks, Algorithmic Aspects of Wireless Sensor Networks, Springer, s.150–161.

[13] **Ngai, E.C.**, **Liu, J. ve Lyu, M.R.**, 2007. An efficient intruder detection algorithm against sinkhole attacks in wireless sensor networks, *Computer Communications*, **30(11)**, 2353–2364.

[14] **Sharmila, S. ve Umamaheswari, G.**, 2012. Detection of Sybil attack in mobile wireless sensor networks, *IJESAT] International Journal of Engineering Science & Advanced Technology*, 256–262.

[15] **Sharmila, S. ve Umamaheswari, G.**, 2012. Detection of Sybil attack in mobile wireless sensor networks, *IJESAT] International Journal of Engineering Science & Advanced Technology*, 256–262.

[16] **Zhang, Q.**, **Wang, P.**, **Reeves, D.S. ve Ning, P.**, 2005. Defending against sybil attacks in sensor networks, Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on, IEEE, s.185–191.

[17] **Karlof, C. ve Wagner, D.**, 2003. Secure routing in wireless sensor networks: Attacks and countermeasures, *Ad hoc networks*, **1(2)**, 293–315.

[18] **Kalita, H.K. ve Kar, A.**, 2009. Wireless sensor network security analysis, *International Journal of Next-Generation Networks (IJNGN)*, **1(1)**, 1–10.

[19] **Can, O.**, 2014. Mobile agent based intrusion detection system, Signal Processing and Communications Applications Conference (SIU), 2014 22nd, IEEE, s.1363–1366.

[20] **Okan Can, O.K.S.**, 2015. A Neural Network Based Intrusion Detection System, Proceedings of23th IEEE Conference on Signal Processing and Communications Applications, IEEE, s.45–52.

[21] **Mitchell, R. ve Chen, R.**, 2014. A survey of intrusion detection in wireless network applications, *Computer Communications*, **42**, 1–23.

[22] **Farooqi, A.H. ve Khan, F.A.**, 2012. A survey of intrusion detection systems for wireless sensor networks, *International Journal of Ad Hoc and Ubiquitous Computing*, **9(2)**, 69–83.

[23] **Jurdak, R.**, **Wang, X.R.**, **Obst, O. ve Valencia, P.**, 2011. Wireless sensor network anomalies: Diagnosis and detection strategies, Intelligence-Based Systems Engineering, Springer, s.309–325.

[24] **Moosavi, H. ve Bui, F.M.**, 2014. A Game-Theoretic Framework for Robust Optimal Intrusion Detection in Wireless Sensor Networks, *Information Forensics and Security, IEEE Transactions on*, **9(9)**, 1367–1379.

[25] **Karapistoli, E. ve Economides, A.A.**, 2013. Anomaly detection and localization in UWB wireless sensor networks, Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on, IEEE, s.2326–2330.

[26] **Coppolino, L.**, **D'Antonio, S.**, **Garofalo, A. ve Romano, L.**, 2013. Applying data mining techniques to intrusion detection in wireless sensor networks, P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on, IEEE, s.247–254.

[27] **Nishanthi, S. ve Virudhunagar, T.**, 2013, Intrusion Detection in Wireless Sensor Networks Using Watchdog Based Clonal Selection Algorithm.

[28] **Sedjelmaci, H. ve Feham, M.**, 2011. Novel hybrid intrusion detection system for clustered wireless sensor network, *arXiv preprint arXiv:1108.2656*.

[29] **Walters, J.P.**, **Liang, Z.**, **Shi, W. ve Chaudhary, V.**, 2007. Wireless sensor network security: A survey, *Security in distributed, grid, mobile, and pervasive computing*, **1**, 367.

[30] **Okan Can, O.K.S.**, 2015. A Survey of Intrusion Detection Systems in Wireless Sensor Networks, Proceedings of 6th International Conference on Modeling, Simulation and Applied Optimization, IEEE.

