ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ÇEVRİMİÇİ SİSTEMLERDE MAKİNE ÖĞRENME YÖNTEMİ İLE DAĞITILMIŞ HİZMET REDDİ (DDoS) SALDIRILARININ TESPİTİ

Dilek BAŞKAYA

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2020

Her hakkı saklıdır

ÖZET

Yüksek Lisans Tezi

ÇEVRİMİÇİ SİSTEMLERDE MAKİNE ÖĞRENME YÖNTEMİ İLE DAĞITILMIŞ HİZMET REDDİ (DDoS) SALDIRILARININ TESPİTİ

Dilek BAŞKAYA

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Refik SAMET

Internetin günlük hayatımızın içinde olmasının sağladığı avantajların yanında, siber tehditlere açık olması dezavantajdır. Dağıtılmıs hizmet reddi (DDoS) saldırıları bu siber tehditlerden biridir ve temel amacı ağ kaynaklarını tükenmesini sağlayarak kullanıcıların sistemlere erişmesini engellemektir. Kullanılan tespit yöntemlerinden biri olan imza tabanlı tespit yöntemi, sadece bilinen saldırı tipleri için uygundur. Günümüzde saldırıların çeşitlerinin ve saldırı yöntemlerinin değistiği görülmektedir. Anomali tabanlı yöntemler, bilinmeyen saldırıların tespiti için önemli avantajdır. Kullanılan algoritmalar dışında, verisetlerinde uygulanan çeşitli ön işlemler ile bulunan tespit oranlarının artışı ve daha hızlı sonuç alınmasını sağlanmaktadır. Bu çalışmada, HTTP (Hypertext Transfer Protocol), TCP SYN (Transport Control Protocol Synchronize), UDP (User Datagram Protocol), ICMP (Internet Control Message Protocol) flood DDoS saldırı verileri kullanılarak hazırlanan eğitim veriseti üzerinde, K-Nearest Neighbor (KNN), Random Forest (RF), Multi Layer Perceptron (MLP), C-Support Vector Machine (SVC) makine öğrenmesi algoritmaları ile ölçeklendirme ve özellik azaltma ön işlemleri uygulanarak, kısa sürede, yüksek saldırı tespit oranları edilmesi için bir model oluşturulmuştur. Oluşturulan model, test veriseti üzerinde uygulanarak, çıkan sonuçlar analiz edilmiştir. Çalışmanın sonunda, farklı DDoS saldırıları üzerinde, kullanılan ön işlem ve makine öğrenmesi algoritmasına göre alınan sonuçların, hem NSL-KDD veriseti üzerinde hem de hazırlanan verisetleri üzerinde farklı olduğu gözlenmiştir. Sonuçlar incelendiğinde, hazırlanan verisetleri üzerinde özellik azaltma ön işlemi sonrasında MLP kullanılarak, 1515 ms sürede %99.2 saldırı tespit oranı elde edilmiştir. Aynı ön işlem NSL-KDD verisetinde uygulandığında, RF algoritması kullanılarak 9744 ms sürede %89.9 ve MLP algoritması kullanılarak 24179 ms sürede %89.9 tespit oranı elde edilmiştir.

Kasım 2020, 84 sayfa

Anahtar Kelimeler: Siber Güvenlik, Siber Saldırılar, Dağıtılmış Hizmet Reddi Saldırıları, Saldırı Tespit Yöntemleri, Makine Öğrenmesi

ABSTRACT

Master Thesis

DISTRIBUTED DENIAL OF SERVICE (DDoS) ATTACKS DETECTION BY USING MACHINE LEARNING METHODS ON ONLINE SYSTEMS

Dilek BAŞKAYA

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof. Dr. Refik SAMET

Besides advantages of having Internet, being open to cyber threats is disadvantage. Distributed denial of service (DDoS) attack is one of cyber threats and main purpose is to prevent users from accessing systems by exhausting network resources. Signature-based detection method, which is one of intrusion detection methods used, is solution for known attacks. Today, it is seen that types and methods of attacks have changed. Anomaly-based methods are important advantage for unknown attacks. Apart from machine learning algorithms, preprocesseses are used on dataset lead to significant increase in found rates and result in shorter time. In this study, model was generated by using K-Nearest Neighbor (KNN), Random Forest (RF), Multi-Layer Perceptron (MLP), C-Support Vector Machine (SVC) algorithms and feature reduction, scaling preprocesses on training dataset which was prepared by using HTTP (Hypertext Transfer Protocol), TCP SYN (Transport Control Protocol Synchronize), UDP (User Datagram Protocol), ICMP (Internet Control Message Protocol) flood. Generated model was applied on test dataset and results were analyzed. At end of study, results obtained according to preprocesses and algorithms are used on different DDoS attacks differ both on NSL-KDD and on prepared datasets. When results were examined, by using feature reduction preprocess before MLP on prepared datasets, rate of 99.2% was achieved in 1515 ms. When same preprocess was applied on NSL-KDD dataset, using RF algorithm, rate of 89.9% in 9744 ms and using MLP algorithm, rate of 89.9% in 9744 ms and using MLP algorithm, rate of 89.9% in 24179 ms was obtained.

November 2020, 84 pages

Key Words: Cyber Security, Cyber Attacks, Distributed Denial of Service Attacks, Intrusion Detection Methods, Machine Learning

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren danışman hocam Prof. Dr. Refik Samet'e, kıymetli tecrübelerinden faydalandığım Bilgisayar Mühendisliği Bölümü öğretim üyelerine, maddi ve manevi destekleri için her zaman yanımda olan aileme, en büyük destekçilerim olan arkadaşlarıma çok teşekkür ederim.

Dilek BAŞKAYA
Ankara, Kasım 2020

İÇİNDEKİLER

TEZ ONAY SAYFASI
ETİK i
ÖZET ii
ABSTRACT iii
TEŞEKKÜR iv
SİMGELER DİZİNİ vii
ŞEKİLLER DİZİNİ viii
ÇİZELGELER DİZİNİ ix
1. GİRİŞ 1
2. İLGİLİ ÇALIŞMALAR 5
3. MAKİNE ÖĞRENMESİ YÖNTEMİ İLE DDoS TESPİTİ İÇİN ÖNERİLEN YÖNTEM 11
3.1 DDoS Saldırı Tespit Topolojisinin Tasarımı 13
3.2 DDoS Saldırı Çeşitlerinin Belirlenmesi 14
3.3 DDoS Saldırı Araçlarının Belirlenmesi 18
3.4 DDoS Veriseti Oluşturulması 18
3.5 Makine Öğrenimini Kullanarak Önerilen DDoS Saldırı Tespit Yöntemleri 20
4. ÖNERİLEN YÖNTEMİN UYGULANMASI 22
4.1 Önerilen Topoloji Tasarımı 23
4.2 DDoS Saldırı Türlerinin ve Araçlarının Topolojiye Uygulanması 24
4.3 Eğitim ve Test Verisetlerinin Oluşturulması 24
4.4 Ön İşleme ve Makine Öğrenimi Yöntemlerinin Uygulanması . . . 27
4.4.1 Ön işlem olmadan makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması 29
4.4.2 Ölçeklendirme ön işlemi ile makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması 32
4.4.3 Özellik azaltma ön işlemi ile makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması 35
4.4.4 Ölçeklendirme ve veri azaltma ön işlemleri ile makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması 41
4.4.5 Hazırlanan verisetleri üzerinde uygulanan ön işlemler ile makine öğrenmesi algoritmalarının NSL-KDD veriseti üzerinde uygulanması 44
5. ELDE EDİLEN SONUÇLARIN DEĞERLENDİRİLMESİ . . 55
6. SONUÇLAR 66
KAYNAKLAR 68
EK 1 Saldiri Çesitlerine Göre Etiketlenmis Verilerden Olusan Egitim Veriseti 71
EK 2 Hazirlanan Verisetlerinde Ön İslem Uygulanmadan Alinan Raporlar 76

EK 3 Hazirlanan Verisetlerinde Ölçeklendirme Ön İslemi Uygulandiktan Sonra Alinan Raporlar 77
EK 4 Hazirlanan Verisetlerinde Özellik Azaltma Ön İslemi Uyguladiktan Sonra Alinan Raporlar 78
EK 5 Hazirlanan Verisetlerinde Özellik Azaltma ve Ölçeklendirme Ön İslemleri Uyguladiktan Sonra Alinan Raporlar 79
EK 6 NSL-KDD Verisetlerinde Ön İslem Uygulanmadan Alinan Raporlar 80
EK 7 NSL-KDD Verisetlerinde Ölçeklendirme Ön İslemi Uygulandiktan Sonra Alinan Raporlar 81
EK 8 NSL-KDD Verisetlerinde Özellik Azaltma Ön İslemi Uygulandiktan Sonra Alinan Raporlar 82
EK 9 NSL-KDD Verisetlerinde Özellik Azaltma ve Ölçeklendirme Ön İslemleri Uygulandiktan Sonra Alinan Raporlar 83
ÖZGEÇMİŞ 84

SİMGELER DİZİNİ

DDoS	Dağıtılmış Hizmet Reddi Saldırıları
DoS	Hizmet Reddi Saldırıları
BBC	British Broadcasting Corporation
MLP	Multi Layer Perceptron
RF	Random Forest
KNN	K-Nearest Neighbor
SVC	C-Support Vector Machine
HTTP	Hypertext Transfer Protocol
TCP	Transport Control Protocol Synchronize
UDP	User Datagram Protocol
ICMP	İnternet Control Message Protocol

ŞEKİLLER DİZİNİ

Şekil 3.1 HTTP flood DDoS saldırısı 15
Şekil 3.2 ICMP flood DDoS saldırısı 16
Şekil 3.3 UDP flood DDoS saldırısı 16
Şekil 3.4 TCP three-way handshake 17
Şekil 3.5 TCP SYN flood DDoS saldırısı 17
Şekil 4.1 Önerilen methodolojinin aşamaları 22
Şekil 4.2 DDoS saldırı veriseti oluşturma ve saldırı tespit topolojisi 23
Şekil 4.3 Kesinlik (precision) değerinin hesaplanması 28
Şekil 4.4 Duyarlılık (recall) değerinin hesaplanması 28
Şekil 4.5 F1-skor (f-score) değerinin hesaplanması 28
Şekil 4.6 Saldırı etkilerine göre korelasyon sonuçlarının ısı haritası 35
Şekil 5.1 Saldırı tespit oranlarının değişimi 56
Şekil 5.2 Eğitim veriseti üzerinde saldırı tespit sürelerinin değişimi 57
Şekil 5.3 Test veriseti üzerinde saldırı tespit sürelerinin değişimi 57
Şekil 5.4 Hazırlanan verisetleri üzerinde 10-çapraz doğrulama sonunda elde edilen tespit oranları 58
Şekil 5.5 Hazırlanan verisetleri üzerinde MLP sonunda elde edilen model başarı oranları 58
Şekil 5.6 Hazırlanan verisetleri üzerinde RF sonunda elde edilen model başarı oranları 59
Şekil 5.7 Hazırlanan verisetleri üzerinde KNN sonunda elde edilen model başarı oranları 59
Şekil 5.8 Hazırlanan verisetleri üzerinde SVC sonunda elde edilen model başarı oranları 60
Şekil 5.9 NSL-KDD veriseti üzerinde saldırı tespit oranlarının değişimi . . . 61
Şekil 5.10 NSL-KDD eğitim veriseti üzerinde saldırı tespit sürelerinin değişimi 62
Şekil 5.11 NSL-KDD test veriseti üzerinde saldırı tespit sürelerinin değişimi . 62
Şekil 5.12 NSL-KDD verisetleri üzerinde 10-çapraz doğrulama sonunda elde edilen tespit oranları 63
Şekil 5.13 NSL-KDD veriseti üzerinde MLP sonunda elde edilen model başarı oranları 63
Şekil 5.14 NSL-KDD veriseti üzerinde RF sonunda elde edilen model başarı oranları 64
Şekil 5.15 NSL-KDD veriseti üzerinde KNN sonunda elde edilen model başarı oranları 64
Şekil 5.16 NSL-KDD veriseti üzerinde SVC sonunda elde edilen model başarı oranları 65

ÇİZELGELER DİZİNİ

Çizelge 3.1 DDoS saldırı çeşitleri 14
Çizelge 4.1 Veriseti içerisindeki özellikler 25
Çizelge 4.2 Hazırlanan verisetleri üzerinde ön işlem yapılmadan alınan sonuçlar 31
Çizelge 4.3 Hazırlanan verisetleri üzerinde ölçeklendirme ön işlemi yapıldıktan sonra alınan sonuçlar 34
Çizelge 4.4 Özellikler arası korelasyon hesaplama sonuçları 37
Çizelge 4.5 Hazırlanan verisetleri üzerinde özellik azaltma ön işlemi yapıldıktan sonra alınan sonuçlar 40
Çizelge 4.6 Hazırlanan verisetleri üzerinde özellik azaltma ve ölçeklendirme ön işlemleri yapıldıktan sonra alınan sonuçlar 43
Çizelge 4.7 NSL-KDD veriseti üzerinde ön işlem yapılmadan alınan sonuçlar 46
Çizelge 4.8 NSL-KDD veriseti üzerinde ölçeklendirme ön işlemi yapıldıktan alınan sonuçlar 49
Çizelge 4.9 NSL-KDD veriseti üzerinde özellik azaltma ön işlemi yapıldıktan alınan sonuçlar 51
Çizelge 4.10 NSL-KDD veriseti üzerinde özellik azaltma ve ölçeklendirme ön işlemleri yapıldıktan sonra alınan sonuçlar 54

1. GİRİŞ

Dünyada gelişen teknolojiyle birlikte finans, alışveriş, hastane randevusu gibi birçok hizmet İnternet üzerinden alınmaktadır. İnternet üzerinden erişilebilen hizmetlerin sağladığı kolaylıklar aynı zamanda siber tehditlere neden olmaktadır. Kullanıcı ve sistem bilgilerinin herkesin erişebileceği noktada olması, siber suç işleyen kişi ya da gruplar için avantajdır. Bilgilerin korunmasının yanında, hizmet sağlayan sistemlerin sürekli erişilebilir olması hem kullanıcılar hem de sistem yöneticileri açısından büyük önem taşımaktadır. Hizmet Reddi saldırıları (DoS), zararlı paketleri ağ üzerinden, seri şekilde göndererek ve ağda kötü amaçlı trafik oluşturarak, ağın iletişim ve hesaplama gücünü tüketmeyi amaçlamaktadır. Her sistemin gönderilen isteklere cevap vermesi için sınırlı hizmet kapasitesi vardır. Eğer bu kapasite aşılırsa gelen isteklere cevap veremez ve erişilemez hale gelir. DDoS saldırısı, DoS saldırısının birçok noktadan ve aynı anda tek bir hedefe karşı yapılmasıdır. Böylece etkisi ve karmaşası yükselmektedir.

DDoS saldırıları finansal çıkar, intikam, ideolojik, güç gösterisi ve siber savaş gibi birçok nedenle, saldırganlar tarafından kurban sistemlere uygulanmaktadır. İlk olarak, finansal çıkarlar için yapılan saldırılarda, saldırganlar herhangi bir kuruluşu veya şirketi mali kazanç için kurban seçerler. Mali olarak hedeflenen bu saldırıları önlemek veya savunmak zordur. Diğer bir motivasyon yöntemi intikamdır. Bu DDoS saldırıları, bazı hayal kırıklığına uğramış bireylerin kişisel intikam veya haksızlık nedeni ile bir saldırı başlatmasının sonucudur. İdeolojik olan saldırılar ülkeler için siber tehlike oluşturmaktadır. Saldırganlar ideolojik inançları nedeniyle hedefe saldırmaktadırlar. Diğer motivasyonlara kıyasla saldırının sıklığı çok olmasa da, etkileri ve boyutları büyüktür. Örneğin Estonya'da 2007 yılında yaşanan DDoS saldırısı Estonya'da yaşayan Ruslar ve Estonya hükümeti arasındaki siyasi huzursuzluk sebebiyle başlamıştır. Saldırılar, başbakan, parlamento, çeşitli bakanlıklar dahil olmak üzere altyapının birçok bölümü etkilemiştir. Güç gösterisi amacı ile DDoS saldırılarını yapanlar, yetenek ve güçlerini göstermek için bu saldırıları hayata geçirirler. Bu saldırılar, saldırganlar arasında üstünlük yarışı olarak kabul edilebilir. Son olarak, siber savaş amacı ile

uygulanan DDoS saldırıları, başka bir devletin, ulusun veya uluslararası bir şirketin bilgisayarına veya bilgi sistemine yönelik başlatılan saldırılardır.

DDoS saldırısının etkilerini daha iyi anlamak ve tehlikesinin farkına varmak için yaşanan DDoS saldırılarını incelemek gereklidir. Geçmişte yaşanan ve büyük şirketlerin kurban olarak seçildiği saldırılardan bazıları, dünyada büyük ses getirmiştir. İlk olarak, Ekim 2016'da Amerika'da bulunan DNS servis sağlayıcısı Dyn'e yapılan DDoS saldırısı bütün dünyada yankı uyandırdı. Bu saldırı, güvensiz nesnelerin İnterneti (IoT) cihazlarındaki güvenlik açıklarından yararlanılarak başlatıldı. Olaydan sonra, IoT cihazlarının güvenlik açıkları konuşulmaya ve düzeltilmeye başlandı. Daha sonra 2018 yılında bilinen en büyük DDoS saldırısı Github sunucularına yapıldı. Saldırı sırasında, saniyede 126.9 milyon paketle 1.35 Tbps'ye ulaşan trafik kaydedilmiştir (Russell 2018). Sadece Avrupa ve Amerika'da değil 2019 yılında Türkiyede, Türk Telekom ve Vodafone servis sağlayıcıları tarafından altyapı desteği verilen Garanti Bankası sunucuları DDoS saldırısına uğramıştır. Saldırı süresi boyunca Garanti Bankasına ait web sitesi, ödeme sistemleri görev yapamamıştır.

Dünyada örnekleri verilen saldırılar incelendiğinde, her saldırganın amacına ve sunucuların verdiği hizmetlere göre DDoS saldırılarının etkisinin ve çeşidinin değiştiği gözlemlenmektedir. Dyn şirketine yapılan saldırı DNS sunucularını etkilemişken, Github'a yapılan saldırı web sunucularını etkilemiştir. Kaspersky şirketi bilgilerine göre, 2020 yılının ilk çeyreğinde dünyada en çok kaydedilen DDoS saldırı çeşidi TCP SYN flood DDoS saldırısıdır ve tüm DDoS saldırılarının %92.57'sini kapsamaktadır. TCP SYN flood saldırısını sırasıyla ICMP flood (%3.62), UDP flood (%1.84), TCP flood (%1.68) ve HTTP flood (%0.29) saldırıları takip etmektedir (Kupreev vd 2020).

Yukarıda belirtildiği gibi, uygulanan farklı DDoS saldırı çeşitleri nedeniyle kullanıcılar sunulan hizmetlerden yararlanamazken, saldırıya uğrayan organizasyonlar ciddi maddi kayıplara uğrayabilmektedir. Kaspersky şirketinin Lab araştırmasına göre, bir DDoS saldırısı, küçük işletmeler için ortalama 120.000 dolar ve büyük işletmeler için ortalama olarak 2.000.000 dolar zarara yol açmaktadır (Berard 2018). Dünyada birçok noktada, oldukça fazla sayıda DDoS saldırısı meydana gelmektedir. Bilinen bir

marka olan Arbor cihazların yönetiminden sorumlu olan Netscout şirketinin Threat Intelligence raporuna göre 2019 yılında gözlenen DDoS saldırı sayısı 8.4 milyondur. Her gün 23.000'den fazla ve dakikada 16 saldırıya eşit gelen bu değer, DDoS saldırılarını engelleme çalışmalarına önem verilmesi gerektiğini kanıtlamaktadır (Hummel 2019).

DDoS saldırıları gün geçtikçe daha karmaşık hale gelmekte ve alınan önlemler yetersiz kalabilmektedir. Saldırganlar ise düşük ücretler karşılığında sistemleri işlevsiz hale getirebilmektedir. Bu durum saldırganlar için büyük bir fırsat olmanın yanında, bilgi güvenliği uzmanları için görev haline gelmiştir. DDoS saldırılarına karşı sistemlerin korunması için birçok yöntem kullanılmaktadır. Kullanılan yöntemlerde, gelen toplam trafik üzerinde çalışılarak, trafik sınıflandırması yapılmakta ve kötü amaçlı olma olasılığı en yüksek olan trafik filtrelenmektedir. Aynı zamanda saldırıların tespit süresinin kısaltılması için iyileştirmeler yapılmaktadır. Tespit süresinin kısaltılmasıyla sistem yöneticilerinin durumu kısa sürede kontrol altına almaları ve saldırının etkisini azaltmaları sağlanır.

DDoS saldırısına yol açan kötü amaçlı trafiği bulmak için kullanılan yöntemler, imza tabanlı ve anomali tabanlı olmak üzere ikiye ayrılmaktadır. İmza tabanlı yöntemler, bilinen yani önceden kayıt altına alınmış saldırıların tespiti için kullanılmaktadır. Fakat gelişen teknoloji ve erişim kolaylıkları nedeniyle DDoS saldırıları gün geçtikçe değişmekte ve gelişmektedir. İmza tabanlı yöntemler bilinmeyen saldırılar için yetersizdir. Ayrıca imzaların sürekli yenilenmesi ve yeni saldırıların kayıt altına alınması bir iş yükü gerektirmektedir. Anomali tabanlı yöntemler ise daha önce meydana gelen saldırıları örnek alarak ve kullanılan çeşitli algoritmalar yardımıyla karşılaştırma yaparak kötü amaçlı trafiği tespit etmeye çalışır. Bu sayede bilinmeyen yani daha önce kayıt altına alınmamış saldırıların da tespitini sağlar.

Karşılaşılan her siber saldırıyı tespit edebilen ve bütün verisetleri ile tam uyumlu çalışan genel bir anomali tabanlı siber saldırı tespit metodolojisi bulunmamaktadır. Bu nedenle diğer siber saldırı çeşitleri gibi DDoS saldırılarını tespit etmek için birçok farklı metodoloji geliştirilmektedir. Geliştirilen metodolojilerde kullanılan farklı makine öğrenmesi algoritmaları gibi, öncesinde uygulanan farklı ön işlemler de saldı-

rıların tespit oranlarını ve tespit sürelerini etkilemektedir. DDoS saldırılarını tespit etmek dışında çeşitlerini belirlemek, tehlike altında bulunan ve önlem alınması gereken sistemi belirlenmesi için büyük rol oynamaktadır ve farklı DDoS saldırı çeşitleri için, kullanılan yöntemlere göre farklı sonuçlar elde edilmektedir.

Bu çalışmada ilk olarak, hazırlanan topoloji üzerinde HTTP flood, TCP SYN flood, UDP flood ve ICMP flood DDoS saldırılarını içeren eğitim ve test verisetleri hazırlanmıştır. Daha sonra hazırlanan eğitim veriseti üzerinde, sırası ile ön işlem uygulanmadan, ölçeklendirme ön işlemi uygulanarak, hesaplanan korelasyon değerlerine göre özellik azaltma yöntemi uygulanarak ve son olarak her iki ön işlem uygulanarak, MLP, KNN, RF ve SVC makine öğrenmesi algoritmaları çalıştırılmıştır. Yapılan işlemler sonunda hazırlanan saldırı tespit modeli, anlık verilerden oluşan test veriseti üzerinde uygulanarak, yapılan ön işlemlerin ve makine öğrenmsi algoritmalarının, saldırı tespit, kesinlik (precision), duyarlılık (recall) oranlarına etkisi ve yapılan yöntemler ile ne kadar sürede sonuç alındığı incelenmiştir. Ayrıca çalışmaya ek olarak, hazırlanan verisetleri üzerinde uygulanan ön işlemler ve makine öğrenmesi algoritmaları, NSL-KDD veriseti içerisinde bulunan DDoS saldırı verileri üzerinde uygulanarak alınan sonuçlar analiz edilmiştir.