[31] **Okan Can, Ozgur Koray Sahingoz, C.T.**, 2015. A Neural Network Based Intrusion Detection System for Wireless Sensor Networks, Recent Researches in Applied Computer Science, IEEE, s.45–52.

[32] **Jadidoleslamy, H.**, 2011. A hierarchical intrusion detection architecture for wireless sensor networks, *International Journal of Network Security and its Applications (IJNSA)*, **3(5)**, 131–154.

[33] **Anantvalee, T. ve Wu, J.**, 2007. A survey on intrusion detection in mobile ad hoc networks, Wireless Network Security, Springer, s.159–180.

[34] **Liu, X.**, 2012. A survey on clustering routing protocols in wireless sensor networks, *Sensors*, **12(8)**, 11113–11153.

[35] **Mamalis, B.**, **Gavalas, D.**, **Konstantopoulos, C. ve Pantziou, G.**, 2009. Clustering in wireless sensor networks, *RFID and Sensor Networks: Architectures, Protocols, Security and Integrations, Y. Zhang, LT Yang, J. Chen, eds*, 324–353.

[36] **Butun, I.**, **Morgera, S.D. ve Sankar, R.**, 2014. A survey of intrusion detection systems in wireless sensor networks, *Communications Surveys & Tutorials, IEEE*, **16(1)**, 266–282.

[37] **Kriesel, D.**, 2007. A brief introduction to neural networks, *Retrieved August*, **15**, 2011.

[38] **Shirazi, H.**, **Namadchian, A. ve khalili Tehrani, A.**, 2012. A combined anomaly base intrusion detection using memetic algorithm and Bayesian networks, *differences*, **16**, 17.

[39] **Eldos, T.**, **Siddiqui, M.K. ve Kanan, A.**, 2012. On the KDD'99 Dataset: Statistical Analysis for Feature Selection, *Journal of Data Mining and Knowledge Discovery*, **3(3)**.

ÖZGEÇMİŞ

Okan CAN, 1985 yılında Ankara'da doğdu. Orta öğretimini Ankara Süleyman Demirel Anadolu Lisesi'nde tamamladıktan sonra, Hava Harp Okulu Bilgisayar Mühendisliği Bölümü'nden 2007 yılında mezun oldu. İleri seviyede İngilizce bilmektedir.

Ad Soyad: Okan CAN

Doğum Yeri ve Tarihi: Ankara / 03.12.1985

E-Posta: okancan@live.com

Lisans: Hava Harp Okulu Bilgisayar Mühendisliği

Yayın ve Patent Listesi:

- **Can O.**,Sahingoz O. K., *Solving Container Loading Problem with Simulated Annealing Algorithm*, 15th IEEE International Symposium on Computational Intelligence and Informatics, Kasım 2014, Budapeşte, Macaristan.
- **Can O.**, Yılmaz G., *Intrusion Detection Systems for Cloud Computing*, Conference on Advanced Technologies for Aviation, Haziran 2014, İstanbul, Türkiye.

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- **Can O.**, *Mobile Agent Based Intrusion Detection Systems*, 22th IEEE Conference on Signal Processing and Communications Applications, Nisan 2014, Trabzon, Türkiye.
- **Can O.**, Sahingoz O. K., Kugu E., *The Architecture of Mobile Agent Based Intrusion Detection System (MABDIDS)*,World Congress on Engineering, Temmuz 2014, Londra, İngiltere.
- **Can O.**, Sahingoz O. K., *Neural Network Based Intrusion Detection*, 23th IEEE Conference on Signal Processing and Communications Applications, Mayıs 2015, Malatya, Türkiye.
- **Can O.**, Sahingoz O. K., *A Survey Of Intrusion Detection Systems In Wireless Sensor Networks*, 6th International Conference on Modeling, Simulation and Applied Optimization, Mayıs 2015, İstanbul, Türkiye.
- **Can O.**,Sahingoz O. K., Turguner C., *A Neural Network Based Intrusion Detection System for Wireless Sensor Networks*, 15th International Conference on Applied Computer Science, Mayıs 2015, Konya, Türkiye.