Çalışmanın bölümlerini incelersek; Bölüm 2'de benzer alanda yapılan çalışmaların incelemesi yer almaktadır. Bölüm 3'te DDoS saldırılarının tespiti için önerilen yöntem anlatılmaktadır. Bölüm 3'ün alt bölümlerinde DDoS saldırılarının tespiti için önerilen topolojiden, topoloji üzerinde uygulanması önerilen saldırı çeşitlerinden ve saldırı araçlarından, eğitim ve test veriseti oluşturulmasından bahsedilmiştir. Bölüm 4'te oluşturulan modelin uygulanmasındaki aşamalar hakkında bilgiler verilmiştir. Beşinci bölümde saldırı tespit metodolojisi sonunda yapılan analizler değerlendirilmiştir. Son bölümde yapılan çalışmanın sonuçları yer almaktadır.

2. İLGİLİ ÇALIŞMALAR

Bilgi güvenliğinin önem kazanması ile konu hakkındaki çalışmalar ve araştırmalar artmıştır. Makine öğrenmesi yöntemlerinin bilgi güvenliğine katkı sağlaması, siber güvenlik ve veri bilimi ile uğraşan birçok insanı tek çatı altında toplamıştır. Araştırmacılar DDoS saldırılarını tespit etmek için çeşitli yöntemler ve verisetleri üzerinde incelemeler yapmaktadırlar. Bu konudaki birçok çalışma incelendiğinde geçmişten günümüze önemli sonuçlar elde edildiği gözlemlenmektedir. Geliştirilen yeni yöntemler ve verisetleri gelecekteki çalışmalara yol göstermektedir.

Bu konuda Awais Ahmed ve arkadaşlarının yayınladığı makalede (Ahmed vd. 2020), DDoS saldırılarının anlık tespit sürelerinin minimuma indirilmesi amacıyla yaptıkları çalışma anlatılmaktadır. Çalışmada, 'Spark Based Anomaly Detection Framework (SAD-F)' ismi ile oluşturdukları sistem ile makine öğrenmesi yöntemleri kullanarak anlık saldırı tespiti yapılmıştır. Veriseti olarak KDD ve UNSW-NB-15 dahil olmak üzere NetFlow trafiği için karşılaştırmalı bir analiz sunulmuştur. Tespit için denetimli makine öğrenimi modelleri olan KNN, Naive Bayes (NB), RF, Decision Tree (DT), Support Vector Machine (SVM) algoritmaları kullanılmıştır. Böylece denetimli makine öğrenme algoritmaları arasında paket sayısına bağlı olarak alınan çalışma süreleri hesaplanmıştır. Bu karşılaştırmalı analiz sonunda anomalileri çok kısa bir zamanda doğru bir şekilde tespit ederek, gerçek zamanlı veri analiz süresi ve tespit oranı bakımından verimli sonuçlar alınmıştır. Çalışmada, SAD-F'nin en kötü durum için %92 tespit değeriyle eğitmek ve test etmek için toplam 210 dakika süreceği teyit edilmiştir. Kullanılan algoritmaların karşılaştırmalı analizleri dikkate alındığında her algoritmanın paket sayısına göre değişen işlem sürelerine dikkat çekilmiştir. Çalışmada, anlık saldırı tespiti için hesaplanan tespit oranları yanında, hesaplanan çalışma süreleri ile zamanın önemine dikkat edilmesini sağlanmıştır.

Jean-Camille Birget ve arkadaşları tarafından yapılan çalışmada (Birget vd. 2018) hacim tabanlı DDoS saldırılarının tespiti için yayınlanan verisetleri yerine ağ izleme yazılımı ve DDoS saldırı aracı kullanılarak bir veriseti oluşturulmuştur. Özellik seçimi, protokole, hedef IP adresine ve verisetindeki zaman bilgilerine göre yapılmış-

tır. İlk olarak, ölçeklendirilmiş verilere uygulanan denetimsiz K-Means++ kümeleme algoritması, daha sonra oluşturulan kümelere denetimli Destek Vektör Makinesi algoritması uygulanmıştır. Saldırı tespit yöntemleri kullanılarak oluşturulan modellenmiş ağa yapılan saldırılar tespit edilmiştir.

Smys ve Pandian'ın yapıtığı çalışmada telekomünikasyon ağı üzerinde SVM ve Neural Network birlikte kullanılarak DdoS saldırı tespiti yapılmıştır (Smys ve Pandian 2019). Hazırlanan metodolojide ilk önce trafik boyutu, paket boyutu, kaynak IP adresi, hedef IP adresi ve port adresleri özellik olarak seçilmiştir. Daha sonra ağın trafik akışı izlenmiş ve saldırıya olma ihtimali bulunan paketler ile ilgili bilgiler toplandıktan sonra, trafik akışındaki normal ve anormal olanlar, paket boyutunu, IP'yi belirlemek için ikinci dereceden entropi uygulanarak değerlendirilmiştir. Çıkarılan değer, belirlenen eşik değerleri ile karşılaştırılarak normal ve anormal değerler olarak ayrılmıştır. Depolanan bilgiler, öğrenme modeli tarafından hacim, protokoller ve uygulamaya dayalı saldırılardan gelen normal trafiği sınıflandırmak için Neural Network ve SVM modelini eğitmek için kullanılmıştır. Çalışmanın sonucunda, gerçek pozitif, yanlış pozitif, yanlış negatif ve yanlış pozitif koşullara dayalı olarak tespitin doğruluğunu numaralandırmak için önerilen birleşik sinir ağı ve destek vektör makinesinin değerlendirilmesinde SVM, NN ve CNSVM'nin performans değerlendirmesi, önerilen yöntemin SVM ve Neural Network'e kıyasla daha kısa tespit süresine sahip olduğu kanıtlanmıştır. Ayrıca tespit sürecinde SVM ve Neural Network metotları ile karşılaştırma için önerilen CNSVM'nin, DDoS saldırıları için %40 fazla tespit oranına sahip olduğu gözlemlenmiştir.

Dinçalp ve arkadaşları (Dinçalp vd 2018), DDoS saldırılarını tespit etmek ve önlemek için DBSCAN kümeleme algoritması kullanmışlardır. Çalışmanın ilk aşamasında saldırı verileri toplanarak bir veriseti oluşturulmuştur. Gelen trafik verileri üzerinden belirli bir eşiğin üstünde gelen veriler seçilmiştir. Sistemdeki, gelen trafik yükü üzerinde, eşiği aşan trafiği belirlemek için belirli zaman aralıkları ile kontrol sağlanmıştır. Belirlenen eşik aşıldığında oluşan alarm süreleri kayıt edilmiştir. Kümeleme aşamasında, alarm durumunda anormallik tespitinin kesinleşmesi için bir süre beklenmiştir. Daha sonra anomali tespiti devam ettiği durumda trafik normalize edilmiştir ve daha sonra saldırının nerede olduğunun tespiti için kümeleme

tekrarlanmıştır. Verisetleri içerindeki özelliklerin seçimi DDoS saldırı tiplerine göre yapılmıştır. İlk verisetinde kayıt süresi, TCP paket sayısı, UDP paket sayısı, ICMP paket sayısı ve diğer paketlerin sayısı kullanılmıştır. İkinci verisetinde kayıt süresi, bir saniyede alınan veri boyutu, bir saniyede alınan paket sayısı, toplam veri boyutu ve toplam paket sayısı kullanılmıştır. Saldırı önleme işlemi için kaynak IP adresi, saldırı çeşidi, saldırı hedefi, saldırı süresi verileri kullanılmıştır. Çalışmada, iyi özellik seçimi yapılmasının sonunda yoğunluk temelli kümelemenin benzer özelliklere sahip verileri zararlı verilerden ayırmak için iyi bir yöntem olduğu gözlemlenmiştir.

Diğer bir makalede Erman Özer ve arkadaşlarının yaptığı çalışmada (Özer ve İskefiyeli 2017), anlık olarak DDoS saldırısının tespiti için ağ trafiği 'libpcap' kütüphanesi üzerinden izlenmiş, paket yakalama ve paket gönderme işlemleri gerçekleştirilmiştir. Anlık DDoS saldırı tespiti için yapılan çalışmada DDoS veya DoS saldırısı olup olmadığı test edilmiştir. Bu çalışmada gelen trafik, kullanıcının belirlediği protokol ve paket sayısına göre değerlendirilmiştir. DDoS saldırıları için ayrı bir sınıflandırma yapılmamıştır.

2017 yılında Gu (Gu vd. 2017) ve arkadaşlarının yaptığı çalışmada DDoS saldırı tespiti için, birden çok özellik tabanlı, yarı-denetimli kümeleme algoritmasına dayanan yeni bir yöntem önerilmiştir ve DDoS saldırı davranışını tespit etmek için yalnızca az miktarda etiketli veri ve nispeten büyük miktarda etiketlenmemiş veri kullanılmıştır. Önerilen yarı-denetimli yöntem, diğer yöntemler ile karşılaştırıldığında daha yüksek tespit oranına sahip olduğu ve tespit süresinin daha kısa olduğu gözlemlenmiştir. Özellik seçimi için tek yönlü bağlantı paketleri yani sadece isteklerin bulunduğu bağlantı paketleri ve paketlerin entropi değerleri kullanılmıştır. Daha sonra DARPA eğitim ve test verisetleri etiketsiz olarak kullanılarak, ilk önce yarı-denetimli K-Means alogirtması, ikinci olarak geliştirilen Multiple-Features-Based Constrained-K-Means (MF-CKM) algoritması uygulanmıştır. K-Means kümeleme yönteminde, algoritmanın ilk kümeleme merkezleri rastgele seçilmektedir, fakat geliştirilen algoritmada ilk kümeleme merkezlerinin seçimine rehberlik etmek için az miktarda etiketli veri kullanılmıştır ve diğer verileri eğitmek için etiketlenmemiş veriler kullanılmıştır. Üç aşamalı MF-CKM algoritmasında; ilk aşamada kaynak IP ve hedef IP özellikleri verisetinden çıkarılmıştır. Eğitim aşamasında kümeleme için ilk kümeleme merkezi

etiketli verilere göre hesaplanır ve değerler yakınlaşana kadar diğer etiketlenmemiş veri kümeleri ile ilk kümeleme merkezi arasındaki benzerlik hesaplanır. Eğitim sürecindeki kümeler oluşturulur. Tespit aşamasında, yeni veri paketlerini yakalayarak ve özniteliklerin ayıklanması, öznitelik değerleri arasındaki mesafeler yeni veri ve her kümeleme merkezi hesaplanır. Sağlanan algoritma, tek özelliğin kullanılmasının neden olduğu düşük algılama verimliliği sorununu azaltmak için özellik algılaması olarak özellik vektörünü kullanmaktadır. Yapılan işlemler sonunda, MF-CKM algoritmasının K-Means algoritmasına kıyasla daha az zaman tükettiği gözlemlenmiştir.

Osanaiye vd. (Osanaiye vd. 2016), DDoS saldırılarının tespiti için farklı makine öğrenme yöntemleriyle uygulanan farklı özellik seçim yöntemlerinin etkisini analiz etmişlerdir. Çalışmada NSL-KDD verisetinde Information Gain, Gain Ratio, Chi-Squared, ReliefF ön işlemleri ile özellik seçimi yapılmıştır. Ardından, dört farklı ön işleme yöntemi için, verisetine altı makine öğrenimi yöntemi uygulanmıştır. RF, One R, DT, Bayes Net, NB makine öğrenme yöntemlerinin sonuçları incelendiğinde farklı ön işlem yöntemleri için farklı makine öğrenme yöntemlerinde farklı sonuçlar elde edildiği görülmektedir. En iyi sonuç Chi-Square ön işleme yöntemi ve J48 sınıflandırma yöntemi ile elde edilmiştir.

Shuai Zhao ve arkadaşlarının yaptığı çalışmada (Zhao vd. 2015), Apache Storm ve Apache Kafka kullanarak ağ izleme ve ağ yönetimi ile anomali tespiti için yeni bir anlık tespit sistemi oluşturulmuştur. NB, SVM ve DT dahil olmak üzere üç farklı makine öğrenmesi öğrenmesi sınıflandırma algoritması kullanılmıştır. Gerçek zamanlı veriler Netflow yardımı ile toplanmıştır ve Weka Makine Öğrenim Aracı ile test edilmiştir. Kaynak IP adresi ve hedef IP adresi kullanılarak yapılan SVM sonucunda %99.9 tespit oranı elde edilmiştir. Çalışmada tespit oranı oldukça yüksektir. Netflow verilerinde özellik seçimi kullanılmıştır. Fakat seçilen özellik olarak kaynak ve hedef IP adresi ile kaynak ve hedef arabirim kullanılması özellik sayısı olarak her zaman yeterli değildir. Çünkü çoğu saldırgan sahte IP adresleri kullanarak, saldırı kaynağını gizlemektedir.

DDoS saldırılarının tespiti için, 2017 yılında Sharma ve arkadaşlarının yaptığı çalışmada (Sharma vd. 2017), DDoS saldırılarının tespitinde daha iyi sonuçlar elde etmek adına KDD'99 veriseti üzerinde özellik azaltma yöntemi kullanılmıştır. Kuru-

lan sistemde bilgi kazancı (information gain), korelasyon kavramları ön işlem olarak kullanılarak, veriseti üzerinde gereksiz ve fazla veriler kaldırılmıştır. ANN kullanılarak oluşturulan sistemde özellik azaltma işlemi ile daha iyi sonuçlar alınmıştır. Çalışmanın sonunda alınan sonuçlar incelendiğinde, veri kaynaklarının daha iyi kullanılması için, verisetinde bulunan bazı özelliklerin kaldırılmasının makine öğrenmesi algoritmalarına yardımcı olduğu gözlemlenmiştir.

Deka ve arkadaşlarının çalışması incelendiğinde (Deka vd. 2019), ağ trafiğinin uygun maliyetli sınıflandırılması ve verisetinin özelliklerini sıralamanması için paralel kümülatif sıralama algoritması uygulanmıştır. Doğrulama yöntemi için farklı verisetleri kullanılmıştır. DDoS saldırı trafiğinin tespiti için SVM ikili sınıflandırıcısını eğitmek için denetimsiz yöntem tarafından uygun özelliklerin seçilmesinde aktif öğrenmenin önemini vurgulayan çalışmada, ağ trafiğinin yüksek oranda sınıflandırılmasını sağlamak için verisetinden küçük eğitim örnekleri seçilmiştir. Özellik sıralaması ve kullanılan algoritma ile büyük verisetlerinde en yararlı özellikler, verisetlerinde tespit edilmiş ve SVM kullanılarak % 92 ile % 97 arasında tespit oranı elde edilmiştir.

Son olarak, Muhammed Aamir ve Syed Zaidi'nin 2019 yılında yayınladığı makale incelediğinde (Aamir ve Zaidi 2019), DDoS saldırılarının tespiti için özellik seçimi yöntemleri ve bu yöntemlerle birlikte, uygulanan makine öğrenmesi algoritmaları karşılaştırılmıştır. Böylelikle her özellik seçiminin farklı modeller üzerindeki etkiside hesaplanmıştır. Tespit için geriye doğru eleme (backward elimination), chi2, bilgi kazancı (information gain) özellik seçimi yöntemleri KNN, SVC, RF, Artificial Neural Network (ANN), NB algoritmaları kullanılarak test edilmiştir. Çalışmada veriseti olarak AI-Kasassbeh ve arkadaşlarının (Al-Kasassbeh vd. 2016) oluşturduğu veriseti kullanılmıştır. Tüm özellik azaltma yöntemleri için %93'ün üzerinde tespit oranı gözlemlenmiştir ve KNN algoritması en iyi performans gösteren model seçilmiştir.

2020 yılında, Başkaya ve Samet tarafından yayınlanan makale incelendiğinde (Başkaya ve Samet 2020), çevirimiçi sistemlerde DDoS saldırı çeşitlerini tespit etmek için farklı ön işlemlerin ve farklı makine öğrenmesi algoritmalarının saldırı tespit oranlarına etkisi analiz edilmiştir. Yapılan çalışmada, ilk adımda farklı DDoS saldırı çeşitlerinden oluşan eğitim veriseti oluşturulmuştur. Daha sonra eğitim veriseti üzerinde ölçeklendirme ve özellik azaltma işlemleri, önce sırasıyla, daha sonra birlikte

uygulanarak, MLP, KNN , RF ve SVC makine öğrenmesi algoritmaları çalıştırılmıştır. Eğitim veriseti ile oluşturulan model, anlık veriler toplananarak oluşturulan test veriseti üzerinde uygulanmıştır. Kullanılan makine öğrenmesi algoritmaları ile uygulanan ön işlemlerin HTTP flood, TCP SYN flood, ICMP flood ve UDP flood saldırılarının tespit oranlarına etkisi incelenmiştir. Çalışmanın sonunda özellik azaltma ön işlemi sonunda kullanılan MLP makine öğrenmesi algoritması ile ortalama %99.2 tespit oranı elde edilmiştir.

Yukarıda bahsedilen çalışmalar incelendiğinde, NSL-KDD vb. gibi yayınlanmış verisetlerinin veya kütüphaneler ve protokoller kullanılarak hazırlanan verisetlerinin kullanıldığı görülmektedir. Daha kısa sürede, yüksek tespit oranı elde etmek için, verisetlerinde farklı ön işleme yöntemleri ve farklı makine öğrenmesi algoritmaları uygulanmıştır. Uygulanan adımlar ve sonuçlar değerlendirildiğinde, uygulanan ön işleme ve makine öğrenimi yöntemlerinin verisetleri üzerinde etkisinin farklı olduğu görülmüştür. Ancak çalışmaların çoğu incelendiğinde, DDoS saldırı türleri için herhangi bir sınıflandırma olmadığı ve çoğunlukla yayınlanmış verisetleri ile yapılan çalışmaların olduğu görülmektedir.

3. MAKİNE ÖĞRENMESİ YÖNTEMİ İLE DDOS TESPİTİ İÇİN ÖNERİLEN YÖNTEM

Günümüzde kullanılan birçok DDoS saldırı tespit yöntemi ve bu konu hakkında yapılan akademik çalışmalar incelendiğinde, hizmet veren birçok organizasyonun DDoS saldırılarından korunması için uygulaması gereken ilk adımın saldırı trafiğini hızlı ve doğru tespit etmek olduğu görülmektedir. Çoğu saldırı tespit tekniği, bilinmeyen saldırıları tespit edememekten veya kabul edilemez derecede yüksek yanlış alarm sayılarından muzdariptir. Maalesef, birçok farklı çalışmadan ve farklı sonuçlardan görüldüğü üzere, günümüzde farklı siber saldırıların neden olduğu sorunların çözümlenmesi için uygulanabilecek genel bir çözüm yöntemi bulunmamaktadır.

Günümüzde kullanılan ağ saldırı tespit sistemleri incelendiğinde, imza tabanlı veya anomali tabanlı yöntemlerin kullandığı görülmektedir. İmza tabanlı yöntemler, daha öncesinde kaydedilen ve bir imzada kodlanan saldırıları tespit edebilirler, fakat yeni saldırıları tespit edemezler. Ayrıca, imza güncelleme, izinsiz giriş varyasyon tespiti ve çok fazla yanlış alarm gibi problemleri bulunmaktadır. Aksine anomali tabanlı çalışan saldırı tespit sistemleri, önce korunması gereken sistemin normal ağ trafiğini ve saldırıların davranışını modellemeye çalışır, daha sonra bu davranışlara göre bir sapma tespit etmek için model üzerinde çalıştırılır (Folino ve Sabatino 2016).

Anomali tabanlı saldırı tespit sistemi, bir saldırıyı tespit etmek için önce saldırı imzası bilgisi gerektirmediğinden, imza tabanlı sistemden daha iyi bir seçenek olarak kabul edilir. Bununla birlikte, sistem tarafından üretilen alarmları yönetmek, imza tabanlı saldırı tespit sistemler ile daha zordur. Bunun nedeni, imza tabanlı yöntemlerin, bildirilen alarmlarla birlikte gereksiz birçok bilgi üretmesidir, oysa anomali tabanlı yöntemler kötü amaçlı olarak algılanan bağlantı akışını tanımlayabilir. Ayrıca normal davranışın temel modelinden sapan bir aktivite algıladığında alarm verir. Bu, tespit sistemlerinin alarmlarının yönetilmesinde ve gerçek alarmlardan yanlış pozitifin tanınmasını zor hale getirmektedir. Bu nedenle, anomali tabanlı algılama sistemleri tarafından tespit edilen saldırı çeşidinin belirlenmesi önemli bir görevdir (Qassim vd. 2016).

Anomali tabanlı saldırı tespit sistemlerinin ilk adımı verisetleri üzerinde yapılan işlemlerdir. Verileri daha verimli kullanabilmek ve anlamlı hale getirmek için verisetleri üzerinde, gerekli ise çeşitli iyileştirmeler yapılmalıdır. Verisetlerinde bulunan sayısal olmayan değerlerin kategori halinde numaralandırılması, verisetinde istenmeyen bazı karakterlerin temizlenmesi ve tek veri tipine dönüştürülmesi bu iyileştirmelerden bazılarıdır.

Günümüzde makine öğrenmesi yöntemlerinin uygulanması ve test edilmesi için kullanılan hazır verisetleri bulunmaktadır. Birçok konuda olduğu gibi, çeşitli kurumlar tarafından hazırlanan ve kullanılan, DDoS saldırı örneklerini ve özelliklerini bulunduran verisetlerinin birçoğu İnternet üzerinden indirilebilmektedir. Bu hazır olan verisetleri dışında çeşitli saldırı araçlarını kullanarak saldırı verisi toplamak mümkündür. Böylece araştırılması istenen DDoS saldırı çeşidine ve çalışma mekanizmasına göre bir model oluşturulabilir.

DDoS saldırı trafiğini bulunduran verisetlerinin hazırlanması için bir ağ izleme cihazı yardımı ile gelen trafik izlenerek veriler kaydedilebilmektedir. Kayıt altına alınan veriler, veri sahipleri tarafından belli bir filtrelemeden geçerek ve etiketlenerek eğitim veriseti olarak saklanmaktadır. Saldırıların hedefi olan kurban cihaz üzerine gelen anlık veriler test veriseti olarak toplanmaktadır. Test verisetinde bulunan veriler etiketsizdir, verisetinde bulunan trafik verileri hakkında bilgilere ve sonuçlara ulaşılması için eğitim verisetleri ve çeşitli makine öğrenmesi algoritmaları uygulanmaktadır.

Makine öğrenmesi yöntemleri ile ulaşılan tespit değerlerinin artışı için toplanan veriler üzerinde, çeşitli ön işlemler yapılabilir. Eğitim ve test verisetleri üzerinde uygulanan algoritmalar haricinde, verisetleri üzerinde saldırı tespitine yardımcı olan özelliklerin seçilmesi ve özelliklerin bir düzene göre sıralanması oluşturulan model için önem taşımaktadır. Saldırı verilerinin ezberlenmesi, gelen yeni saldırıların tespit edilmesini önleyen aşırı öğrenme problemini ve modelin sonuca hiç ulaşamamasına neden olan eksik öğrenme problemini önlemek için veriseti ve üzerinde yapılan iyileştirmeler büyük önem taşımaktadır. Ayrıca kullanılan yöntemler ile sonuca ulaşma süresi saldırı tespiti için önemli bir detaydır. Çünkü saldırının tespit süresi, aynı zamanda saldırının kontrol altına alınması ve önlenmesi için gereken zaman ile doğru orantılıdır.

Bu bölümde, önerilen anomali tabanlı saldırı tespit metodolojisi beş adımda açıklanmıştır:

- DDoS Saldırı Tespit Topolojisinin Tasarımı
- DDoS Saldırı Çeşitlerinin Belirlenmesi
- DDoS Saldırı Araçlarının Belirlenmesi
- DDoS Veriseti Oluşturulması
- Makine Öğrenimini Kullanarak Önerilen DDoS Saldırı Tespit Yöntemleri

3.1 DDoS Saldırı Tespit Topolojisinin Tasarımı

Saldırganların DDoS saldırılarını gerçekleştirebilmesi için ilk kural, DDoS saldırı elemanlarının kurban olarak seçtikleri cihaza erişimlerinin sağlanmasıdır. İlgili IP adresi ve arabirime erişimler sağlandıktan sonra, saldırı trafiğini kurban cihaza aynı anda yönlendirilebilir. DDoS saldırısının etkisi gelen paket sayısına ve paketlerin kurban cihaza ulaşma sürelerine göre artmaktadır. Saldırının elemanları olan cihazların sayısı, yapılan saldırının boyutuna ve amacına göre değişmektedir. Bu sayı aynı zamanda saldırıyı yöneten kişi ya da grupların ellerindeki güce ve yeteneklerine bağlıdır.

Saldırı trafiğinin yönlendirildiği kurban cihazlar, çeşitli yazılım ve araçlar kullanarak, arabirimlerine gelen trafik ile ilgili fiziksel katmandan uygulama katmanına kadar tüm bilgileri izleyebilirler. Network trafiğini izlemek için kullanılan araçlar sayesinde gelen paketlerin kaynak IP adresleri, aynı kaynak IP adresinden kaç paket geldiği, seri şekilde gelen paketlerin ne zaman gönderilmeye başlandığı, ne kadar süre ile devam ettiği, hangi arabirimlere paketlerin yönlendirildiği gibi bilgiler elde edilebilirler.

Bu bilgiler paketin bir özelliğini temsil etmektedir. Bütün özellikler aslında bir matematik denklemindeki değişkenlerdir. Bu değişkenler, denklemin sonuçları olarak beklenen saldırı çeşitlerinin bulunması için, uygulanması gereken işlemler hakkında ipucu vermektedirler.

Saldırgan cihazlar tarafından erişebilir bir kurban cihaza yönlendirilen saldırılar çeşitlerine göre kaydedilerek, eğitim verisi olarak kullanılabilir. Bilinçli saldırı paketi gönderimi sıralı şekilde yapılarak farklı çeşitlerde DDoS saldırıları bir veriseti haline

getirilebilir. Bu işlemlerin gerçekleşmesi için birbirine erişebilen bir kurban ve iki saldırgan olmak üzere minimum üç cihaz yeterli olacaktır.

Daha sonra saldırı paketlerinin gönderilmesi amacı ile farklı DDoS saldırılarına göre belirlenen saldırı yazılım araçları kullanılabilir. DDoS saldırılarını gerçekleştirmek için kullanılan test amaçlı birçok DDoS saldırı aracı bulunmaktadır. Bu saldırı araçları amaçlarına yani gerçekleştirmek istedikleri DDoS saldırı çeşidine göre farklılık göstermektedir. Veriseti oluşturmak için saldırgan olarak belirlenen cihazlardan gelen farklı DDoS saldırıları, kurban bilgisayar tarafından ağ izleme sistemi kullanılarak izlenebilir, bu sayede gerekli olan özelliklerden oluşan verisetleri kolaylıkla oluşturulabilir. Boyutu azaltılan paketler üzerinde uygulanan yöntemler aynı zamanda çalışma süresi için faydalı olacaktır.

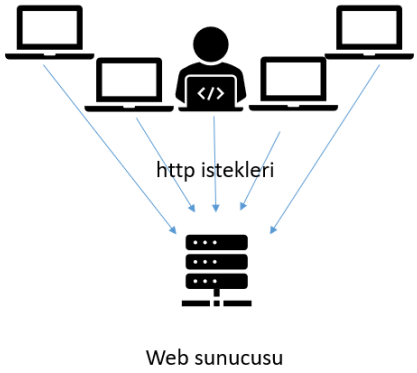
3.2 DDoS Saldırı Çeşitlerinin Belirlenmesi

DDoS saldırını tespit etmek için birçok saldırı çeşidi örneğini verisetinde bulundurmak, tespit oranlarının artmasını sağlayacaktır. Her DDoS saldırı çeşidinin farklı bir amacı, sistemi erişilemez hale getirmek için kullandıkları farklı bir yol bulunmaktadır. Tasarlanan topolojide Şekil 1.1.'de bahsedilen ve en çok karşılaşılan DDoS saldırı çeşitleri tercih edilebilir. Saldırı çeşitlerine dikkat edildiğinde; ilki uygulama tabanlı DDoS saldırısı olan HTTP flood saldırısıdır (Çizelge 3.1).

Çizelge 3.1 DDoS saldırı çeşitleri

Hacim Tabanlı Saldırılar	Protokol tabanlı Saldırılar	Uygulama Tabanlı
ICMP flood	TCP SYN Flood	DNS flood
UDP flood	Ping of Death	HTTP flood

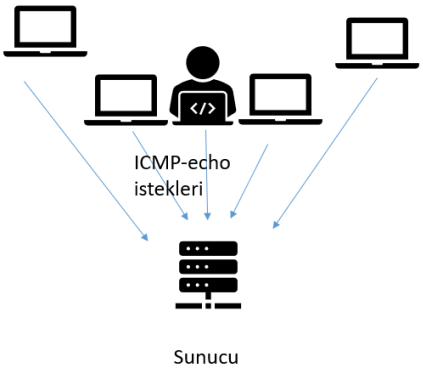
Uygulama katmanı protokolü HTTP'nin bir mağdura saldırmak için kullanıldığı kaynak tüketme saldırılarına bir örnektir. HTTP flood saldırılarında, saldırgan sunucuyla veya belirli bir uygulamayla konuşurken 'HTTP GET' ve 'HTTP POST' isteklerini değiştirir (Şekil 3.1). Bu saldırıyı gerçekleştirmek için geçerli IP adresiyle bir TCP bağlantısı kurulması gerekir. Saldırgan, bağlantıları kurmak için saldırı ajanlarının IP adreslerini kullanır (Mahjabin vd. 2017).



Şekil 3.1 HTTP flood DDoS saldırısı

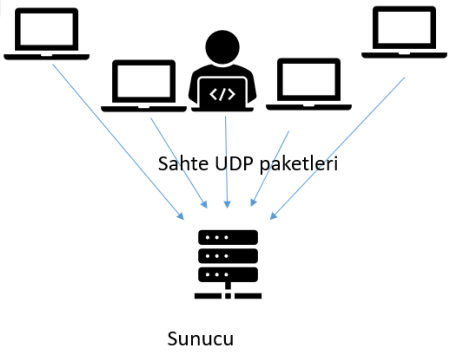
Diğer saldırı tipi ICMP flood saldırısıdır. Kurban cihaza sürekli olarak 'PING' paketleri göndererek bant genişliğinin tükenmesine sebep olan hacim tabanlı bir DDoS saldırısıdır. Ping flood saldırısı olarak da bilinen saldırı IP katmanından yararlanır. 'ICMP_ECHO_REQUEST' paketleri, uzak bir makinenin erişilebilir olup olmadığını kontrol etmek için kullanılır. DDoS saldırılarında, saldırgan bu paketi yayın IP adresini kullanarak gönderir. Böylece, mağdurun ağındaki tüm makinelere teslim edilir. Makineler, kurbanı hedefleyen sahte kaynak adresine 'ICMP_ECHO_REPLY' paketi ile yanıt verecektir. Sonuç olarak, mağdurun ağındaki bant genişliği tükenir ve sonuç olarak kurban sistem meşru kullanıcıların isteklerine cevap veremez (Mahjabin vd. 2017) (Şekil 3.2).

UDP flood, ICMP flood gibi bant genişliğini tüketmeyi amaçlayan DDoS saldırısıdır. UDP protokolünün birim zamanda daha fazla paket gönderimi sağlaması, sırasız ve düzensiz paket iletimi gibi avantajları sayesinde uygulanan saldırıdır. UDP flood saldırısı, saldırganın saldırı ordusundan büyük bir UDP paketi akışı gönderdiği çok yaygın bir DDoS saldırısıdır. Burada saldırgan, kurbanın belirli veya rastgele ser-



Şekil 3.2 ICMP flood DDoS saldırısı

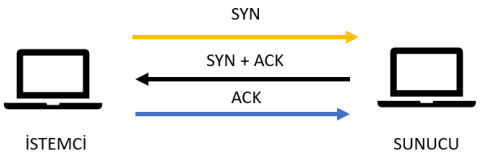
visini hedefleyebilir. Genel olarak, bir sistem UDP paketi alındığında, hedef bağlantı noktasında bekleyen uygulamanın türünü belirlemeye çalışır. Saldırgan sahte IP adresleri kullanır ve tüm bant genişlikleri tüketilene ve kurban normal çalışmayı sonlandırana kadar paketleri göndermeye devam eder (Mahjabin vd. 2017) (Şekil 3.3).



Şekil 3.3 UDP flood DDoS saldırısı

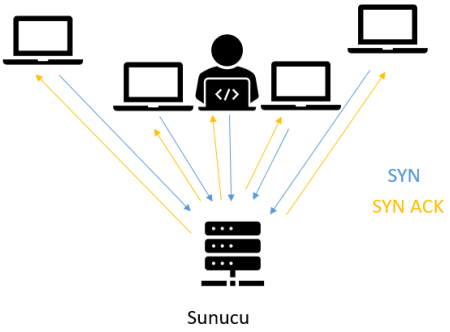
Son olarak incelenen saldırı çeşidi, 2020 yılının ilk çeyreğinde en çok karşılaşılan TCP SYN flood saldırısıdır. Bu saldırı, üçüncü ve dördüncü katman protokollerinin niteliklerinden faydalanarak uygulanan protokol tabanlı saldırı örneğidir (Çizelge 1). Saldırgan, TCP bağlantı kurma işleminin üç yollu el sıkışma mekanizmasından yararlanır. Uygulanması sırasında TCP bağlantısı oluşturmak isteyen iki taraf arasında

birbirini takip eden onaylar gereklidir. Bu, üç yönlü el sıkışma ile gerçekleştirilir (Şekil 3.4).



Şekil 3.4 TCP three-way handshake

Üç yönlü el sıkışmada, ilk önce SYN paketi bir istemciden bir sunucuya gönderilir (Mahjabin vd. 2017). Sunucu, bir istemciden ilk isteği aldıktan sonra, SYN/ACK paketi geri gönderir ve istemcinin ACK'yi göndermesini bekler. Bununla birlikte, karşılık gelen ACK'leri göndermeden, ilk SYN'lerin bir bölümünü göndererek, sunucuyu var olmayan ACK'ları beklemesini sağlamak mümkündür. Sunucunun yeni bağlantılar için sınırlı arabellek kuyruğu olduğu düşünüldüğünde, SYN Flood (Şekil 3.5), bir süreden sonra sunucunun diğer gelen bağlantıları işleyememesine neden olur (Paharia ve Bhuskan 2018).



Şekil 3.5 TCP SYN flood DDoS saldırısı

Yukarıda anlatılan DDoS saldırı çeşitleri günümüzde en çok görülen ve kullanılan DDoS saldırı çeşitleridir. Bu saldırılar aynı zamanda amaçlarına göre üç kola ayrılan DDoS saldırılarının örnekleridir. Verisetinde bulunan bu saldırı çeşitleri, hem günümüzde görülen saldırıların incelenmesine, hem de farklı amaçlarına göre ayrılan DDoS saldırılarının mekanizmasının analiz edilmesine yardımcı olacaktır.

3.3 DDoS Saldırı Araçlarının Belirlenmesi

DDoS saldırı çeşitlerinin belirlenmesinden sonra, bu saldırıların uygulanması için kullanılacak olan saldırı araçlarının belirlenmesi gereklidir. Günümüzde test ya da uygulama amaçlı kullanılan birçok DDoS saldırı aracı bulunmaktadır.

TCP SYN flood saldırısı için DDoSIM ve Torshammer araçları tercih edilebilir. DDoSIM, TCP bağlantısıyla sahte IP adreslerini kullanan DDoS saldırı aracıdır (Behal ve Kumar 2017), (Furtuna 2014). Torshammer ise TOR ağı üzerinden çalışan rasgele kaynak IP adresi kullanan saldırı aracıdır (Solarstone 2015).

HTTP flood saldırısı için HULK ve PyLoris araçları kullanılabilir. HULK, HTTP sunucusunun yükünü 'GET' istekleri ile doğrudan etkiler. Her istek için yönlendireni gizleyebilecek, farklı saldırı istek kalıpları gönderme yeteneğine sahiptir (Bargot ve Kulkarni 2019) ve bu sayede gerçek kullanıcı aracısını gizleyebilir. PyLoris komut dosyası tabanlı bir araçtır ve belirli bir Hizmet Reddi saldırısı sınıfına yönelik hizmet düzeyi güvenlik açığını sınamak için kullanılır (Motoma 2013).

Hacim tabanlı DDoS saldırısı ICMP flood için, rastgele paket boyutunu ve paket parçalanmalarını işleyebilen, etkin şekilde kullanılan HPING tercih edilebilir (Vanney 2020).

Son olarak, UDP flood saldırısı simülasyonu için Unicorn ve LOIC araçları kullanılabilmektedir. Unicorn, belirli bir hedefe saldırı için UDP paketleri oluşturmak için kullanılır. Saldırıyı gerçekleştirmek için hedefin IP adresini, bağlantı noktası numarasını, paket boyutu gibi bazı bilgilere ihtiyaç duyar ve rastgele modu etkinleştirilerek gerçek kullanıcı gizlenebilir (Paharia ve Bhuskan 2018).

Yukarıda anlatılan DDoS saldırı araçları, DDoS saldırıları hakkında araştırma yapan birçok kişiye yardımcı olmaktadır. Pek çok İnternet sitesinde ve araştırma makalelerinde kullanılan araçlar topolojide bulunan saldırgan cihazlar üzerinde kurulmalıdır.

3.4 DDoS Veriseti Oluşturulması

Seçilen DDoS saldırı araçları ve farklı DDoS saldırı verileri kullanılarak, veriseti oluşturmak için öncelikle kullanılması planlanan özelliklerin belirlenmesi gereklidir. Gelen ağ trafik bilgileri incelendiğinde, kaynak IP adres bilgisi paketlerin hangi IP

üzerinden geldiği bilgisi ile birlikte, aynı kaynaktan kaç tane paket alındığı bilgisi için de kullanılabilir. Verilerin içinde bulunan zaman bilgisi ile paketler için iletişimi başlatma ve aralarında bağlantının süresi hakkında bilgiler alınabilir. Aynı kaynaktan tekrarlı alınan paketler için süre bilgisi, saldırı tespitini kolay hale getirir. Her paketin ve segmentin uzunluğu, kullanılan protokole ve uygulanan işleme göre değişiklik göstermektedir. UDP ve TCP paketlerinin uzunluğunun aynı olmaması buna bir örnektir. Alınan uzunluk değerleri ile DDoS saldırılarının çeşitleri tespit edilebilir. TCP üç yollu el sıkışma bayrak değerleri ve sıralama özellikleri ile gelen TCP trafiğinin TCP SYN Flood saldırısını, gelen ACK paketi sayısı ve aynı kaynak IP adresine gönderilen sıralama numaralamaları ile tespit edebilir. Hedef olarak verilen arabirim numaraları trafiğin geldiği noktayı tespit etmeyi ve bu sayede hangi servisin saldırının hedefi olduğunu belirlemeye yardımcı olmaktadır. İlgili olan özellikler seçildikten ve veriler üzerinde filtreleme yapıldıktan sonra saldırı çeşitlerine göre etiketleme yapılmalıdır. Etiketleme işlemi daha öncesinde belirtildiği gibi, denklemde ulaşmak istenilen sonuçtur. Etiketlenen veriler, örnek çözümlü denklemler yani eğitim veriseti olarak kullanılabilmektedir. Oluşturulan eğitim verisetinde uygulanan ve sonuçları bulmak için geliştirilen yöntemlerin test edilmesi için anlık verilerin toplanarak etiket olmadan bir test veriseti olarak kaydedilmesi gereklidir. Bölüm 2'de yapılan çalışmalarda anlık verilerin toplanması ve kaydedilmesi için birçok kütüphane ve protokolden bahsedilmiştir. Kullanılan çeşitli kütüphaneler yardımı ile toplanan veriler ile çözümsüz denklemlerden oluşan bir veriseti yani test veriseti elde edilebilir. Oluşturulan eğitim ve test verisetleri üzerinde makine öğrenmesi yöntemleri kullanılarak ön işleme ve çeşitli algoritmalar uygulanabilir. Eğitim veriseti üzerinde çeşitli saldırılar için uygulanan çözüm yolları test verisetinde uygulanır. Ayrıca makine öğrenmesi algoritmaları kullanılarak bulunan çözümler ile eğitim ve test verisetleri üzerinde çalışma süreleri hesaplanabilir, anlık saldırı tespiti için uygun olan çözüm elde edilebilir.

3.5 Makine Öğrenimini Kullanarak Önerilen DDoS Saldırı Tespit Yöntemleri

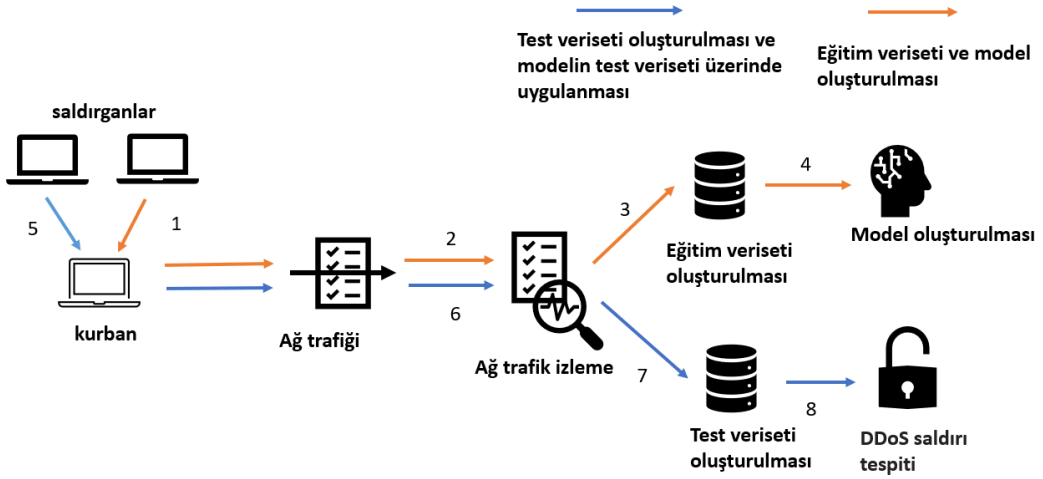
Anomali tabanlı saldırı tespit yöntemlerinin önemli diğer adımı, verisetlerinde gerçekleştirilen işlemlerlerdir. Makine öğrenimi yöntemleri, gelen ağ trafiğinin bir modelini oluşturmak için verisetlerini kullanır. Bir ağ izleme cihazı yardımıyla, gelen trafik izlenebilir ve verisetlerinin hazırlanması için veriler kaydedilebilir. Kaydedilen veriler, veri sahipleri tarafından belirli filtrelemelerden ve etiketlemelerden geçerek bir eğitim veriseti olarak saklanır. Saldırıların hedefi olan kurban cihaz üzerinden anlık olarak toplanan veriler ile test veriseti oluşturulur. Test verisetindeki paketler etiketsizdir. Eğitim veriseti ve çeşitli makine öğrenimi yöntemleri, test verisetindeki bilgilere ve sonuçlara ulaşmak için kullanılır. Makine öğrenimi yöntemleriyle ulaşılan tespit oranlarını artırmak için toplanan veriler üzerinde çeşitli ön işlemler yapılmalıdır.

Önerilen metodoloji özetlendiğinde, DDoS saldırılarının en kısa sürede ve doğru olarak tespit edilebilmesi için, doğru iletişim topolojisi oluşturmak ve topoloji içerisinde yer alan saldırgan cihazlar ile kurban cihazı doğru konumlandırmak önemli bir adımdır. Saldırgan cihazlar, DDoS saldırılarını kurban cihazın üzerinde çalışan ve erişime açık olan arabirimi üzerinden yapmaktadırlar. Aynı şekilde, kurban cihazlar, erişime açık arabirimler üzerinden saldırganların bilgilerine erişebilmektedirler. Diğer adımda, saldırgan cihazlar üzerinden gönderilmesi planlanan DDos saldırı çeşitleri, en çok rastlananlar arasından seçilmelidir. Bu sayede oluşturulan verisetlerinin ve saldırı tespit modelininin kullanım yelpazesi genişlemektedir. Daha sonra, saldırı trafiklerinin oluşturulması için dünyada en çok kullanılan ve uyguladıkları saldırı çeşidine göre farklı çalışma mekanizmaları bulunan araçlar tercih edilmelidir. Bir sonraki işlem olarak, oluşturulan saldırı trafiği üzerinden, saldırıları tespit etmemiz için en yararlı olan özellikler seçilmelidir. Böylece, yanlış tespit edilmesi ya da hiç tespit edilememesi gibi problemlerin riski azaltılabilir. Seçilen özelliklerden oluşan, filtrelenmiş saldırı paketleri, saldırı modeli oluşturmak için kullanacağımız eğitim ve test verisetleri olarak kayıt edilmelidir. Son olarak, oluşturulan eğitim veriseti üzerinde makine öğrenimi yöntemi kullanılarak oluşturulan model, ezberleme deni-

len aşırı öğrenme ya da öğrenememe problemlerinden kaçınılarak, test verisetinde uygulanarak test edilmelidir.

4. ÖNERİLEN YÖNTEMİN UYGULANMASI

Çalışmanın bu bölümünde, Bölüm 3'te önerilen metodoloji adımlarının uygulamaları yapılmış ve elde edilen sonuçlar verilmiştir (Şekil 4.1).



Şekil 4.1 Önerilen methodolojinin aşamaları

1. Adım- Oluşturulan metodolojinin ilk adımında saldırganlar tarafından kurban cihaza farklı DDoS saldırı çeşitlerinden oluşan ağ trafiği ve normal ağ trafiği gönderilmiştir.

2. Adım- İkinci aşamada, gönderilen ağ trafiği, kurban cihaz üzerine kurulu olan ağ trafik izleme programı aracılığı ile kaydedilmiştir .

3. Adım- Üçüncü aşamada her saldırı çeşidi için ayrı olarak kaydedilen ağ trafiği saldırı çeşitlerine göre etiketlenerek eğitim veriseti oluşturulmuştur.

4. Adım- Dördüncü adımda oluşturulan eğitim veriseti üzerinde bulunan sayısal olmayan veriler Scikit-Learn (Anonymous 2007) kütüphanesi içerinde tanımlı olan 'LabelEncoder' fonksiyonu ile numaralandırılmıştır. Gerekli iyileştirmeler yapılan eğitim veriseti üzerinde, farklı ön işlemler, ilk önce sırayla daha sonra aynı anda uygulanmıştır. En son aşamada farklı makine öğrenmesi algoritmaları ile saldırı tespit modeli oluşturulmuştur.

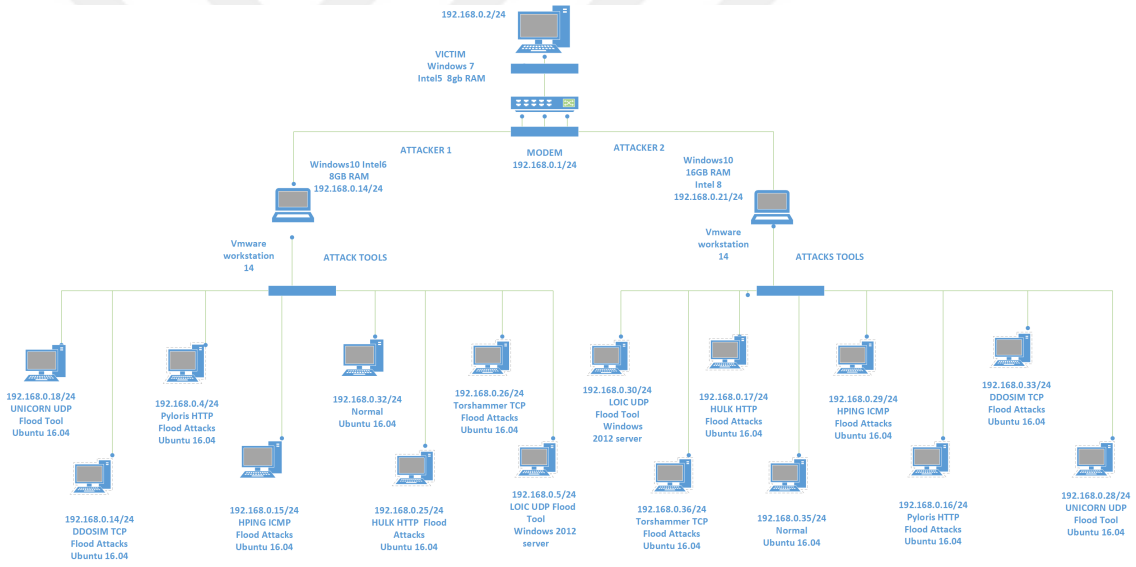
5.-6.-7. Adım- Diğer adımda anlık gelen saldırı trafiği ağ trafik izleme cihazı ile

toplanarak etiketsiz olarak test veriseti oluşturulmuştur.

8. Adım- Son olarak oluşturulan test verisetleri üzerinde, farklı ön işleme yöntemleri ve makine öğrenmesi algoritmaları ile oluşturulan model uygulanmıştır. Uygulanan yöntemler sonunda elde edilen saldırı tespit, kesinlik, duyarlılık oranları ve saldırı tespit süreleri kayıt edilmiştir.

4.1 Önerilen Topoloji Tasarımı

Yukarıdaki adımlar, bu konuda yapılan çalışmalar, saldırı çeşitleri ve araçları dikkate alındığında, DDoS saldırı çeşitlerini içeren bir eğitim veriseti oluşturmak ve daha sonra makine öğrenmesi algoritmaları ile saldırıları tespit etmek için Şekil 4.2'te görülen topoloji oluşturulmuştur.



Şekil 4.2 DDoS saldırı veriseti oluşturma ve saldırı tespit topolojisi

Topoloji incelendiğinde, üç adet dizüstü bilgisayar modem cihazı ile birbirine bağlanmıştır. Topolojide üst bölümünde yer alan dizüstü bilgisayar kurban cihaz olarak seçilmiştir. UDP trafiği için 53 (DNS), TCP trafiği için 25 (SMTP) ve HTTP trafiği için 443 arabirimleri ayarlanmıştır ve bu arabirimlerden takip sağlanmıştır. Saldırgan cihazlar üzerinde Windows 10 işletim sistemi ve işlemci olarak Intel i6 ve i8 bulunmaktadır. Bu cihazlar üzerine DDoS saldırı araçlarının kurulması için VMware 14 sanallaştırma sistemi kurulmuştur. Daha sonra VMware programı üzerine

köprü (bridged) ağ ile Windows Server 2012 ve Ubuntu 16.04 işletim sistemlerine sahip sekiz adet sanal cihaz kurulmuştur.

4.2 DDoS Saldırı Türlerinin ve Araçlarının Topolojiye Uygulanması

UDP flood DDoS saldırısı için tercih edilen UDP flood saldırı araçları LOIC ve UNICORN, Windows 2012 server işletim sistemi üzerine kurulmuştur. HTTP flood için kullanılan HULK ve PyLoris saldırı araçları, TCP SYN flood için kullanılan Torshammer ve DDoSIM saldırı araçları, ICMP flood için kullanılan HPING saldırı aracı Ubuntu 16.04 işletim sistemi üzerinde kurulmuştur. DDoS saldırı araçları uyumlu olarak çalıştıkları işletim sistemlerinin üzerine kurulmuştur. Son olarak bir adet sanal cihaza normal trafik örneği için bir adet saldırı aracı kurulmamıştır. Normal trafik için bırakılan cihaz üzerinde Ubuntu 16.04 işletim sistemi bulunmaktadır.
Kullanılan saldırı araçları, saldırı çeşitlerine göre göre gruplanmıştır. İlk grupta HTTP flood için kullanılan HULK ve PyLoris saldırı araçları, ikinci grupta ICMP flood için kullanılan HPING saldırı araçları bulunmaktadır. Üçüncü grupta normal trafik verileri için kurulan sanal cihazlar bulunmaktadır. Dördüncü grupta TCP flood için kullanılan, Torshammer ve DDoSIM araçlarını bulunduran ve son grupta UDP saldırı araçları, UNICORN ve LOIC bulunduran sanal cihazlar bulunmaktadır.

4.3 Eğitim ve Test Verisetlerinin Oluşturulması

Eğitim ve test verisetlerini oluşturmak için, topoloji üzerinde bulunan saldırı araçlarının bulunduğu sanal makineler ve kurban bilgisayar kullanılmıştır. İlk olarak alınan 'pcap' dosyaları, 'csv' formatına dönüştürülerek, saldırı çeşidine göre etiketlenmiş ve eğitim veriseti oluşturulmuştur (EK 1). Paket gönderimi sonunda, 'csv' dosyalarından Çizelge 4.1'de yer alan özellikler seçilmiştir.
Diğer adımda, oluşturulan eğitim verisetinde kullanılan yöntemlerin uygulanması için anlık verilerin toplanarak etiket olmadan test veriseti olarak kaydedilmesi gereklidir. Bölüm 2'de yapılan çalışmalarda anlık verilerin toplanması için birçok kütüphane ve protokolden bahsedilmiştir. Topoloji üzerinde veriseti oluşturmak için ilk olarak saldırı araçları sırasıyla çalıştırılmıştır.

Çizelge 4.1 Veriseti içerisindeki özellikler

Seçilen Özellikler	Seçilen Özelliklerin Tanımları
Time	Paket süresi hesaplamaları için başlangıç noktasıdır.
Source	Paketin geldiği kaynak IP adresini belirtir.
Destination	Paketin gönderilmesi istenen hedef IP adresini belirtir.
UDP Source Port	UDP paketinin geldiği arabirim numarasını belirtir.
UDP Destination Port	UDP paketinin gönderilmesi istenen arabirim numarasını belirtir.
TCP Source Port	TCP paketinin geldiği arabirim numarasını belirtir.
TCP Destination Port	TCP paketinin gönderilmesi istenen arabirim numarasını belirtir.
Protocol	IP paketine göre kullanılan protokolü belirtir.
Length	Ethernet çerçeve uzunluğunu vermektedir.
Total Length	IP paketi uzunluğunu vermektedir.
TCP Length	TCP segment uzunluğunu verir.
UDP Length	UDP segment uzunluğunu verir.
Rel Start	İki uç arasında yakalanan ilk pakete göre konuşmanın zamanını gösterir.
Header Length	TCP başlığı uzunluk bilgisini verir.
SYN flag	TCP 3 yönlü el sıkışma bayrağı, istemciden gönderilen ilk TCP bağlantı isteğidir.
ACK flag	TCP, 3 yönlü bir el sıkışmada bir TCP bağlantısının kurulduğunu gösteren bayraktır.
FIN flag	TCP bağlantısını sonlandırmak için TCP bağlantısında gönderilen bayraktır.
RESET flag	TCP bağlantısının yeniden başlatılmasına izin veren bayraktır.
Delta time	İstemci ve hedef arasındaki paket iletim süresini gösterir.
TTL	Bir paketin ağ üzerinden geçebileceği maksimum uç sayısını belirtir.

Çizelge 4.1'de verilen özellikler ve görevleri incelendiğinde; 'Time', 'Rel Start' ve 'Delta Time' özellikleri bağlantının başlama ve aralarında bağlantının süresi hakkında bilgiler vermektedir. Uzun süredir devam eden bağlantı istekleri, sistemler tarafından tehlike anlamına gelmektedir. Ayrıca aynı kaynaktan tekrarlı alınan paketler için süre bilgisi saldırı tespiti yapılmasını kolay hale getirir.

'Length', 'Total Length', 'TCP Length', 'UDP Length', 'Header Length' özellikleri ile alınan uzunluk bilgileri DDoS saldırının ve DDoS saldırı çeşidinin tespiti için önemlidir. Her paketin ve segmentin uzunluğu kullanılan protokole ve uygulanan işleme göre değişiklik göstermektedir. Alınan paket, segment ve başlık uzunluk değerleri ile saldırıların çeşitlerini tespit edebiliriz. TCP üç yollu el sıkışma bayrak değerleri 'SYN_flag', 'FIN_flag', 'ACK_flag', ve 'RESET_flag' ile gelen TCP trafiğindeki TCP SYN flood saldırısını tespit edebiliriz. Sıralı halde, sürekli olarak gelen SYN istekleri ve ona eşit sayıda gelmeyen ACK paketleri, sunucunun TCP SYN flood saldırısı altında olduğu anlamına gelmektedir. Kaynak ve hedef verilen IP adresleri trafiğin geldiği noktayı ve aynı kaynaktan gelen kaç tane istek olduğunun tespit edilmesine yardımcı olmaktadır. Aynı şekilde kaynak ve hedef arabirimleri sayesinde isteğin hangi birime iletildiği bilgisi ile DDoS saldırı çeşidi belirlenebilir.

Çizelge 4.1'de verilen 20 adet özellik yukarıda belirtilen sebepler ile seçilmiştir. Daha sonra '.pcap' dosyaları herhangi karmaşanın önlenmesi için hedef IP adresi kurban IP adresi olarak sabitlenerek '.csv' formatına dönüştürülmüştür, son olarak HTTP flood, TCP SYN flood, UDP flood, ICMP flood ve normal olarak etiketlenmiştir. Elde edilen beş adet '.csv' dosyaları, 20 adet özellik, 1 adet etiket sütunlarından oluşmaktadır ve model için eğitim veriseti olarak kullanılmıştır. Saldırı çeşitlerine göre etiketlenen veriler tek bir veriseti haline getirilmiştir. Homojen olarak dağılımın sağlanması için, beş verisetinin her birinden 5000 satır seçilmiştir.

Çalışmada, anlık verilerin toplanması için Scapy kütüphanesi kullanılmıştır. Scapy, ağ paketlerini işlemek için yazılmış Python programıdır. Philippe Biondi tarafından geliştirilmiştir. Scapy ile ağ paketleri gönderilebilir ve incelenebilir. Scapy, çeşitli protokollerin paketlerini oluşturmak veya şifresini çözmek, göndermek, yakalamak, istekleri ve yanıtları eşleştirmek ve çok daha fazlası için kullanılabilir. Metin tabanlı protokolleri (HTTP, HTTPS), Ethernet, TCP/IP, UDP, IPv4 ve IPv6, ARP, ICMP

ve ICMPv6, IGMP, STP, DNS, FTP ve TFTP protokollerini desteklemektedir (Rohitah vd. 2018).

Scapy kütüphanesi ile toplanan ağ paketleri boyutu 2500 olarak sınırlandırılmıştır. Böylelikle eğitim veriseti boyutunun onda biri boyutunda test veriseti oluşturulması sağlanmıştır. Oluşturulan '.pcap' dosyasından veriler IPv4 paket tipine ve ICMP, TCP ve UDP protokollerine göre filtrelenmiştir. Scapy ile daha önce eğitim veriseti oluşturmak için kullanılan 20 tane özellik, paketlerin IP protokol değerine göre seçilmiş ve daha sonra '.csv' formatında test veriseti olarak kaydedilmiştir. Scapy ile alınan ağ trafik verileri incelendiğinde TCP SYN, FIN, ACK ve RESET flag bilgisi Netflow hexadecimal değerler olarak geldiği görülmektedir. Bu problemin çözümü olarak Netflow TCP flag bilgilerini bulunduran bir sözlük oluşturulmuştur ve uygun değerleri göre 'Set' ve 'Not set' değerleri ile değiştirilmiştir.

Alınan anlık veriler üzerinden hedef IP adresi sunucunun IP adresi olmayanlar silinmiştir. Daha sonra eğitim ve test verisetinde bulunan sayısal değer olmayan değerler Scikit-Learn kütüphanesi fonksiyonlarından 'LabelEncoder' fonksiyonu kullanılarak, sayısal verilere dönüştürülmüştür Scikit-Learn (Anonymous 2007). 'NaN' yani boş olan sayısal veriler sıfır (0) ile doldurulmuştur ve bütün veriler 'float' sayısal veri tipine dönüştürülmüştür.

4.4 Ön İşleme ve Makine Öğrenimi Yöntemlerinin Uygulanması

Önerilen metodolojinin son aşamasında oluşturulan eğitim ve test verisetleri üzerinde, oluşturulan modellerin uygulanması için saldırı senaryoları oluşturulmuştur. Saldırı senaryolarının uygulanması için ilk olarak topolojide yer alan LOIC DDoS saldırı aracının yüklü olduğu sanal cihazlar saldırgan cihazlar olarak belirlenmiştir. LOIC saldırı aracı UDP dışında HTTP, ICMP ve TCP saldırılarını da uygulayabilmektedir. Bu özellikten faydalanılarak her iki bilgisayarda bulunan LOIC yüklü sanal cihazların IP adresleri değiştirilmiştir. Aynı şekilde içerisinde bir saldırı aracı bulunmayan sanal cihaz IP adresi değiştirilerek meşru kullanıcı olarak çalışması sağlanmıştır.

Saldırgan cihazlardan gelen dört farklı saldırı ve meşru kullanıcıdan gelen normal ağ paketleri, değerlendirilmek üzere kurban bilgisayara yönlendirilmiştir. MLP, RF,

KNN ve SVC makine öğrenmesi algoritmaları ve ön işleme yöntemleri olarak ele alınan özellik azaltma ile ölçeklendirme ayrı ve birlikte olmak üzere oluşturulan iki veriseti üzerinde uygulanmıştır. Bu işlemler için Scikit-Learn ve Pandas (Anonymous 2020) kütüphaneleri kullanılmıştır.

Yapılan işlemler sonunda, saldırı tespit oranları, kesinlik (precision) değeri, duyarlılık (recall) değeri ve her iki değerin harmonik ortalaması olan F1-skor (f1-score) değeri hesaplanmıştır. Hesaplanan değerler sonucunda modelin başarısı gözlemlenmiştir.

Kesinlik değeri, saldırı olarak etiketlediğimiz veriler (TP (True-Positive) + FP (False-Positive)) ile gerçekte saldırı olan ve saldırı olarak etiketlediğimiz verilerin (TP) oranı hesaplanarak bulunur (Şekil 4.3).

$$Kesinlik = \frac{TP}{TP+FP}$$

Şekil 4.3 Kesinlik (precision) değerinin hesaplanması

Duyarlılık değeri, saldırı olan veriler (TP + FN (False-Negative)) ile gerçekte saldırı olan ve saldırı olarak etiketlediğimiz verilerin (TP) oranı hesaplanarak bulunur (Şekil 4.4).

$$Duyarlılık = \frac{TP}{TP+FN}$$

Şekil 4.4 Duyarlılık (recall) değerinin hesaplanması

F1-skor değeri, kesinlik ve duyarlılık değerlerinin harmonik ortalamasıdır (Şekil 4.5).

$$F1 - Skor = \frac{2 \times Duyarlılık \times Kesinlik}{Duyarlılık + Kesinlik}$$

Şekil 4.5 F1-skor (f-score) değerinin hesaplanması

4.4.1 Ön işlem olmadan makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması

Oluşturulan ilk senaryoda 4 farklı çeşidi HTTP flood, ICMP flood, TCP SYN flood ve UDP flood DDoS saldırı ve normal trafik verilerini tespit etmek için, verisetleri üzerinde herhangi özellik azaltma veya ölçeklendirme ön işlemi yapılmamıştır. Eğitim ve test verisetleri üzerinde sırası ile MLP, RF, KNN ve SVC algoritmaları uygulanmıştır. Yapılan işlem sonunda elde edilen tespit oranları ve tespit süreleri analiz edilmiştir. Çizelge 4.2'te görülen sonuçlar değerlendirdiğinde, veriler üzerinde özellik azaltma ya da ölçeklendirme işlemi yapılmadığında, RF ile UDP flood saldırısının ve SVC algoritması ile ICMP flood saldırıları tespit edilebilmiştir. Eğitim veriseti üzerinde ve test veriseti üzerinde çalıştırılan modellerin çalışma süreleri incelendiğinde; RF ve KNN algoritmalarının daha hızlı çalıştığı gözlemlenmiş olsa bile KNN algoritması tespit değeri için çok düşük bir değerde kalmıştır. Senaryo sonucunda kullanılan makine öğrenmesi algoritmaların tek başına başarısız olduğu açıkça görülmektedir. Birçok saldırı saldırı çeşidi, farklı algoritmalar için normal ağ trafiği olarak etiketlenmiştir. Herhangi bir öğrenme işlemi gerçekleşmemiştir. Bu durum sistemleri tehlikeye atmaktadır. Çalışma süreleri incelendiğinde, eğitim veriseti üzerinde MLP algoritması ortalama 2.285 saniyede saldırı tespiti yapabilmişken, RF algoritması ortalama 0.114 saniyede, KNN algoritması ortalama 0.166 saniyede, SVC algoritması ise ortalama 0.114 saniye sürede saldırı tespiti yapabilmiştir. Ayrıca; test veriseti üzerinde ortalama çalışma sürelerini incelersek; MLP algoritması ortalama 0.003 saniyede, RF algoritması ortalama 0.002 saniyede, KNN algoritması ortalama 0.001 saniyede ve SVC algoritması ortalama 0.98 saniyede saldırı tespiti yapabilmiştir. Tüm süreler karşılaştırıldığında RF algoritması en kısa sürede tespit yapabilmiştir. Saldırı tespiti için en uzun süre SVC algoritması için kaydedilmiştir.

Hazırlanan test veriseti anlık veriler ve etiketler yardımıyla 10'lu çapraz doğrulama (10-fold cross validation) yöntemi kullanılmıştır. K-çapraz doğrulama yöntemi ile eğitim ve test olarak veriseti parçalanırken, dağılıma bağlı olarak modelin eğitim ve test verisetinde bazı sapmalar ve hatalar oluşabilmesi engellenir. Veri belirlenen k sayısına göre eşit parçalara bölünür, her bir parçanın hem eğitim hem de test için

kullanılması sağlanır, böylelikle dağılım ve parçalanmadan kaynaklanan sapma ve hataları asgariye indirilir. Yapılan 10'lu çapraz doğrulama sonunda, uygulanan makine öğrenmesi tespit sonuçları incelendiğinde, yukarıdaki sonuçlara benzer şekilde, en iyi sonuç %40 ile RF sonucunda elde edilmiştir. Diğer algoritmalar içinde alınan değerler, MLP ve SVC için %20 iken, KNN için %21 olarak hesaplanmıştır. Yapılan çaprazlama sonucunda veriler üzerinde alınan tespit oranlarında yükseliş gerçekleşmemiştir. Hesaplanan kesinlik ve duyarlılık değerlerine bakıldığında (EK 2), %60 ile KNN sonucunda elde edilen kesinlik değeri diğer algoritmalara göre yüksek olsada F1-skor kontrol edildiğinde, %40 ile RF daha başarılı görülmektedir. Nedeni ise RF ile elde edilen duyarlılık oranı KNN'e göre daha yüksektir. SVC ve MLP için %20 ile en düşük F1-skor değerleri hesaplanmıştır.

Çizelge 4.2 Hazırlanan verisetleri üzerinde ön işlem yapılmadan alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 2.285 Saniye Ortalama Test Süresi : 0.003 Saniye	HTTP flood	%0
		ICMP flood	%0
		Normal	%100
		TCP SYN flood	%0
		UDP flood	%0
RF	Ortalama Eğitim Süresi: 0.114 Saniye Ortalama Test Süresi: 0.002 Saniye	HTTP flood	%0
		ICMP flood	%0
		Normal	%100
		TCP SYN flood	%0
		UDP flood	%100
KNN	Ortalama Eğitim Süresi: 0.166 Saniye Ortalama Test Süresi: 0.01 Saniye	HTTP flood	%0
		ICMP flood	%0
		Normal	%99
		TCP SYN flood	%0
		UDP flood	%0
SVC	Ortalama Eğitim Süresi: 174.53 Saniye Ortalama Test Süresi: 0.98 Saniye	HTTP flood	%0
		ICMP flood	%100
		Normal	%0
		TCP SYN flood	%0
		UDP flood	%0

4.4.2 Ölçeklendirme ön işlemi ile makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması

İkinci senaryoda verisetleri üzerinde Scikit-Learn ön işleme algoritmalarından 'MinMax()' fonksiyonu kullanılarak veriler 0 ve 1 arasında ölçeklendirilmiştir. Böylece verilerin eşit temelde karşılaştırılması sağlanmıştır. Uygulanan MLP algoritması sonucunda, HTTP flood, TCP SYN flood, UDP flood DDoS saldırıları tespit edilememiştir. ICMP flood tespit oranı %7 ve normal trafik verilerinin tespit oranı %92'dir. Çalışma süreleri incelendiğinde eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 0.191 saniye iken test veriseti harcanan süre 0.005 saniyedir. RF algoritması için alınan sonuçlar karşılaştırıldığında HTTP flood, TCP SYN flood ve UDP flood saldırıları tespit edilememiştir. ICMP flood tespiti için %53 ve normal trafik verileri için %25 tespit oranı hesaplanmıştır. Eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 20.37 saniye iken test veriseti harcanan süre 0.01 saniyedir. KNN algoritması için alınan sonuçlarda, HTTP flood ve UDP flood tespiti başarısız olmuştur. TCP SYN flood saldırısının tespit tespit oranı %53, ICMP flood saldırısı için %40 ve normal ağ trafiğini tespit tespit oranı %33'tür. Çalışma süreleri incelendiğinde eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 0.39 saniye iken test veriseti üzerinde harcanan süre 0.07 saniyedir. Son olarak SVC algoritması incelendiğinde, TCP SYN flood ve UDP flood saldırılarının tespit edilemediğini görülmektedir. HTTP flood saldırısı için tespit tespit oranı %6 iken, ICMP flood için %17'dir. Normal saldırısı için %84 tespit oranı kaydedilmiştir. Eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 4.685 saniye iken test veriseti harcanan süre 0.12 saniyedir. Ölçeklendirme işlemi uygulandığında SVC algoritması hariç diğer algoritmalar için tespit değerleri düşmüştür. RF için tespit değeri ortalama %40 iken %15.6'a gerilemiştir. Aynı zamanda eğitim ve test veriseti üzerinde çalışma sürelerinde bir artış gözlemlenmiştir. MLP algoritması için eğitim çalışma süresi neredeyse 10 katına çıkmıştır. SVC algoritması için eğitim çalışma süresi 174.53 saniyeden 4.685 saniyeye düşmüştür. Çizelge 4.3'te yer alan sonuç-

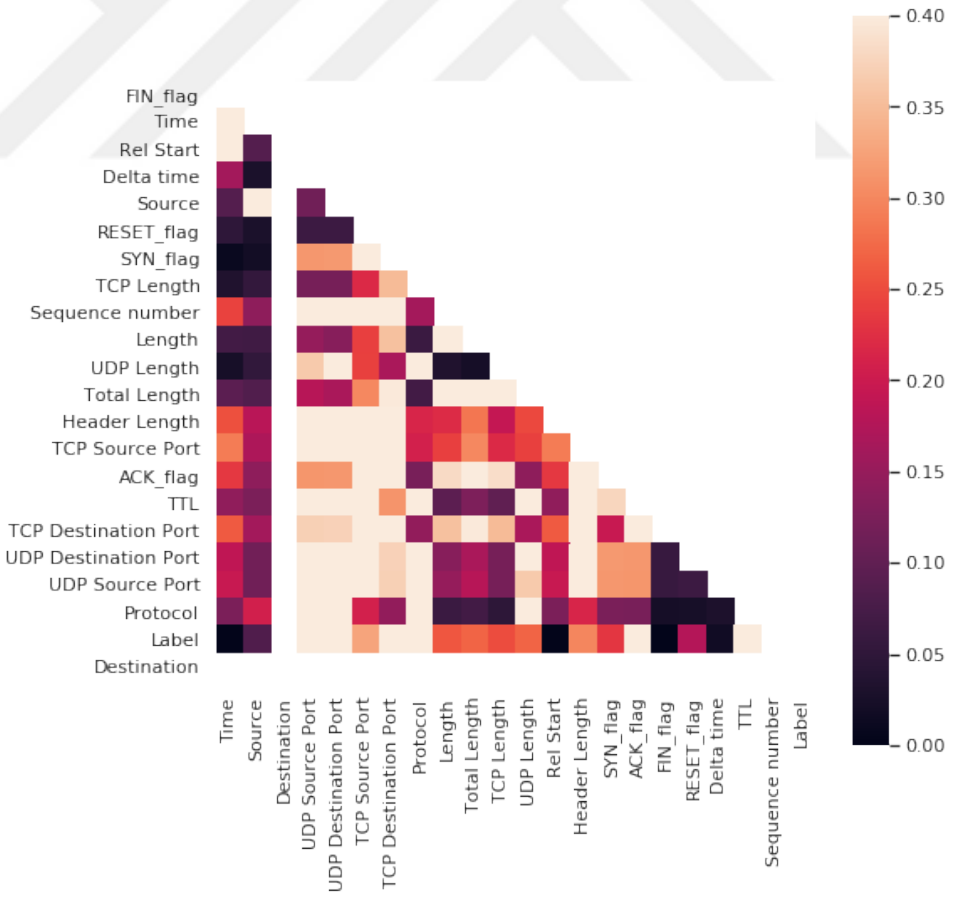
lar incelendiğinde kullanılan ölçeklendirme yöntemi, saldırı verilerinin normal trafik olarak etiketlenmesini engelleyememiştir. Bu yöntemin veriseti için uygun olmadığı açıkça görülmektedir. Ölçeklendirme ön işlemi sonunda modelin başarısını kontrol etmek için uygulanan çapraz doğrulama işlemi sonunda, tespit oranlarında ön işlemsiz adıma göre ciddi bir artış gözlemlenmemiştir. Tespit oranları arasında en yüksek değer KNN ile %28 olarak hesaplanmıştır. RF ile ön işlemsiz adımda hesaplanan tespit değeri %24'e düşmüştür. Fakat SVC için hesaplanan değer %20'den %28'e yükselmiştir. Kesinlik ve duyarlılık değerlerine bakıldığında, SVC için hesaplanan F1-skor başarı oranı %20 den %28'e yükselmiştir. Yapılan ölçeklendirme ön işlemi kesinlik değerlerinde tüm algoritmalar için bir yükselme meydana gelmiştir. Fakat düşük duyarlılık değerleri nedeni ile KNN ile hesaplanan F1-skor değeri %12 olarak hesaplanmıştır (EK 3).

Çizelge 4.3 Hazırlanan verisetleri üzerinde ölçeklendirme ön işlemi yapıldıktan sonra alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 20.37 Saniye Ortalama Test Süresi : 0.005 Saniye	HTTP flood	%0
		ICMP flood	%7
		Normal	%92
		TCP SYN flood	%0
		UDP flood	%0
RF	Ortalama Eğitim Süresi: 0.191 Saniye Ortalama Test Süresi: 0.01 Saniye	HTTP flood	%0
		ICMP flood	%53
		Normal	%25
		TCP SYN flood	%0
		UDP flood	%0
KNN	Ortalama Eğitim Süresi: 0.39 Saniye Ortalama Test Süresi: 0.07 Saniye	HTTP flood	%0
		ICMP flood	%40
		Normal	%33
		TCP SYN flood	%53
		UDP flood	%0
SVC	Ortalama Eğitim Süresi: 4.685 Saniye Ortalama Test Süresi: 0.12 Saniye	HTTP flood	%6
		ICMP flood	%17
		Normal	%84
		TCP SYN flood	%0
		UDP flood	%0

4.4.3 Özellik azaltma ön işlemi ile makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması

Veri ön işleme yöntemlerinden biri gereksiz olan özelliklerin verisetinden silinmesidir. Böylece doğru sonuca ulaşmak için ihtiyaç olmayan ya da doğru sonuca ulaşmayı engelleyen gereksiz veriler, verisetinden çıkarılabilir. Diğer senaryoda özellik azaltma işlemi uygulanarak, makine öğrenmesi algoritmaları uygulanmıştır. Her özellik ve sınıflandırma etiketi arasındaki korelasyon hesaplanmıştır, daha sonra özellik sıralaması yapılmıştır. Korelasyon ne kadar yüksek olursa, saldırı çeşidi hakkında daha fazla bilgi içeriğine sahiptir. En büyük korelasyona sahip özellikler ön plandadır ve en az korelasyona sahip özellikler ise arka plandadır. Bu nedenle, en büyük korelasyon veya en yararlı özellikler yüksek dereceye sahiptir (Long vd. 2019).



Şekil 4.6 Saldırı etkilerine göre korelasyon sonuçlarının ısı haritası

Senaryo için **'corr matrix=dataset:corr():abs():sort values(Label)'** işlemi uygulanarak veriseti üzerinde bulunan özelliklerin, saldırı etiketlerinin belirtildiği son satıra göre korelasyon değerleri hesaplanmıştır. Şekil 4.6'de yapılan işlem sonrasında oluşan ısı haritası kullanılarak elde edilen değerlere göre hangi özelliklerin saldırı etiket bilgisi ile ne kadar ilişkili olduğu saptanmaya çalışılmıştır. Yapılan korelasyon işlemine göre 20 özellik için bulunan değerler sırası ile kaydedilmiştir. Bulunan veriler Çizelge 4.4'tedir. Elde edilen değerlere göre korelasyon incelenmiş, en iyi çözümün elde edildiği değer ile özellik azaltma işlemi uygulanarak, 0.1 değerinden düşük olan özellikler verisetinden çıkarılmıştır. Verisetlerinden çıkarılan özellikler sırası ile aşağıdaki şekildedir: • Time • Source • Destination • Rel Start • Delta Time

Çizelge 4.4 Özellikler arası korelasyon hesaplama sonuçları

Time	0.000391
Source	0.083613
Destination	NaN
UDP Source Port	0.707108
UDP Destination Port	0.703446
TCP Source Port	0.327583
TCP Destination Port	0.600881
Protocol	0.708566
Length	0.258075
Total Length	0.270183
TCP Length	0.269910
UDP Length	0.25037
Rel Start	0.000391
Header Length	0.299556
SYN Flag	0.231269
ACK Flag	0.545457
FIN Flag	2.01309
RESET Flag	0.177797
Delta Time	0.018732
TTL	0.562017

Uygulanan işlem sonucunda elde edilen sonuçlara göre (Çizelge 4.5) en iyi tespit sonuçları MLP ve KNN algoritmaları ile elde edilmiştir. İlk olarak MLP algoritması incelendiğinde, HTTP flood ve TCP SYN flood saldırılarının tespit oranı %98 ve diğer saldırı çeşitleri olan ICMP flood, UDP flood ve normal trafik verisi için bulunan tespit oranı %100 olarak bulunmuştur. Eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 1.51 saniye iken, test veriseti harcanan süre 0.005 saniyedir. RF algoritması için HTTP flood saldırısının tespit oranı %97, normal trafik verisinin tespit oranı %69 ve TCP flood saldırısının tespit oranı %65 iken ICMP ve UDP flood için tespit oranı %100 olarak kaydedilmiştir. Bu oranlar MLP ile kıyaslandığında düşük kalmıştır. Ama ön işleme olmadan uygulanan RF ile kıyaslandığında yüksektir. Çalışma süreleri incelendiğinde eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 0.80 saniye iken test veriseti harcanan süre 0.002 saniyedir. Uygulanan KNN algoritması ile alınan sonuçları incelendiğinde, HTTP flood saldırısının tespit oranı %97, ICMP flood, UDP flood ve normal trafik verisi için bulunan tespit oranı %100, normal trafik verisi ve TCP SYN saldırısı için bulunan tespit değeri %98 olarak hesaplanmıştır. Çalışma süreleri incelendiğinde eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 0.141 saniye iken, test veriseti harcanan süre 0.006 saniyedir. En son uygulanan SVC algoritması ile bulunan tespit değerleri diğer yöntemlere göre düşüktür ve çalışma süreleri de daha uzundur. Sadece TCP SYN flood saldırısının tespitinde başarılı olunmuştur. Çalışma süreleri incelendiğinde eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 30.305 saniye iken, test veriseti harcanan süre 0.921 saniyedir. Senaryonun sonuçları incelendiğinde, test edilen saldırı çeşitleri için ortalama çalışma süreleri karşılaştırıldığında MLP algoritması ortalama eğitim süreleri 1.51 saniye, test süresi 0.005 saniye ve KNN algoritması için ortalama eğitim süresi 0.141 saniye, ortalama test süresi 0.006 saniyedir. Özellik azaltma işlemi uygulanarak uygulanan KNN algoritması için ortalama %98.6 ve MLP için ortalama %99.2 tespit oranları elde edilmiştir. Bu oranlar özellik azaltma işlemi yapılmadan ve ölçeklendirme işlemi yapıldıktan sonra alınan tespit oranlarına göre yüksektir. Ek olarak hem eğitim hem

test veriseti üzerinde hesaplanan çalışma sürelerinde düşüş görülmüştür. Uygulanan MLP algoritması için eğitim veriseti üzerinde çalışma süresi daha önce uygulanan iki yönteme göre düşmüştür ve 1.51 saniye sürmüştür. Aynı şekilde RF ve KNN algoritmaları için çalışma sürelerinde düşüş görülmüştür. SVC algoritması için ön işlem yapılmadan önce alınan eğitim süresine göre bir düşüş fakat ölçeklendirmeden sonra hesaplanan eğitim süresine göre artış gözlemlenmiştir. Çapraz doğrulama işlemi sonunda alınan tespit oranları incelendiğinde SVC (%40) hariç iyi sonuçlar kayıt edilmiştir. MLP için %99, RF için %70 ve KNN için %96 değeleri alınmıştır. Model başarısı incelersek (EK 4), RF için alınan kesinlik oranı (%60), MLP (%99) ve KNN (%80) algoritmalarına göre düşüktür ve SVC için alınan değere (%40) göre yüksektir. Aynı şekilde duyarlılık değerine bakıldığında RF için hesaplanan değer (%49), MLP (%99) ve KNN'e (%96) göre düşüktür ve SVC'ye (%40) göre yüksektir.

Çizelge 4.5 Hazırlanan verisetleri üzerinde özellik azaltma ön işlemi yapıldıktan sonra alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 1.51 Saniye Ortalama Test Süresi : 0.005 Saniye	HTTP flood	%98
		ICMP flood	%100
		Normal	%100
		TCP SYN flood	%98
		UDP flood	%100
RF	Ortalama Eğitim Süresi: 0.80 Saniye Ortalama Test Süresi: 0.002 Saniye	HTTP flood	%97
		ICMP flood	%100
		Normal	%69
		TCP SYN flood	%65
		UDP flood	%100
KNN	Ortalama Eğitim Süresi: 0.141 Saniye Ortalama Test Süresi: 0.006 Saniye	HTTP flood	%97
		ICMP flood	%100
		Normal	%98
		TCP SYN flood	%98
		UDP flood	%100
SVC	Ortalama Eğitim Süresi: 30.305 Saniye Ortalama Test Süresi: 0.921 Saniye	HTTP flood	%0
		ICMP flood	%0
		Normal	%84
		TCP SYN flood	%100
		UDP flood	%0

4.4.4 Ölçeklendirme ve veri azaltma ön işlemleri ile makine öğrenmesi algoritmalarının verisetleri üzerinde uygulanması

Son senaryoda, yapılan işlemde ön işleme olarak belirtilen ölçeklendirme ve özellik azaltma işlemleri aynı anda uygulamıştır (Çizelge 4.6). İlk olarak MLP algoritması ile alınan sonuçlarda, HTTP flood için alınan tespit oranı %47, normal trafik verisinin tespiti için alınan tespit oranı %33 iken, ICMP flood başarılı şekilde tespit edilmiştir. TCP SYN flood ve UDP flood tespiti başarısız olmuştur. Çalışma sürelerine bakıldığında eğitim süresi 42.92 saniye, test süresi 0.006 saniye olarak gözlemlenmiştir. RF algoritması incelendiğinde, TCP SYN flood ve UDP flood saldırılarının tespiti başarısız olmuştur. HTTP flood için %79, ICMP flood için %100, normal trafik verilerinin tespiti için %99 tespit oranı elde edilmiştir. Çalışma sürelerine bakıldığında eğitim veriseti üzerinde çalışma süresi 0.076 saniye, test veriseti üzerinde çalışma süresi 0.014 saniye olarak gözlemlenmiştir. Diğer yöntem olan KNN algoritması ile normal trafik verisi, TCP SYN flood ve UDP flood saldırılarının tespiti yapılamamıştır. HTTP flood için %97, ICMP flood için %100 tespit oranı elde edilmiştir. Çalışma sürelerine bakıldığında eğitim veriseti üzerinde çalışma süresi 0.56 saniye, test veriseti üzerinde çalışma süresi 0.018 saniye olarak gözlemlenmiştir. Son olarak SVC algoritması incelendiğinde, TCP SYN flood ve UDP flood saldırılarının tespiti yapılamamıştır. HTTP flood ve normal trafik verileri incelendiğinde alınan tespit oranı %98, ICMP flood için %99 tespit oranı elde edilmiştir.

Uygulanan her iki işlem sonucunda alınan tespit oranları MLP, RF ve KNN algoritmaları için sadece özellik azaltma işlemi uygulandığında alınan sonuçlara ve sadece ölçeklendirme uygulanan sonuçlara göre düşüktür. Sadece SVC algoritması için daha iyi sonuçlar elde edildiği gözlemlenmiştir. Tespit oranı ortalama %59'a yükselmiştir. Çalışma süreleri karşılaştırıldığında MLP algoritması için hesaplanan eğitim süresi ön işlem olmadan alınan, özellik azaltma işlemi uygulandıktan sonra alınan ve ölçeklendirme işlemi uygulandıktan sonra alınan değerlere göre yükselmiştir. Fakat RF algoritması için kaydedilen en düşük çalışma süresi elde edilmiştir. KNN algoritması içinde eğitim veriseti üzerinde alınan çalışma süreleri artmıştır. Son olarak SVC algoritması incelendiğinde, özellik azaltma işlemi sonunda alınan eğitim veriseti üze-

rindeki çalışma süresine göre bir düşüş, fakat ölçeklendirme işleminden sonra alınan süreye göre bir artış kaydedilmiştir. Yapılan çapraz doğrulama işlemi sonunda elde edilen tespit oranları, yukarıdaki sonuçlara benzer şekilde ölçeklendirme ve özellik azaltma ön işlemleri sonrasında alınan oranların arasında değerlerde hesaplanmıştır. SVC için ise %80 ile en yüksek saldırı tespit oranı elde edilmiştir. Modelin başarısı kontrol edildiğinde (EK 5), elde edilen kesinlik ve duyarlılık oranlarında SVC hariç düşüş meydana gelmiştir. MLP ve RF için hesaplanan kesinlik oranı %60 ve KNN için hesaplanan kesinlik oranı %40'tır. Duyarlılık değerleri ise sırası ile %29, %40 ve %20 olarak hesaplanmıştır. SVC için bu değerler %80 olarak kayıt edilmiştir.

Çizelge 4.6 Hazırlanan verisetleri üzerinde özellik azaltma ve ölçeklendirme ön işlemleri yapıldıktan sonra alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 42.92 Saniye Ortalama Test Süresi : 0.06 Saniye	HTTP flood	%65
		ICMP flood	%100
		Normal	%33
		TCP SYN flood	%0
		UDP flood	%0
RF	Ortalama Eğitim Süresi: 0.076 Saniye Ortalama Test Süresi: 0.018 Saniye	HTTP flood	%79
		ICMP flood	%100
		Normal	%99
		TCP SYN flood	%0
		UDP flood	%0
KNN	Ortalama Eğitim Süresi: 0.56 Saniye Ortalama Test Süresi: 0.014 Saniye	HTTP flood	%97
		ICMP flood	%100
		Normal	%0
		TCP SYN flood	%0
		UDP flood	%0
SVC	Ortalama Eğitim Süresi: 9.91 Saniye Ortalama Test Süresi: 0.44 Saniye	HTTP flood	%98
		ICMP flood	%99
		Normal	%98
		TCP SYN flood	%0
		UDP flood	%0

4.4.5 Hazırlanan verisetleri üzerinde uygulanan ön işlemler ile makine öğrenmesi algoritmalarının NSL-KDD veriseti üzerinde uygulanması

Çalışmaya ek olarak, hazırlanan eğitim veriseti ve test verisetine uygulanan adımlar, NSL-KDD eğitim verisetine ve test verisetine uygulanmıştır, ardından sonuçlar incelenmiştir. Hazırlanan verisetilerinde uygulanan adımlar için, NSL-KDD veriseti içerisinde bulunan 'smurf' (ICMP flood), 'land' (SYN flood), 'neptune' (SYN flood), 'apache2' (HTTP flood), 'udpstorm' (UDP flood) DoS saldırı türleri ve normal trafik verileri kullanılarak yeni test verisetleri oluşturulmuştur.

Uygulanan adımlardan elde edilen veriler analiz edildiğinde, verisetlerinde ön işlem uygulanmadığında, hazırlanan verisetlerinde elde edilen tespit oranlarına göre daha iyi sonuçlar elde edilmiştir (Çizelge 4.7). Sonuçlar KNN algoritması için % 78.6, RF algoritması için % 79.6, MLP algoritması için % 79.9 ve SVC algoritması için % 31.8 dir. Çalışma sürelerini incelediğinde KNN algortiması uygulandığında, eğitim veriseti üzerinde çalışma süresi ortalama 12.720 saniye iken test verisetinde çalışma süresi 4.826 saniyedir.

RF algoritması uygulandığında eğitim veriseti üzerinde hesaplanan çalışma süresi, KNN ile alınan sonuca yakındır. Eğitim veriseti üzerinde hesaplanan ortalama çalışma süresi 13.481 saniye iken, test veriseti üzerinde hesaplanan süre 0.502 saniyedir. Bu süre KNN için hesaplanan test veriseti üzerinde çalışma süresinden kısadır. Eğitim ve test verisetleri üzerinde MLP algoritması çalıştırıldığında ortalaman hesaplanan çalışma süreleri sırası ile 33.744 saniye ve 0.266 saniyedir.

MLP için hesaplanan eğitim veriseti üzerinde çalışma süresi KNN ve RF için hesaplanan sürelere göre uzundur. Son olarak SVC için hesaplanan ortalama çalışma sürelerine bakılırsa, eğitim veriseti için 135.913 saniye ve test veriseti için 20.121 saniye hesaplanmıştır. Hesaplanan süreler karşılaştırıldığında, ön işlem olmadan uygulanan SVC algoritması diğer algoritmalara göre daha uzun sürede saldırı tespiti yapabilmiştir. Ön işlem uygulanmayan verisetinde çapraz doğrulama yapıldığında, yukarıdaki sonuçlara benzer şekilde hazırlanan verisetlerinde alınan tespit oranlarına göre yüksek değerler elde edilmiştir. MLP ve RF için %79, KNN için %78 ve en

düşük olarak SVC için %31 değerleri elde edilmiştir. Modelin başarısı incelendiğinde (EK 6), kesinlik ve duyarlılık oranları, hazırlanan verisetleri üzerinde hesaplanan değerlerin üstündedir. SVC hariç uygulanan diğer makine öğrenmesi algoritmaları sonunda %79 F1-skor değeri elde edilmiştir. SVC için ise %32 olarak kayıt edilmiştir.

Çizelge 4.7 NSL-KDD veriseti üzerinde ön işlem yapılmadan alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 33.744 Saniye Ortalama Test Süresi : 0.266 Saniye	HTTP flood	%99.8
		ICMP flood	%99.9
		Normal	%99.7
		TCP SYN flood	%99.8
		UDP flood	%0
RF	Ortalama Eğitim Süresi: 13.481 Saniye Ortalama Test Süresi: 0.507 Saniye	HTTP flood	%99.9
		ICMP flood	%99.7
		Normal	%98.9
		TCP SYN flood	%98.7
		UDP flood	%0
KNN	Ortalama Eğitim Süresi: 12.720 Saniye Ortalama Test Süresi: 4.826 Saniye	HTTP flood	%99.2
		ICMP flood	%95.7
		Normal	%99.7
		TCP SYN flood	%99.9
		UDP flood	%0
SVC	Ortalama Eğitim Süresi: 135.913 Saniye Ortalama Test Süresi: 20.121 Saniye	HTTP flood	%99.7
		ICMP flood	%59.7
		Normal	%0
		TCP SYN flood	%0
		UDP flood	%0

Ölçeklendirme ön işleme sonunda uygulanan KNN, RF ve MLP makine öğrenme algoritmalarının sonuçları % 73.4, % 24.4, % 78.6'dır (Çizelge 4.8). Bu oranlar ilk sonuçlara göre düşük olsa da SVC için elde edilen tespit oranı ön işlem olmadan alınan sonuca göre artmış ve % 62,1 olarak hesaplanmıştır. DDoS saldırı türleri tespit oranları incelendiğinde, RF algoritması için ölçeklendirme işlemi sonrasında HTTP flood tespit oranı % 99.7'den % 21.30'a, ICMP flood için tespit oranı % 98.9'dan % 0.81'e düşmüş ve SYN flood için tespit oranı % 99,8'den % 21.30'a düşmüştür. Çalışma süreleri incelendiğinde, ölçeklendirme sonunda uygulanan KNN algoritması eğitim veriseti üzerinde ortalama 77.388 saniyede saldırı tespiti yapabilmişken, test veriseti üzerinde 16.738 saniye çalışmıştır. ön işlem yapılmadan alınan eğitim ve test süreleri ile karşılaştırma yapıldığında hem eğitim hem de test veriseti üzerinde saldırı tespit sürelerinin uzadığı gözlemlenmiştir. RF için hesaplanan çalışma süreleri incelendiğinde, KNN algoritması için alınan sonucun tam tersi bir durum gözlenmiştir. Eğitim veriseti için ortalama çalışma süresi 9.440 saniyeye ve test veriseti için ortalama çalışma süresi 0.38 saniyeye düşmüştür. Ölçeklendirme işlemi sonunda eğitim ve test veriseti üzerinde MLP algoritması çalışma sürelerini incelendiğinde, ölçeklendirme ön işleminin MLP çalışma süresini çok etkilemediğini gözlenmiştir. Eğitim veriseti için hesaplanan süre 33.83 saniye iken, test veriseti için hesaplanan süre 0.09 saniyedir. Ölçeklendirme sonunda, verisetleri üzerinde uygulunan SVC algoritmasının, verisetleri üzerinde ön işlem olmadan uygulanan SVC algoritmasına göre daha hızlı sonuç verdiği gözlemlenmiştir. Eğitim veriseti üzerinde hesaplanan ortalama çalışma süresi 1.768 saniye iken test veriseti üzerinde hesaplanan ortalama çalışma süresi 0.313 saniyedir.

Yapılan çapraz doğrulama işlemi sonunda, görülen tespit oranları, yukarıdakine benzer şekilde SVC hariç diğer algoritmalar için düşmüştür. SVC için hesaplanan tespit oranı %64'e yükselmiştir. MLP için %78, RF için %41 ve KNN için %73 olarak hesaplanmıştır.

Saldırı olarak etiketlenen verilerin, gerçekte ne kadarının saldırı olduğuna yani kesinlik değerine bakarsak RF için alınan değer ön işlemsiz adıma göre düşmüştür, fakat SVC için bu değer yükselmiştir (EK 7). MLP ve KNN için değişiklik görülmemiştir. Saldırı olanların ne kadarının tespit edildiği duyarlılık oranına baktığımızda, yine

aynı şekilde RF için düşüş meydana gelmişken, SVC için alınan değer yükselmiştir. SVC algoritması sonunda alınan F1-skor yükselmiş olsa da en iyi model MLP (%79) olarak kayıt edilmiştir.

Çizelge 4.8 NSL-KDD veriseti üzerinde ölçeklendirme ön işlemi yapıldıktan alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 33.83 Saniye Ortalama Test Süresi : 0.09 Saniye	HTTP flood	%99.8
		ICMP flood	%95.1
		Normal	%98.7
		TCP SYN flood	%99.8
		UDP flood	%0
RF	Ortalama Eğitim Süresi: 9.440 Saniye Ortalama Test Süresi: 0.381 Saniye	HTTP flood	%99.9
		ICMP flood	%21.3
		Normal	%0.81
		TCP SYN flood	%21.8
		UDP flood	%0
KNN	Ortalama Eğitim Süresi: 77.388 Saniye Ortalama Test Süresi: 4.826 Saniye	HTTP flood	%99.3
		ICMP flood	%77.7
		Normal	%90.7
		TCP SYN flood	%99.9
		UDP flood	%0
SVC	Ortalama Eğitim Süresi: 1.768 Saniye Ortalama Test Süresi: 0.313 Saniye	HTTP flood	%99.7
		ICMP flood	%30.8
		Normal	%80
		TCP SYN flood	%99.9
		UDP flood	%0

Verilere özellik azaltma ön işleme uygulandığında verisetleri içerisinden 'flag' özelliği kaldırılmıştır. Sonrasında uygulunan KNN algoritması için % 88.8, RF ile MLP algoritmaları için % 89.9 ve son olarak SVC algoritması için % 41.88 tespit oranları elde edilmiştir (Çizelge 4.9). SVC için elde edilen sonuç, ölçeklemenin sonunda elde edilen sonuçtan daha düşük olmasına rağmen, ön işleme olmadan elde edilen sonuçtan daha yüksektir. Verisetleri üzerinde özelllik azaltma işleminin makine öğrenmesi algoritmalarının çalışma süresine etkisi incelendiğinde, KNN algoritmasını eğitim veriseti üzerinde çalışma süresi, ölçeklendirme sonunda hesaplanan süreye göre düşmüştür. Ortalama çalışma süresi 13.105 saniye olarak hesaplanmıştır. Test verisetinde çalışma süresi 2.464 saniyedir. RF algoritması için hesaplanan süreler, ölçeklendirme sonunda alınan sürelerle hemen hemen aynıdır. Ortalama süre eğitim verisetinde 9.744 saniye hesaplanmışken, test verisetinde 0.461 saniyedir. RF algoritması sonunda alınan sonuçlar gibi, özellik azaltma ve ölçeklendirme sonunda MLP algoritması ile hesaplanan süreler birbirine yakın değerdedir. Ortalama süre eğitim verisetinde 24.179 saniye hesaplanmışken, test verisetinde 0.329 saniyedir. SVC algoritması için hesaplanan süreler, ölçeklendirme sonunda elde edilen sürelere göre uzundur ve ön işlem yapılmadan alınan sonuçlara yakındır. Ortalama süre eğitim verisetinde 143.29 saniye hesaplanmışken, test verisetinde 12.836 saniyedir. Yapılan çapraz doğrulama ile elde edilen tespit oranları, SVC hariç diğer adımlar sonrasında elde edilen en yüksek değerlerdir. MLP için %91, RF için %90, KNN için %89 ve SVC için %31 değerleri elde edilmiştir. Kesinlik ve duyarlılık değerleri harmonik ortalamasına bakıldığında (EK 8), MLP için %92, RF için %90, KNN için %89 değerleri elde edilmişken, SVC için ön işlemsiz adımla aynı F1-skor değeri elde edilmiştir (%31).

Çizelge 4.9 NSL-KDD veriseti üzerinde özellik azaltma ön işlemi yapıldıktan alınan sonuçlar

Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 24.179 Saniye Ortalama Test Süresi : 0.09 Saniye	HTTP flood	%99.5
		ICMP flood	%99.1
		Normal	%98.9
		TCP SYN flood	%99.8
		UDP flood	%51.1
RF	Ortalama Eğitim Süresi: 9.744 Saniye Ortalama Test Süresi: 0.461 Saniye	HTTP flood	%99.9
		ICMP flood	%99.8
		Normal	%99.9
		TCP SYN flood	%99.8
		UDP flood	%50.8
KNN	Ortalama Eğitim Süresi: 13.105 Saniye Ortalama Test Süresi: 2.464 Saniye	HTTP flood	%99.2
		ICMP flood	%95.7
		Normal	%99.7
		TCP SYN flood	%99.3
		UDP flood	%52.1
SVC	Ortalama Eğitim Süresi: 143.29 Saniye Ortalama Test Süresi: 12.836 Saniye	HTTP flood	%99.7
		ICMP flood	%59.8
		Normal	%0
		TCP SYN flood	%0
		UDP flood	%0

Son olarak, verisetlerine her iki ön işlem uygulandıktan sonra, makine öğrenimi yöntemlerinin sonuçları incelenmiştir. KNN algoritması için % 84.7, RF algoritması için % 74.4, MLP algoritması için % 89.6 ve SVC için % 81.5 tespit oranları alınmıştır (Çizelge 4.10). KNN, RF ve MLP algoritmaları için elde edilen sonuçlar, özellik azaltma ön işlemesinden sonra elde edilen sonuçlara kıyasla düşük ve ölçekleme ön işleminden sonra elde edilen sonuçlardan daha yüksek çıkmıştır. SVC kullanılarak elde edilen HTTP flood, ICMP flood, SYN flood tespit oranları diğer algoritmalara kıyasla düşük değerdedir. SVC ile verisetlerine hiçbir ön işleme uygulanmadığında veya özellik azaltma ön işleme uygulandığında, HTTP flood için % 59.7, ICMP flood için % 0 ve SYN flood için % 0 tespit oranları alınmıştır. Daha sonra SVC ile verisetlerine ölçeklendirme uygulandığında HTTP flood için % 30.8, ICMP flood için % 80 ve SYN flood için % 99.9 tespit oranları alınmıştır. SVC için en iyi sonuç, her iki ön işleme uygulanarak elde edilmiştir. Ölçekleme ön işlemi, SVC için yüksek tespit oranları elde edilmesini sağlanmıştır, ancak diğer algoritmalar için başarısız olmuştur. Her iki ön işlem sonunda uygulanan makine öğrenmesi algoritmalarının çalışma sürelerini incelersek, KNN algoritmasının eğitim veriseti üzerinde saldırı tespit süresi uzamış ve 66.62 saniye olarak hesaplanmıştır.Test verisetinde tespit süresi 2.464 saniye olarak kaydedilmiştir. Her iki ön işlemin uygulandığı verisetleri üzerinde RF algoritmasının, ayrı ayrı uygulanan ölçeklendirme ve özellik azaltma işlemleri sonunda uygulanan RF algoritmasına göre daha yavaş saldırı tespiti yaptığı gözlenmiştir. Eğitim verisetinde ortalama çalışma süresi 14.362 saniye ve test verisetinde ortalama çalışma süresi 0.51 saniyedir. MLP algoritması sonunda hesaplanan, eğitim veriseti üzerinde ortalama çalışma süresi 31.692 saniyeye yükselmiştir. Test verisetinde hesaplana çalışma süresi 0.194 saniye olarak kayıt edilmiştir. Son olarak Verisetleri üzerinde her iki ön işlemin yapıldıktan sonra uygulanan SVC algoritmasının ortalama çalışma sürelerini incelersek, SVC için en kısa çalışma süreleri kaydedilmiştir. Eğitim veriseti üzerinde çalışma süresi 1.305 saniye iken, test veriseti üzerinde çalışma süresi 0.233 saniyedir.

Her iki ön işlem sonunda 10'lu çapraz doğrulama işlemi sonunda, MLP, RF ve KNN için alınan değerler, özellik azaltma sonunda alınan değerlere göre düşük olsa da yukarıdaki sonuca benzer şekilde SVC için en iyi tespit oranı hesaplanmıştır (%81).

Modelin başarısını incelersek (EK 9), tespit oranları ile aynı şekilde MLP, RF ve KNN için alınan değerler düşmüştür. Fakat SVC için alınan değer artmıştır.

Çizelge 4.10 NSL-KDD veriseti üzerinde özellik azaltma ve ölçeklendirme ön işlemleri yapıldıktan sonra alınan sonuçlar

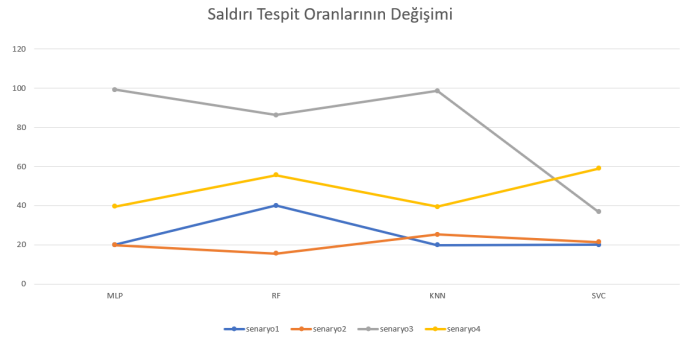
Algoritmalar	Çalışma Süreleri	Saldırı Çeşitleri	Sonuçlar
MLP	Ortalama Eğitim Süresi : 32.692 Saniye Ortalama Test Süresi : 0.195 Saniye	HTTP flood	%99.9
		ICMP flood	%98.1
		Normal	%99.3
		TCP SYN flood	%99.6
		UDP flood	%51.1
RF	Ortalama Eğitim Süresi: 14.362 Saniye Ortalama Test Süresi: 0.51 Saniye	HTTP flood	%99.9
		ICMP flood	%38.12
		Normal	%84.4
		TCP SYN flood	%99.8
		UDP flood	%51.3
KNN	Ortalama Eğitim Süresi: 66.62 Saniye Ortalama Test Süresi: 15.62 Saniye	HTTP flood	%99.9
		ICMP flood	%83.4
		Normal	%90.7
		TCP SYN flood	%99.9
		UDP flood	%50.7
SVC	Ortalama Eğitim Süresi: 1.305 Saniye Ortalama Test Süresi: 0.233 Saniye	HTTP flood	%99.8
		ICMP flood	%68.8
		Normal	%90.7
		TCP SYN flood	%99.4
		UDP flood	%53.1

5. ELDE EDİLEN SONUÇLARIN DEĞERLENDİRİLMESİ

Yapılan çalışmada dört farklı senaryo, dört farklı makine öğrenmesi algoritması üzerinde uygulanmış ve farklı DDoS saldırı çeşitlerinin tespiti için alınan tespit oranları ve süreleri analiz edilmiştir. İlk olarak tespit oranları incelendiğinde (Şekil 5.1); makine öğrenmesi algoritmalarının, özellik azaltma ve ölçeklendirme işlemleri yapılmadan saldırı tespit oranlarının çok düşük olduğu ve saldırı verilerinin normal trafik olarak etiketlendiği gözlemlenmiştir. Sadece RF ile UDP flood ve SVC ile ICMP flood için %100 tespit yapılabilmiştir. En yüksek tespit oranı bütün saldırı çeşitlerinin ortalaması alındığında %40 ile RF kullanılarak elde edilmiştir. İlk senaryoda makine öğrenmesi algoritmalarının, verisetleri üzerinde ön işlem yapılmadan uygulandığında başarısız olduğu gözlemlenmiştir. Daha sonra verileri 0 ve 1 sayıları arasında ölçeklendirme işlemi uyguladığımızda, MLP, KNN, RF algoritmalarının sonucunda ulaşılan tespit değerleri düşmüştür. Uygulanan makine öğrenmesi algortimaları sonunda hiçbir saldırı %100 tespit edilememiştir. Sadece normal trafik verileri, MLP sonunda %92 ve SVC sonunda %84 oranında tespit edilemiştir. Diğer adımda veriler üzerinde ölçeklendirme yerine, özelliklerin saldırı etiketleriyle olan ilişkilerinin hesaplandığı korelasyon işlemi sonucunda özellik azaltma ön işlemi yapılmıştır. Verisetinden 'Time', 'Rel Start', 'Source', 'Destination', 'Delta Time' özellikleri çıkarıldıktan sonra uygulanan makine öğrenmesi algoritmalarının tespit oranları ve çalışma sürelerini incelediğimizde; SVC için alınan tespit değerleri düşmüştür. Fakat alınan en iyi saldırı tespit oranları KNN (%98.6) ve MLP (%99.2) algoritmaları ile elde edilmiştir. MLP için en düşük saldırı tespit oranı %98'dir. KNN için ise %97'dir.

Son olarak ölçeklendirme ve özellik azaltma işleminin aynı anda uygulanması, sadece SVC için yüksek tespit oranları alınmasını sağlamıştır. TCP SYN flood ve UDP flood saldırılarının tespiti yapılamamıştır. HTTP flood ve normal trafik verileri incelendiğinde alınan tespit oranı %98, ICMP flood için %99 tespit oranı elde edilmiştir. MLP algoritması sonunda sadece ICMP flood %100 tespit edilebilmiştir. HTTP flood %47 ve normal trafik verisi %33 tespit edilemiştir. Eğitim veriseti üzerinde çalışma süresi

42.92 saniye, test veriseti üzerinde çalışma süresi 0.006 saniye kaydedilmiştir. RF incelendiğinde MLP den daha iyi sonuçlar gözlemlenmiştir. ICMP flood %100 tespit edilmişken, HTTP flood %79 ve normal trafik verileri %99 oranında tespit edilmiştir.



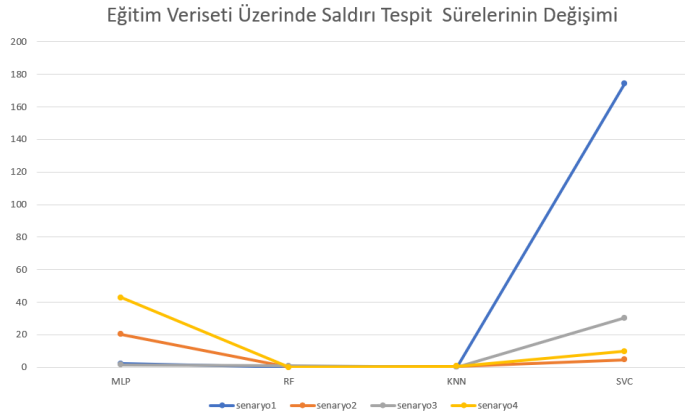
Şekil 5.1 Saldırı tespit oranlarının değişimi

Uygulanan senaryolar sonunda, makine öğrenmesi algoritmalarının eğitim ve test verisetleri üzerindeki çalışma sürelerini incelediğimizde (Şekil 5.2) (Şekil 5.3); uygulanan ilk senaryoda; RF için eğitim ve test verisetleri üzerinde, ortalama çalışma süreleri kontrol edildiğinde, sırasıyla 0.114 saniye ve 0.002 saniyedir. KNN için, sırası ile 0.166 saniye ve 0.001 saniyedir. RF ve KNN algoritmalarının daha hızlı çalıştığı gözlemlenmiştir, fakat KNN algoritması tespit değeri çok düşüktür.

İkinci senaryo için, eğitim ve test veriseti üzerinde çalışma sürelerinde yüksek bir artış gözlemlenmiştir. MLP, eğitim veriseti üzerinde çalışma süresi 20.37 saniyeye ve test veriseti üzerinde çalışma süresi 0.005 saniyeye yükselmiştir. Uygulanan ölçeklendirme işlemi SVC hariç diğer makine öğrenmesi algoritmaları için çalışmanın sürelerinin uzamasına ve ortalama tespit oranlarının düşmesine neden olmuştur.

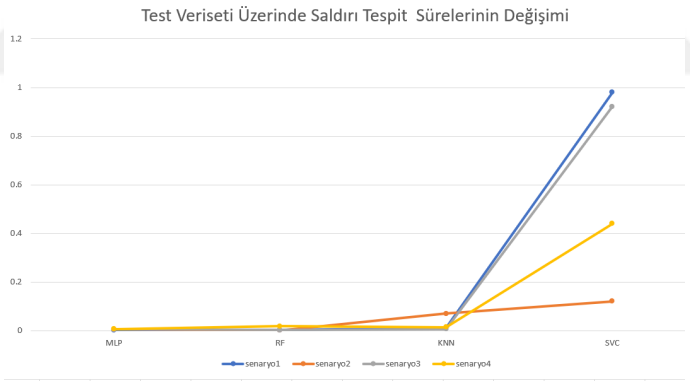
Üçüncü senaryoda toplam çalışma zamanlarında düşüş görülmüştür. MLP algoritması için eğitim veriseti üzerinde için harcanan ortalama süre 1.51 saniye iken, test veriseti harcanan süre 0.005 saniyedir. KNN algoritması için eğitim veriseti üzerinde tüm saldırı çeşitlerinin ve normal trafik verisinin tespiti için harcanan ortalama süre 0.141 saniye iken, test veriseti harcanan süre 0.006 saniyedir.

Dördüncü senaryo sonunda ölçeklendirme ön işlemi sonunda alınan çalışma süresinin arttığı gözlemlenmiştir. MLP, RF ve KNN için ise ölçeklendirme işlemi, alınan tespit oranlarında düşüşe ve çalışma sürelerinde artışa neden olmuştur. Çalışma sü-



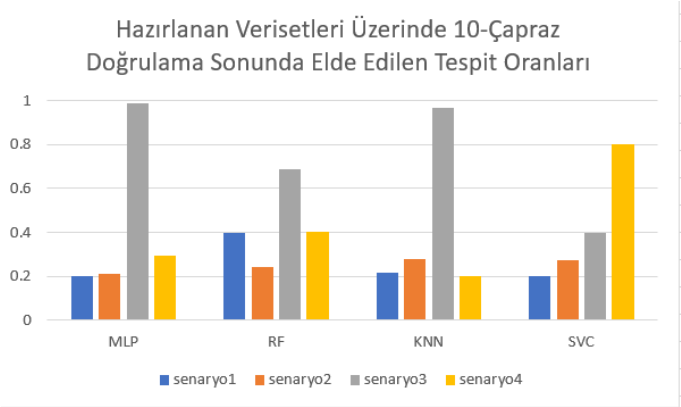
Şekil 5.2 Eğitim veriseti üzerinde saldırı tespit sürelerinin değişimi

relerine bakıldığında eğitim veriseti üzerinde çalışma süresi 0.076 saniye, test veriseti üzerinde çalışma süresi 0.014 saniye olarak gözlemlenmiştir. Diğer algoritma KNN sonuçları incelendiğinde, normal trafik verisi, TCP SYN flood ve UDP flood saldırılarının tespiti yapılamamıştır. Çalışma süreleri incelendiğinde eğitim veriseti için harcanan süre 0.56 saniye ve test veriseti için harcanan süre 0.0014 saniyedir.



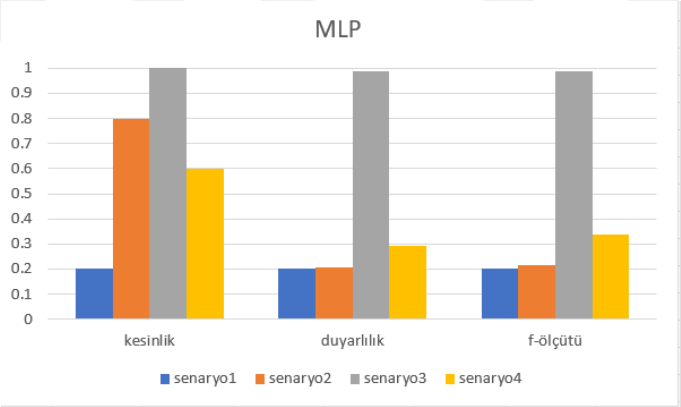
Şekil 5.3 Test veriseti üzerinde saldırı tespit sürelerinin değişimi

Hazırlanan verisetleri üzerinde, 10'lu çapraz doğrulama işlemi uygulanarak eğitim ve test verisetleri üzerinde uygulanan ön işlemlerin, makine öğrenmesi algoritmaları üzerindeki tespit oranları değişimine etkisini incelersek, çapraz doğrulama olmadan alınan sonuçlara benzer şekilde, en iyi sonuçlar özellik azaltma ile edilmişken, bu durum SVC için her iki ön işlemin uygulanması sonunda elde edilmiştir (Şekil 5.4). Hazırlanan verisetleri üzerinde uygulanan senaryoların, farklı makine öğrenmesi algoritmaları üzerindeki başarılarını incelersek(Şekil 5.5); MLP için, özellik azaltma



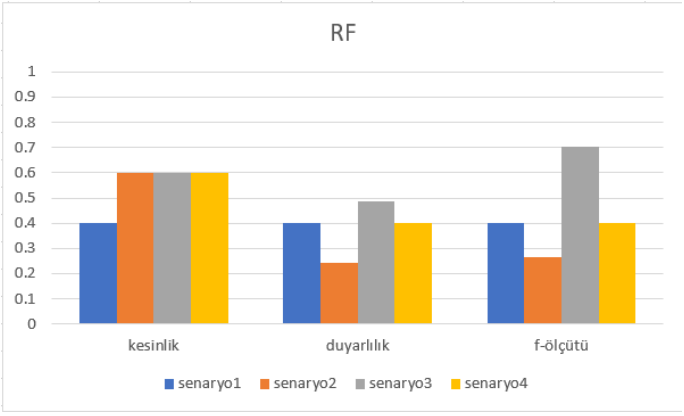
Şekil 5.4 Hazırlanan verisetleri üzerinde 10-çapraz doğrulama sonunda elde edilen tespit oranları

ön işlemi sonunda en başarılı model elde edilmiştir. Ölçeklendirme ön işlemi, saldırı olarak etiketlediğimiz verilerin doğru olması oranı olan kesinlik değerinin artmasına neden olsa da, saldırı verilerinin tespit oranı olan duyarlılık için aynı etkiyi gösterememiştir. MLP için en başarısız modeller, ön işlemsiz senaryo ile ölçeklendirme senaryolarına aittir.



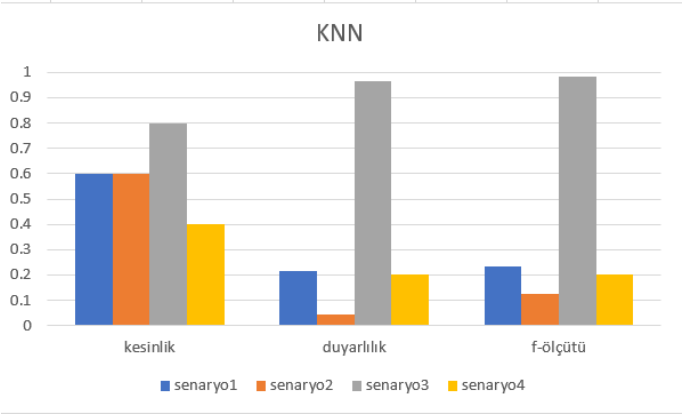
Şekil 5.5 Hazırlanan verisetleri üzerinde MLP sonunda elde edilen model başarı oranları

RF sonunda elde edilen model başarı değerlerine bakıldığında, MLP algoritmasına benzer şekilde en iyi değerler özellik azaltma ön işlemi sonunda elde edilmiştir. RF için en belirleyici değer duyarlılık olarak görülmektedir(Şekil 5.6). Özellikle ölçeklendirme ön işlemi sonunda, saldırıların tespit edilememesi alınan F1-skor değerinde düşüşe neden olmuştur.



Şekil 5.6 Hazırlanan verisetleri üzerinde RF sonunda elde edilen model başarı oranları

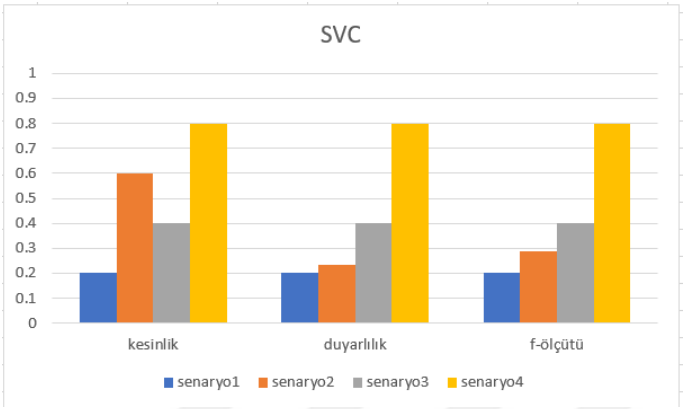
KNN ile elde edilek kesinlik, duyarlılık ve F1-skor değerlerine bakıldığında (Şekil 5.7); diğer yöntemlere göre büyük bir fark ile özellik azaltmanın daha başarılı olduğu görülmektedir. Özellikle ölçeklendirme ön işlemi sonunda alınan duyarlılık değerleri düşüktür ve saldırı verileri normal olarak etiketlenmiştir. Her iki ön işlem sonunda, kesinlik değeri daha düşük olsa da, alınan duyarlılık değeri nedeni ile uygulanan ölçeklendirme sonunda daha düşük F1-skor değeri elde edilmiştir.



Şekil 5.7 Hazırlanan verisetleri üzerinde KNN sonunda elde edilen model başarı oranları

Son olarak hazırlanan verisetleri üzerinde, senaryolar sonunda SVC modellerini incelersek (Şekil 5.8); en iyi kesinlik, duyarlılık ve F1-skor değerlerinin her iki ön işlem sonunda elde edildiği görülmektedir. En kötü değerler ise ön işlem olmadan elde edilmiştir. Ölçeklendirme sonunda elde edilen kesinlik oranı, özellik azaltma sonunda

elde edilen değerden yükse olsa da, duyarlılık değerinin özellik azaltma sonunda elde edilen duyarlılık değerinden düşük olması neden ile üçüncü senaryo daha başarılı görülmektedir.



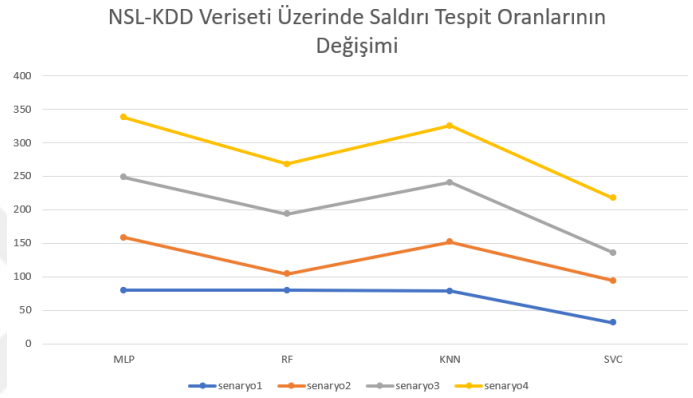
Şekil 5.8 Hazırlanan verisetleri üzerinde SVC sonunda elde edilen model başarı oranları

NSL-KDD veriseti üzerinde aynı adımlar uygulandığında alınan tespit sonuçlarını incelersek (Şekil 5.9), ön işlem yapılmadan uygulanan makine öğrenmesi algoritmalarının sonuçları hazırlanan verisetlerinde ön işlem yapılmadan alınan sonuçlardan daha yüksek hesaplanmıştır. MLP algoritması sonunda %79.9, RF algoritması sonunda %79.6, KNN algoritması sonunda %78.6 ve son olara SCV algoritması sonunda %31.8 tespit oranı elde edilmiştir. Hazırlanan verisetlerinde en yüksek tespit oranı RF ile %40 iken NSL-KDD verisetinde, MLP sonunda alınan %79.9 değeridir. Verisetlerine uygulanan ölçeklendirme ön işleminden sonra, KNN, RF ve MLP algoritmaları sonuçlarının ilk sonuçlara göre düştüğü gözlemlenirken, SVC algoritması için arttığı gözlemlenmiştir. Sırası ile alınan tespit oranları KNN için % 73.4, RF için %24.4 ve MLP için %78.6'dır. Tespit oranlarındaki düşüş en fazla RF için yaşanmıştır. Fakat SVC ile alınan sonuç neredeyse ön işlem olmadan alınan sonucun iki katıdır ve %62.1 olarak kayıt edilmiştir.

Verilere özellik azaltma ön işlemi uygulandığında, ön işlem uygulanmadığında ve ölçeklendirme ön işlemi uygulandığında alınan sonuçlara kıyasla daha yüksek tespit oranları alınmıştır. SVC algoritması için elde edilen sonuç, ölçeklendirmeden daha düşük ve ön işlem olmadan alınan sonuçtan yüksektir. KNN algoritması sonunda

% 88.8, RF ile MLP algoritmaları sonunda % 89.9 ve SVC algoritması sonunda % 41.88 tespit oranları hesaplanmıştır.

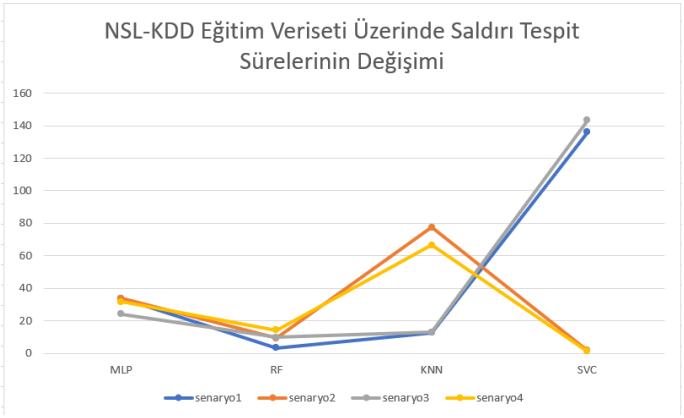
Son olarak, verisetlerine her iki ön işlem uygulandıktan sonra, MLP, KNN ve RF algoritmaları için ölçeklendirme sonucu ile özellik azaltma ön işlemlerinin sonuçları hesaplanmıştır. SVC algoritması için daha iyi sonuçlar hesaplanmıştır. Tespit oranlarına bakılacak olursa KNN algoritması için %84.7, RF algoritması için % 74.4, MLP algoritması için % 89.6 ve SVC için % 81.5 tespit oranları alınmıştır.



Şekil 5.9 NSL-KDD veriseti üzerinde saldırı tespit oranlarının değişimi

Uygulanan senaryolar sonunda, makine öğrenmesi algoritmalarının NSL-KDD eğitim ve test verisetleri üzerindeki çalışma sürelerini incelediğimizde (Şekil 5.10) (Şekil 5.11); ilk senaryo için tüm algoritmaların çalışma süreleri hazırlanan veriseti üzerinde hesaplanan çalışma sürelerine göre yüksek olduğu gözlemlenmiştir. Örneğin; hazırlanan eğitim verisetleri üzerinde hesaplanan süre KNN için 2.285 saniye iken bu değer NSL-KDD eğitim veriseti içn 12.720 saniyedir. Aynı şekilde KNN için hazırlanan test verisetinde hesaplanan çalışma süresi 0.003 saniye iken bu değer NSL-KDD test veriseti içn 4.826 saniyedir.

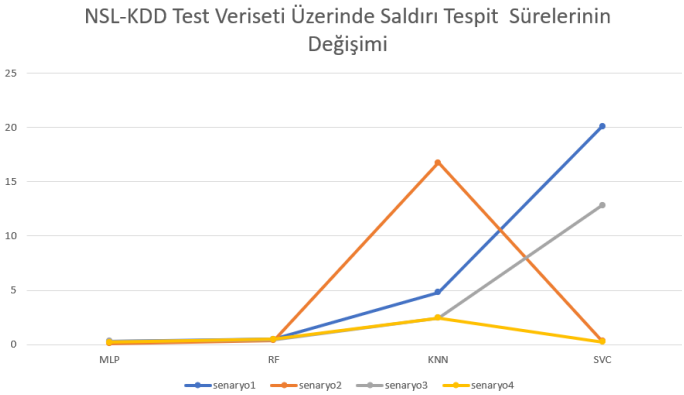
İkinci senaryo sonunda; ilk senaryo ile karşılaştırıldığında, RF için çalışma süresinde büyük bir fark olmasa da, KNN için eğitim veriseti üzerinde çalışma süresi 12.72 saniyeden 77.388 saniyeye çıkmıştır. Aynı şekilde test veriseti üzerinde çalışma süresi 4.82 saniyeden 16.73 saniyeye çıkmıştır. Fakat SVC için çalışma süreleri ilk senaryoya göre azalmıştır. Eğitim veriseti üzerindeki çalışma süresi 135.913 saniyeden, 1.768 saniyeye düşmüştür.



Şekil 5.10 NSL-KDD eğitim veriseti üzerinde saldırı tespit sürelerinin değişimi

Üçüncü senaryoda, MLP algoritması için en iyi çalışma süreleri elde edilmişken, SVC için en kötü sonuçlar alınmıştır. MLP için eğitim veriseti üzerindeki çalışma süresi 24.175 saniye iken, SVC için 143.29 saniye olarak hesaplanmıştır. KNN için hesaplanan süre ise ikinci senaryoya göre düşük değerde ve ilk senaryoya çok yakın değerdedir. Eğitim veriseti üzerinde harcanan süre 13.105 saniyedir.

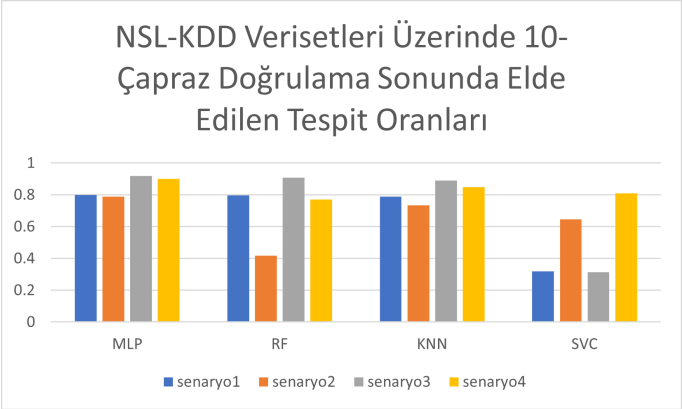
Son senaryoda, SVC için en iyi çalışma süreleri hesaplanmıştır. Eğitim veriseti üzerinde hesaplanan süre 1.305 saniye ve test eriseti üzerinde hesaplanan süre 0.233 saniyedir. Diğer bütün algoritmalar için hesaplanan çalışma sürelerinde artış görülmüştür. RF için en kötü süreler hesaplanmıştır.



Şekil 5.11 NSL-KDD test veriseti üzerinde saldırı tespit sürelerinin değişimi

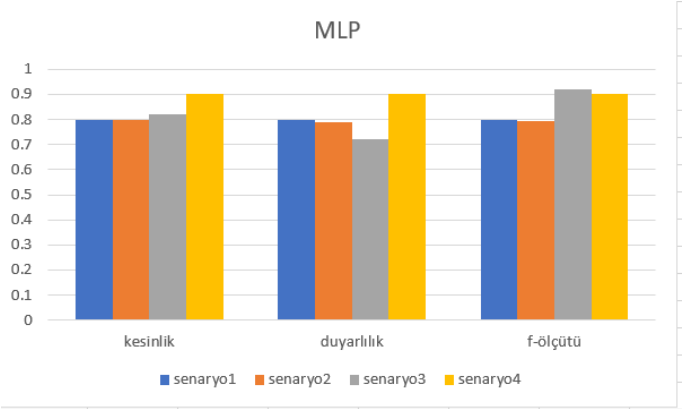
NSL-KDD verisetleri üzerinde, 10'lu çapraz doğrulama işlemi uygulanarak eğitim ve test verisetleri üzerinde uygulanan ön işlemlerin, makine öğrenmesi algoritmaları üzerindeki etkisini incelersek, çapraz doğrulama olmadan alınan sonuçlara benzer

şekilde, en iyi sonuçlar özellik azaltma ile edilmişken, bu durum SVC için her iki ön işlemin uygulanması sonunda elde edilmiştir (Şekil 5.12).



Şekil 5.12 NSL-KDD verisetleri üzerinde 10-çapraz doğrulama sonunda elde edilen tespit oranları

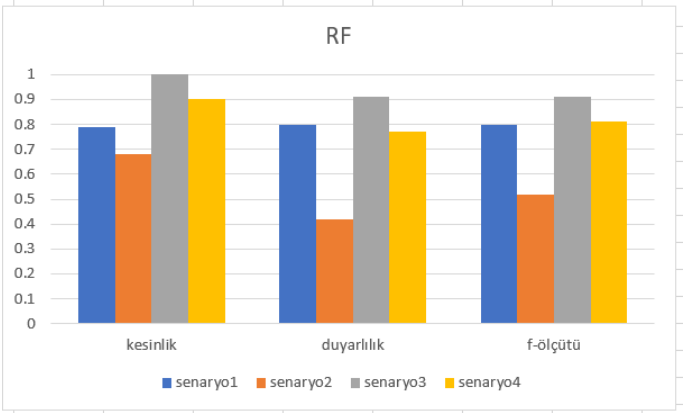
NSL-KDD veriseti üzerinde uygulanan senaryoların, farklı makine öğrenmesi algoritmaları üzerindeki başarılarını incelersek (Şekil 5.13); MLP için, hazırlanan verisetleri üzerinde alınan kesinlik ve duyarlılık değerlerinde artış gerçekleşmiştir. Bütün senaryolar karşılaştırıldığında ön işlemsiz adım sonunda alınan değerler ile ölçeklendirme sonunda alınan değerlerin hemen hemen aynı olduğu görülmüştür. Özellik azaltma ön işlemi sonunda, kesinlik ve duyarlılık değerleri, her iki ön işlem uygulandığında elde edilen değerlere göre düşük olarak hesaplanmıştır.



Şekil 5.13 NSL-KDD veriseti üzerinde MLP sonunda elde edilen model başarı oranları

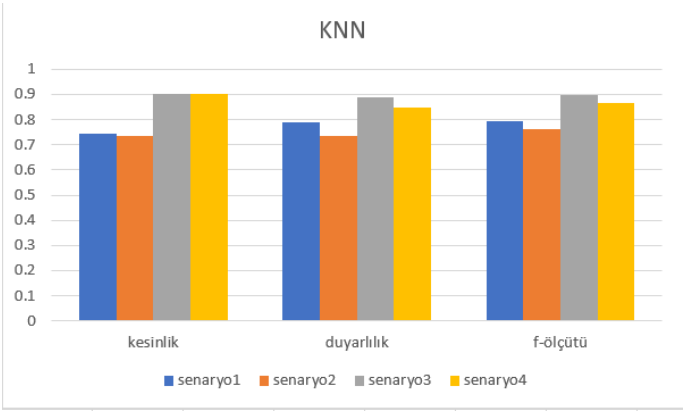
RF sonunda elde edilen sonuçlara bakarsak (Şekil 5.14); özellik azaltma ön işlemi

sonunda elde edilen kesinlik, duyarlılık ve F1-skor en yüksek değerlerdir. En kötü sonuçlar ise ölçeklendirme ön işlemi sonunda elde edilmiştir.



Şekil 5.14 NSL-KDD veriseti üzerinde RF sonunda elde edilen model başarı oranları

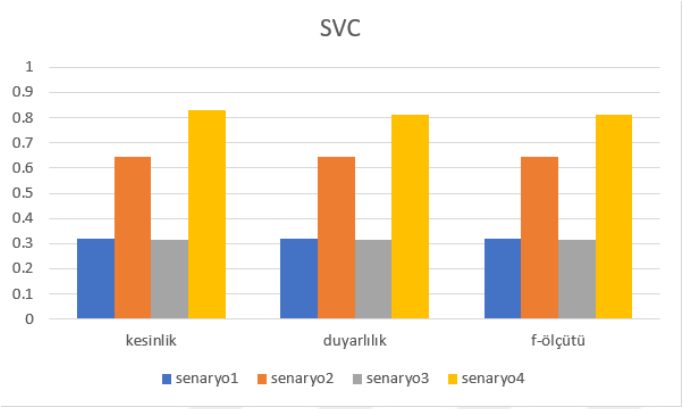
KNN ile uygulanan senaryolar sonunda, en iyi kesinlik sonuçları özellik azaltma ön işlemi ve her iki ön işlemin uygulanması sonucunda elde edilmiştir (Şekil 5.15). Diğer senaryolar içinde sonuçlar yakındır. Fakat duyarlılık değeri, özellik azaltma ön işlemi sonunda, her iki ön işlem işlem için kayıt edilen değerden yüksek hesaplanmıştır. Aynı şekilde ölçeklendirme ön işlemi sonunda alınan duyarlılık değeri, ön işlemsiz adıma göre yüksektir.



Şekil 5.15 NSL-KDD veriseti üzerinde KNN sonunda elde edilen model başarı oranları

Son olarak; SVC için uygulanan senaryolar sonunda başarı oranlarını incelediğimizde (Şekil 5.16), diğer algoritmalardan farklı olarak, özellik azaltma ön işlemi, alınan

kesinlik, duyarlılık değerlerinin düşmesine sebep olurken, ölçeklendirme ön işlemi daha yüksek değerler alınmasını sağlamıştır. En iyi değerler ise her iki ön işlem uygulanarak elde edilmiştir.



Şekil 5.16 NSL-KDD veriseti üzerinde SVC sonunda elde edilen model başarı oranları

6. SONUÇLAR

Günümüze bakıldğında, gelişen teknoloji ile birlikte İnternet üzerinden yapılan işlemler ve verilen hizmetler hayatımızın önemli parçası haline gelmiştir. Kullanılan hizmetlerin çeşidi ve kullanıcıların sayısının artması ile birlikte siber dünyada bulunan bilgilerin önemi artmaktadır. Siber saldırılar, kullanıcıların bilgilerini ele geçirip, kullanmak dışında kullanıcıların hizmetlerden faydalanmasını engelleyerek, hem hizmet verenleri hem de kullanıcıları zor durumda bırakabilmektedir. DDoS saldırıları birçok noktadan, tek hedefe doğru, aynı anda yapılan zararlı paket gönderimidir. Gününmüzde hizmet sağlayanların sistem kapasitelerini gereksiz yere kullanmasını sağlayan bu saldırıları engellemek için; daha önce kayıt altına alınan saldırılar dışında, elinde bulunan diğer bilinen saldırılar ile karşılaştırma yaparak bilinmeyen saldırılarının bulunmasını sağlayan anomali tabanlı tespit yöntemleri kullanılmaktadır. Her DDoS saldırı çeşidinin farklı bir çalışma mekanizması vardır ve maalesef her saldırı tipi için genel bir çözüm bulunmamaktadır. Ayrıca makine öğrenmesi algoritmaları öncesinde kullanılan, verisetleri üzerinde yapılan ön işlemler saldırı tespit oranını ve tespit süresini etkilemektedir. Bu çalışmada ilk olarak HTTP flood, ICMP flood, TCP SYN flood ve UDP flood saldırılarının bulunduğu eğitim veriseti hazırlanmıştır. Hazırlanan veriseti üzerinde bulunan her bir DDoS saldırı çeşidi ayrı olarak incelenmiş, makine öğrenmesi algoritmaları ve ön işlemler uygulanarak model oluşturulmuştur. Ayrıca makine öğrenmesi algoritmaları öncesinde uygulanan ölçeklendirme ve özellik azaltma ön işlemlerinin, saldırı çeşitlerinin tespit, kesinlik, duyarlılık oranlarına ve tespit süresine etkileri analiz edilmiştir. Daha sonra Scapy kütüphanesi yardımıyla anlık veriler alınarak test veriseti olarak kullanılmıştır ve yapılandırılan model anlık veriler üzerinde uygulanarak çıkan tespit oranları ve çalışma süreleri kaydedilerek, karşılaştırma yapılmıştır. Yapılan ön işlemler ve uygulanan makine öğrenmesi algoritmalarının sonunda, farklı DDoS çeşitleri üzerinde farklı sonuçlar elde edilmiştir. Ayrıca farklı ön işlemlerin, farklı makine öğrenmesi algoritmalarına etkisi, süre ve tespit oranı olarak karşılaştırılmıştır. Çalışmanın sonunda, ön işlem yapılmadan uygulanan makine öğrenmesi algoritmalarının tek başına yeter-

siz olduğu, ölçeklendirme ön işleminin sadece SVC üzerinde olumlu etkileri olduğu, diğer algoritmalar için tespit, kesinlik ile duyarlılık oranlarının düşüşüne ve tespit süresinin uzamasına neden olduğu gözlemlenmiştir. Yapılan analizler sonucunda, özellikle azaltma ön işlemi ile birlikte uygulanan MLP, %99.2 tespit oranı ve 1.515 saniye toplam çalışma süresiyle en iyi çözüm olarak belirlenmiştir. Çalışmaya ek olarak, hazırlanan verisetlerinde DDoS saldırısının tespiti için uygulanan adımlar, NSL-KDD veriseti üzerinde uygulandığında, elde edilen sonuçların, hazırlanan verisetlerinden elde edilen sonuçlar gibi uygulanan makine öğrenme yöntemlerine ve uygulanan ön işlemlere göre farklılık gösterdiği gözlemlenmiştir. NSL-KDD veriseti için aynı adımlar uygulandığında, özellik azaltma sonucu MLP ve RF kullanılarak %89.9 olarak hesaplanmıştır.

KAYNAKLAR

Aamir, S., Zaidi, S. 2019. *DDoS Attack Detection with Feature Engineering and Machine Learning: the Framework and Performance Evaluation*. International Journal of Information Security , 18:7, 761–785.

Adrian, F. 2015. DDoSIM. Web Sitesi: https://sourceforge.net/projects/ddosim/ Erişim Tarihi: 08.03.2020.

Ahmed, A., Hameed S., Muhammed, R. ve Mirza, K.2020.*An Intelligent and Time-Efficient DDoS Identification Framework for Real-Time Enterprise Networks SAD-F: Spark Based Anomaly Detection Framework*. arXiv: 2001.08155, Web Sitesi: https://arxiv.org/abs/2001.08155.

AI-Kasassbeh, M., Al- Naymat, G., Hassanat, A. ve Almseidin M.2016.*Detecting distributed denial of service attacks using data mining techniques*.. International Journal of Advanced Computer Science and Applications 7(1). Şubat 2016. doi: 10.14569/IJAC SA.2016.070159.

Anonymous. 2007. Scikit-learn. Web Sitesi: https://scikit-learn.org/stable/about.html. Erişim Tarihi: 18.03.2020.

Anonymous. 2020. Pandas. Web Sitesi: https://pandas.pydata.org/. Erişim Tarihi: 18.03.2020.

Bargot, B.,Kulkarni,U. 2019.*A Survey of Different Tools and Approaches for AL-DDoS Attack*. 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom). 13-15 Mart 2019. New Delhi, India.

Başkaya, D., Samet, R. 2020 *DDoS Attacks Detection by Using Machine Learning Methods on Online Systems*. 2020 5th International Conference on Computer Science and Engineering (UBMK). 9-11 Eylül 2020. Diyarbakır, Turkey . doi: 10.1109/UBMK50275.2020.9219476.

Behal, S., Kumar, K. 2017.*Characterization and Comparison of DDoS Attack Tools and Traffic Generators A Review*. International Journal of Network Security, Vol.19, No.3, PP.383-393.

Berard, D. 2018. *A single DDoS attack can cost a company more than $400,000. Kasperksy*.. Kasperksy. Web Sitesi: https://usa.kaspersky.com/about/press-releases/2018_ddos-breach-costs-rise-to-over-2m-for-enterprises-finds-kaspersky-lab-report Erişim Tarihi: 01.04.2020.

Birget, J., Shende, S. ve Ramaswawi, S. 2018.*DDoS Detection and Mitigation Using Machine Learning*. MasterDissertation, Dept. Scientific Computing. Camden Graduate School, Rutgers, The State University of New Jersey.

Choo, K.,Dlodlo, M. 2016. *Analysing Feature Selection and Classification Techniques for DDoS Detection in Cloud*. Southern Africa Telecommunication Networks and Applications Conference (SATNAC). Fancourt, George, Western Cape, South Africa.

Deka, R., Bhattacharyya, D. ve Kalita, J. 2019.*Active Learning To Detect DDoS Attack Using Ranked Features*. Computer Communications. Volume 145, September 2019, Pages 203-222.

Dinçalp, U., Güzel, M.,Sevine,O., Bostancı, E. ve Askerzade, I. 2018.*Anomaly Based Distributed Denial of Service Attack Detection and Prevention with Machine Learning*. 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT).19-21 Oct.18309170.Ankara,Turkey.

Folino, G.,Sabatino, P. 2016.*Ensemble based collaborative and distributed intrusion detection systems: A survey*. Journal of Network and Computer Applications,16, 1-16.

Gu, Y.,Wang,Y.,Yang,Z.,Xiong,F and Gao,Y. 2017. *Multiple-Features-Based Semi-supervised Clustering DDoS Detection Method*. Hindawi Mathematical Problems in Engineering Volume 2017, Article ID 5202836, 10 pages. Web Sitesi: https://doi.org/10.1155/2017/5202836

Hummel, R. 2019. *Netscout Threat Intelligence Report*. Web Sitesi: https://www.netscout.com/threatreport. Erişim Tarihi: 04.04.2020.

Kupreev, O., Badovskaya, E. ve Gutnikov, A. 2020. *DDoS attacks in Q1 2020*.. Kasperksy. Web Sitesi: https://securelist.com/ddos-attacks-in-q1-2020/96837/ Erişim Tarihi: 30.05.2020.

Long, J., Wei, J., Zhang, Y., Zhao, J. ve Du, G. 2019.*An Effective RF-based Intrusion Detection Algorithm with Feature Reduction and Transformation*. Proceedings of the 21st International Conference on Enterprise Information Systems, Volume 2: ICEIS, 221-228.

Mahjabin, T., Xiao, Y., Sun, G. ve Jiang, W. 2017. *A Survey of Distributed Denial of Service Attack Prevention and Mitigation Techniques*.International Journal of Distributed Sensor Networks 13(12): 155014771774146. doi: 10.1177/155014771774 1463.

Motoma.2013. PyLoris. Web Sitesi: https://sourceforge.net/projects/pyloris/. Erişim Tarihi: 08.03.2020.

Özer, E., İskefiyeli, M. 2017. *Detection of DDoS Attack via Deep Packet Analysis in Real Time Systems* . 2017 International Conference on Computer Science and Engineering (UBMK). doi: 10.1109/UBMK.2017.8093526.

Paharia, B. ve Bhuskan, K. 2018. *DDoS Detection and Mitigation in Cloud via FogFiter: A Defence Mechanism*. 9th International Conference on Computing Communication and Networking Technologies. doi: 10.1109/ICCCNT.2018.8493704.

Pandian, A., Smys, S. 2019. *DDOS attack detection in telecommunication network using machine learning*. Journal of Ubiquitous Computing and Communication Technologies (UCCT) (2019) Vol.01/ No. 01Pages: 33-44.

Qassim. Q, Zin, A., Aziz, M. 2016. *Anomalies Classification Approach for Network-based Intrusion Detection System*.International Journal of Network Security, Vol.18, No.6, PP.1159-1172.

Rohitah, R., Moharir, M., Shobha, G. 2018. *SCAPY- A powerful interactive packet manipulation program*. 2018 International Conference on Networking, Embedded and Wireless Systems (ICNEWS). doi: 10.1109/ICNEWS.2018.8903954.

Russell, J. 2018. *The World's Largest DDoS Attack Took Github Offline for Fewer than 10 minutes*.. Web Sitesi: https://techcrunch.com/2018/03/02/the-worlds-largest-ddos-attack-took-github-offline-for-less-than-tens-minutes/ Erişim Tarihi:10.03.2020.

Sharma, A., Manzoor, I., Kumar, N. 2017. *A Feature Reduced Intrusion Detection System Using ANN Classifier*. 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT). Web Sitesi: https://doi.org/10.1016/j.eswa.2017.07.005.

Solarstone. 2015. Torshammer . Web Sitesi: https://sourceforge.net/projects/torshammer/. Erişim Tarihi: 08.03.2020.

Vanney, I. 2020. PyLoris. Web Sitesi: https://linuxhint.com/hping3/. Erişim Tarihi: 08.03.2020.

Zhao, S., Chandrashekar, M., Lee, Y. ve Medhi, D. 2015. *Real-Time Network Anomaly Detection System Using Machine Learning*.11th International Conference on the Design of Reliable Communication Networks (DRCN). 24-27 Mart 2015. doi: 10.1109/DRCN. 2015.7149025.

EK 1

SALDIRI ÇESITLERINE GÖRE ETIKETLENMIS VERILERDEN OLUSAN EGITIM VERISETI

0.000778	192.168.0	192.168.0.2		51986	443	TCP	66	52	0	0.000778	32	Not set	Set	Not set	Not set	0	64	HTTPFlood
0.001514	192.168.0	192.168.0.2		51986	443	TCP	192	178	126	0.001514	32	Not set	Set	Not set	Not set	0.000736	64	HTTPFlood
0.048025	192.168.0	192.168.0.2		51986	443	TCP	66	52	0	0.048025	32	Not set	Set	Not set	Not set	0.046511	64	HTTPFlood
0.129998	192.168.0	192.168.0.2		51948	443	TCP	67	53	1	0.129998	32	Not set	Set	Not set	Not set	0.081973	64	HTTPFlood
0.147956	192.168.0	192.168.0.2		51954	443	TCP	67	53	1	0.147956	32	Not set	Set	Not set	Not set	0.017958	64	HTTPFlood
0.192648	192.168.0	192.168.0.2		51954	443	TCP	66	52	0	0.192648	32	Not set	Set	Not set	Not set	0.044692	64	HTTPFlood
0.222592	192.168.0	192.168.0.2		51980	443	TCP	67	53	1	0.222592	32	Not set	Set	Not set	Not set	0.029944	64	HTTPFlood
0.25651	192.168.0	192.168.0.2		51974	443	TCP	67	53	1	0.25651	32	Not set	Set	Not set	Not set	0.033918	64	HTTPFlood
0.276429	192.168.0	192.168.0.2		51962	443	TCP	67	53	1	0.276429	32	Not set	Set	Not set	Not set	0.019919	64	HTTPFlood
0.27643	192.168.0	192.168.0.2		51968	443	TCP	67	53	1	0.27643	32	Not set	Set	Not set	Not set	0.000001	64	HTTPFlood
0.43207	192.168.0	192.168.0.2		51980	443	TCP	67	53	1	0.43207	32	Not set	Set	Not set	Not set	0.15564	64	HTTPFlood
1.006621	192.168.0	192.168.0.2		51988	443	TCP	74	60	0	1.006621	40	Set	Not set	Not set	Not set	0.574551	64	HTTPFlood
1.007294	192.168.0	192.168.0.2		51988	443	TCP	66	52	0	1.007294	32	Not set	Set	Not set	Not set	0.000673	64	HTTPFlood
1.007736	192.168.0	192.168.0.2		51988	443	TCP	583	569	517	1.007736	32	Not set	Set	Not set	Not set	0.000442	64	HTTPFlood
1.019425	192.168.0	192.168.0.2		51988	443	TCP	66	52	0	1.019425	32	Not set	Set	Not set	Not set	0.011689	64	HTTPFlood
1.019997	192.168.0	192.168.0.2		51988	443	TCP	192	178	126	1.019997	32	Not set	Set	Not set	Not set	0.000572	64	HTTPFlood
1.064265	192.168.0	192.168.0.2		51988	443	TCP	66	52	0	1.064265	32	Not set	Set	Not set	Not set	0.044268	64	HTTPFlood
1.16017	192.168.0	192.168.0.2		51956	443	TCP	67	53	1	1.16017	32	Not set	Set	Not set	Not set	0.095905	64	HTTPFlood
1.198045	192.168.0	192.168.0.2		51976	443	TCP	67	53	1	1.198045	32	Not set	Set	Not set	Not set	0.037875	64	HTTPFlood
1.20395	192.168.0	192.168.0.2		51956	443	TCP	66	52	0	1.20395	32	Not set	Set	Not set	Not set	0.005905	64	HTTPFlood
1.274813	192.168.0	192.168.0.2		51982	443	TCP	67	53	1	1.274813	32	Not set	Set	Not set	Not set	0.070863	64	HTTPFlood
1.274815	192.168.0	192.168.0.2		51950	443	TCP	67	53	1	1.274815	32	Not set	Set	Not set	Not set	0.000002	64	HTTPFlood
1.275305	192.168.0	192.168.0.2		51970	443	TCP	67	53	1	1.275305	32	Not set	Set	Not set	Not set	0.00049	64	HTTPFlood
1.484172	192.168.0	192.168.0.2		51982	443	TCP	67	53	1	1.484172	32	Not set	Set	Not set	Not set	0.208867	64	HTTPFlood
2.024824	192.168.0	192.168.0.2		51990	443	TCP	74	60	0	2.024824	40	Set	Not set	Not set	Not set	0.540652	64	HTTPFlood
2.025642	192.168.0	192.168.0.2		51990	443	TCP	66	52	0	2.025642	32	Not set	Set	Not set	Not set	0.000818	64	HTTPFlood
2.02602	192.168.0	192.168.0.2		51990	443	TCP	583	569	517	2.02602	32	Not set	Set	Not set	Not set	0.000378	64	HTTPFlood
2.056031	192.168.0	192.168.0.2		51990	443	TCP	66	52	0	2.056031	32	Not set	Set	Not set	Not set	0.030011	64	HTTPFlood
2.056702	192.168.0	192.168.0.2		51990	443	TCP	192	178	126	2.056702	32	Not set	Set	Not set	Not set	0.000671	64	HTTPFlood
2.111884	192.168.0	192.168.0.2		51990	443	TCP	66	52	0	2.111884	32	Not set	Set	Not set	Not set	0.055182	64	HTTPFlood
2.22426	192.168.0	192.168.0.2		51978	443	TCP	67	53	1	2.22426	32	Not set	Set	Not set	Not set	0.112376	64	HTTPFlood
2.224262	192.168.0	192.168.0.2		51964	443	TCP	67	53	1	2.224262	32	Not set	Set	Not set	Not set	0.000002	64	HTTPFlood
2.57335	192.168.0	192.168.0.2		51984	443	TCP	67	53	1	2.57335	32	Not set	Set	Not set	Not set	0.349088	64	HTTPFlood
2.784662	192.168.0	192.168.0.2		51984	443	TCP	67	53	1	2.784662	32	Not set	Set	Not set	Not set	0.211312	64	HTTPFlood
2.932322	192.168.0	192.168.0.2		51958	443	TCP	67	53	1	2.932322	32	Not set	Set	Not set	Not set	0.14766	64	HTTPFlood
2.976142	192.168.0	192.168.0.2		51958	443	TCP	66	52	0	2.976142	32	Not set	Set	Not set	Not set	0.04382	64	HTTPFlood
3.066006	192.168.0	192.168.0.2		51992	443	TCP	74	60	0	3.066006	40	Set	Not set	Not set	Not set	0.089864	64	HTTPFlood
3.066798	192.168.0	192.168.0.2		51992	443	TCP	66	52	0	3.066798	32	Not set	Set	Not set	Not set	0.000792	64	HTTPFlood
3.067291	192.168.0	192.168.0.2		51992	443	TCP	583	569	517	3.067291	32	Not set	Set	Not set	Not set	0.000493	64	HTTPFlood
3.097192	192.168.0	192.168.0.2		51992	443	TCP	66	52	0	3.097192	32	Not set	Set	Not set	Not set	0.029901	64	HTTPFlood
3.097697	192.168.0	192.168.0.2		51992	443	TCP	192	178	126	3.097697	32	Not set	Set	Not set	Not set	0.000505	64	HTTPFlood
3.144041	192.168.0	192.168.0.2		51992	443	TCP	66	52	0	3.144041	32	Not set	Set	Not set	Not set	0.046344	64	HTTPFlood
3.247551	192.168.0	192.168.0.2		51952	443	TCP	67	53	1	3.247551	32	Not set	Set	Not set	Not set	0.10351	64	HTTPFlood
3.275435	192.168.0	192.168.0.2		51960	443	TCP	67	53	1	3.275435	32	Not set	Set	Not set	Not set	0.027884	64	HTTPFlood
3.275437	192.168.0	192.168.0.2		51972	443	TCP	67	53	1	3.275437	32	Not set	Set	Not set	Not set	0.000002	64	HTTPFlood
3.275438	192.168.0	192.168.0.2		51966	443	TCP	67	53	1	3.275438	32	Not set	Set	Not set	Not set	0.000001	64	HTTPFlood
3.303348	192.168.0	192.168.0.2		51986	443	TCP	67	53	1	3.303348	32	Not set	Set	Not set	Not set	0.02791	64	HTTPFlood
3.316386	192.168.0	192.168.0.2		51954	443	TCP	67	53	1	3.316386	32	Not set	Set	Not set	Not set	0.013038	64	HTTPFlood
3.316388	192.168.0	192.168.0.2		51980	443	TCP	67	53	1	3.316388	32	Not set	Set	Not set	Not set	0.000002	64	HTTPFlood
3.320163	192.168.0	192.168.0.2		51960	443	TCP	66	52	0	3.320163	32	Not set	Set	Not set	Not set	0.003775	64	HTTPFlood

0.15897	192.168.0	192.168.0.2			44834	25	TCP	74	60	0		0.15897	40	Set	Not set	Not set	Not set	0	64	TCPFlood
0.158972	192.168.0	192.168.0.	67	68			UDP	342	328		308	0.158972						0.000002	255	TCPFlood
0.159331	192.168.0	192.168.0.2			44836	25	TCP	74	60	0		0.159331	40	Set	Not set	Not set	Not set	0.000359	64	TCPFlood
0.159587	192.168.0	192.168.0.2			44838	25	TCP	74	60	0		0.159587	40	Set	Not set	Not set	Not set	0.000256	64	TCPFlood
0.159853	192.168.0	192.168.0.2			44840	25	TCP	74	60	0		0.159853	40	Set	Not set	Not set	Not set	0.000266	64	TCPFlood
0.160103	192.168.0	192.168.0.2			44842	25	TCP	74	60	0		0.160103	40	Set	Not set	Not set	Not set	0.00025	64	TCPFlood
0.160351	192.168.0	192.168.0.2			44844	25	TCP	74	60	0		0.160351	40	Set	Not set	Not set	Not set	0.000248	64	TCPFlood
0.160613	192.168.0	192.168.0.2			44846	25	TCP	74	60	0		0.160613	40	Set	Not set	Not set	Not set	0.000262	64	TCPFlood
0.160857	192.168.0	192.168.0.2			44848	25	TCP	74	60	0		0.160857	40	Set	Not set	Not set	Not set	0.000244	64	TCPFlood
0.161154	192.168.0	192.168.0.2			44850	25	TCP	74	60	0		0.161154	40	Set	Not set	Not set	Not set	0.000297	64	TCPFlood
0.161409	192.168.0	192.168.0.2			44852	25	TCP	74	60	0		0.161409	40	Set	Not set	Not set	Not set	0.000255	64	TCPFlood
0.161693	192.168.0	192.168.0.2			44854	25	TCP	74	60	0		0.161693	40	Set	Not set	Not set	Not set	0.000284	64	TCPFlood
0.161952	192.168.0	192.168.0.2			44856	25	TCP	74	60	0		0.161952	40	Set	Not set	Not set	Not set	0.000259	64	TCPFlood
0.162228	192.168.0	192.168.0.2			44858	25	TCP	74	60	0		0.162228	40	Set	Not set	Not set	Not set	0.000276	64	TCPFlood
0.162229	192.168.0	192.168.0.2			44860	25	TCP	74	60	0		0.162229	40	Set	Not set	Not set	Not set	0.000001	64	TCPFlood
0.162524	192.168.0	192.168.0.2			44862	25	TCP	74	60	0		0.162524	40	Set	Not set	Not set	Not set	0.000295	64	TCPFlood
0.16279	192.168.0	192.168.0.2			44864	25	TCP	74	60	0		0.16279	40	Set	Not set	Not set	Not set	0.000266	64	TCPFlood
0.163054	192.168.0	192.168.0.2			44866	25	TCP	74	60	0		0.163054	40	Set	Not set	Not set	Not set	0.000264	64	TCPFlood
0.163326	192.168.0	192.168.0.2			44868	25	TCP	74	60	0		0.163326	40	Set	Not set	Not set	Not set	0.000272	64	TCPFlood
0.163327	192.168.0	192.168.0.2			44870	25	TCP	74	60	0		0.163327	40	Set	Not set	Not set	Not set	0.000001	64	TCPFlood
0.164266	192.168.0	192.168.0.2			44872	25	TCP	74	60	0		0.164266	40	Set	Not set	Not set	Not set	0.000939	64	TCPFlood
0.16455	192.168.0	192.168.0.2			44874	25	TCP	74	60	0		0.16455	40	Set	Not set	Not set	Not set	0.000284	64	TCPFlood
0.164831	192.168.0	192.168.0.2			44876	25	TCP	74	60	0		0.164831	40	Set	Not set	Not set	Not set	0.000281	64	TCPFlood
0.165111	192.168.0	192.168.0.2			44878	25	TCP	74	60	0		0.165111	40	Set	Not set	Not set	Not set	0.00028	64	TCPFlood
0.165391	192.168.0	192.168.0.2			44880	25	TCP	74	60	0		0.165391	40	Set	Not set	Not set	Not set	0.00028	64	TCPFlood
0.16567	192.168.0	192.168.0.2			44882	25	TCP	74	60	0		0.16567	40	Set	Not set	Not set	Not set	0.000279	64	TCPFlood
0.165949	192.168.0	192.168.0.2			44884	25	TCP	74	60	0		0.165949	40	Set	Not set	Not set	Not set	0.000279	64	TCPFlood
0.166229	192.168.0	192.168.0.2			44886	25	TCP	74	60	0		0.166229	40	Set	Not set	Not set	Not set	0.00028	64	TCPFlood
0.166512	192.168.0	192.168.0.2			44888	25	TCP	74	60	0		0.166512	40	Set	Not set	Not set	Not set	0.000283	64	TCPFlood
0.166516	192.168.0	192.168.0.2			44890	25	TCP	74	60	0		0.166516	40	Set	Not set	Not set	Not set	0.000004	64	TCPFlood
0.166516	192.168.0	192.168.0.2			44892	25	TCP	74	60	0		0.166516	40	Set	Not set	Not set	Not set	0	64	TCPFlood
0.167076	192.168.0	192.168.0.2			44894	25	TCP	74	60	0		0.167076	40	Set	Not set	Not set	Not set	0.00056	64	TCPFlood
0.16733	192.168.0	192.168.0.2			44896	25	TCP	74	60	0		0.16733	40	Set	Not set	Not set	Not set	0.000254	64	TCPFlood
0.167331	192.168.0	192.168.0.2			44898	25	TCP	74	60	0		0.167331	40	Set	Not set	Not set	Not set	0.000001	64	TCPFlood
0.167637	192.168.0	192.168.0.2			44900	25	TCP	74	60	0		0.167637	40	Set	Not set	Not set	Not set	0.000306	64	TCPFlood
0.167638	192.168.0	192.168.0.2			44902	25	TCP	74	60	0		0.167638	40	Set	Not set	Not set	Not set	0.000001	64	TCPFlood
0.16818	192.168.0	192.168.0.2			44904	25	TCP	74	60	0		0.16818	40	Set	Not set	Not set	Not set	0.000542	64	TCPFlood
0.168181	192.168.0	192.168.0.2			44906	25	TCP	74	60	0		0.168181	40	Set	Not set	Not set	Not set	0.000001	64	TCPFlood
0.174102	192.168.0	192.168.0.2			44908	25	TCP	74	60	0		0.174102	40	Set	Not set	Not set	Not set	0.005921	64	TCPFlood
0.174892	192.168.0	192.168.0.2			44910	25	TCP	74	60	0		0.174892	40	Set	Not set	Not set	Not set	0.00079	64	TCPFlood
0.175775	192.168.0	192.168.0.2			44912	25	TCP	74	60	0		0.175775	40	Set	Not set	Not set	Not set	0.000883	64	TCPFlood
0.176129	192.168.0	192.168.0.2			44914	25	TCP	74	60	0		0.176129	40	Set	Not set	Not set	Not set	0.000354	64	TCPFlood
0.176559	192.168.0	192.168.0.2			44916	25	TCP	74	60	0		0.176559	40	Set	Not set	Not set	Not set	0.00043	64	TCPFlood
0.176988	192.168.0	192.168.0.2			44918	25	TCP	74	60	0		0.176988	40	Set	Not set	Not set	Not set	0.000429	64	TCPFlood
0.178775	192.168.0	192.168.0.2			44920	25	TCP	74	60	0		0.178775	40	Set	Not set	Not set	Not set	0.001787	64	TCPFlood
0.179508	192.168.0	192.168.0.2			44922	25	TCP	74	60	0		0.179508	40	Set	Not set	Not set	Not set	0.000733	64	TCPFlood
0.179929	192.168.0	192.168.0.2			44924	25	TCP	74	60	0		0.179929	40	Set	Not set	Not set	Not set	0.000421	64	TCPFlood
0.179931	192.168.0	192.168.0.2			44926	25	TCP	74	60	0		0.179931	40	Set	Not set	Not set	Not set	0.000002	64	TCPFlood
0.180384	192.168.0	192.168.0.2			44928	25	TCP	74	60	0		0.180384	40	Set	Not set	Not set	Not set	0.000453	64	TCPFlood
0.180787	192.168.0	192.168.0.2			44930	25	TCP	74	60	0		0.180787	40	Set	Not set	Not set	Not set	0.000403	64	TCPFlood

10.69688	192.168.0	192.168.0.	55103	53			UDP	60	32	12	10.69688					0	128	UDPFlood
10.70396	192.168.0	192.168.0.	55103	53			UDP	60	32	12	10.70396					0.007085	128	UDPFlood
10.70441	192.168.0	192.168.0.	55102	53			UDP	60	32	12	10.70441					0.000442	128	UDPFlood
10.70441	192.168.0	192.168.0.	55101	53			UDP	60	32	12	10.70441					0.000002	128	UDPFlood
10.70441	192.168.0	192.168.0.	55100	53			UDP	60	32	12	10.70441					0.000001	128	UDPFlood
10.70441	192.168.0	192.168.0.	55099	53			UDP	60	32	12	10.70441					0.000001	128	UDPFlood
10.70441	192.168.0	192.168.0.	55098	53			UDP	60	32	12	10.70441					0	128	UDPFlood
10.70497	192.168.0	192.168.0.	55097	53			UDP	60	32	12	10.70497					0.000559	128	UDPFlood
10.70497	192.168.0	192.168.0.	55096	53			UDP	60	32	12	10.70497					0.000002	128	UDPFlood
10.70497	192.168.0	192.168.0.	55095	53			UDP	60	32	12	10.70497					0	128	UDPFlood
10.70497	192.168.0	192.168.0.	55094	53			UDP	60	32	12	10.70497					0.000001	128	UDPFlood
10.70497	192.168.0	192.168.0.	55093	53			UDP	60	32	12	10.70497					0.000001	128	UDPFlood
10.70551	192.168.0	192.168.0.	55092	53			UDP	60	32	12	10.70551					0.00054	128	UDPFlood
10.70552	192.168.0	192.168.0.	55091	53			UDP	60	32	12	10.70552					0.000002	128	UDPFlood
10.70552	192.168.0	192.168.0.	55090	53			UDP	60	32	12	10.70552					0	128	UDPFlood
10.70552	192.168.0	192.168.0.	55089	53			UDP	60	32	12	10.70552					0.000001	128	UDPFlood
10.70552	192.168.0	192.168.0.	55088	53			UDP	60	32	12	10.70552					0.000001	128	UDPFlood
10.70552	192.168.0	192.168.0.	55087	53			UDP	60	32	12	10.70552					0.000001	128	UDPFlood
10.7061	192.168.0	192.168.0.	55086	53			UDP	60	32	12	10.7061					0.000579	128	UDPFlood
10.7061	192.168.0	192.168.0.	55085	53			UDP	60	32	12	10.7061					0.000002	128	UDPFlood
10.7061	192.168.0	192.168.0.	55084	53			UDP	60	32	12	10.7061					0.000001	128	UDPFlood
10.7258	192.168.0	192.168.0.	55084	53			UDP	60	32	12	10.7258					0.019703	128	UDPFlood
10.72624	192.168.0	192.168.0.	55085	53			UDP	60	32	12	10.72624					0.000438	128	UDPFlood
10.72668	192.168.0	192.168.0.	55086	53			UDP	60	32	12	10.72668					0.000441	128	UDPFlood
10.72744	192.168.0	192.168.0.	55087	53			UDP	60	32	12	10.72744					0.000753	128	UDPFlood
10.72787	192.168.0	192.168.0.	55088	53			UDP	60	32	12	10.72787					0.000438	128	UDPFlood
10.72831	192.168.0	192.168.0.	55089	53			UDP	60	32	12	10.72831					0.000439	128	UDPFlood
10.72875	192.168.0	192.168.0.	55090	53			UDP	60	32	12	10.72875					0.000441	128	UDPFlood
10.72875	192.168.0	192.168.0.	55091	53			UDP	60	32	12	10.72875					0.000001	128	UDPFlood
10.72954	192.168.0	192.168.0.	55092	53			UDP	60	32	12	10.72954					0.000785	128	UDPFlood
10.72995	192.168.0	192.168.0.	55093	53			UDP	60	32	12	10.72995					0.000415	128	UDPFlood
10.73039	192.168.0	192.168.0.	55094	53			UDP	60	32	12	10.73039					0.000437	128	UDPFlood
10.73083	192.168.0	192.168.0.	55095	53			UDP	60	32	12	10.73083					0.00044	128	UDPFlood
10.73127	192.168.0	192.168.0.	55096	53			UDP	60	32	12	10.73127					0.000437	128	UDPFlood
10.73171	192.168.0	192.168.0.	55097	53			UDP	60	32	12	10.73171					0.000438	128	UDPFlood
10.7343	192.168.0	192.168.0.	55098	53			UDP	60	32	12	10.7343					0.002594	128	UDPFlood
10.73507	192.168.0	192.168.0.	55099	53			UDP	60	32	12	10.73507					0.000774	128	UDPFlood
10.73547	192.168.0	192.168.0.	55100	53			UDP	60	32	12	10.73547					0.000393	128	UDPFlood
10.73547	192.168.0	192.168.0.	55101	53			UDP	60	32	12	10.73547					0.000002	128	UDPFlood
10.73585	192.168.0	192.168.0.	55102	53			UDP	60	32	12	10.73585					0.000383	128	UDPFlood
10.7362	192.168.0	192.168.0.	55103	53			UDP	60	32	12	10.7362					0.000344	128	UDPFlood
10.73654	192.168.0	192.168.0.	55085	53			UDP	60	32	12	10.73654					0.000343	128	UDPFlood
10.73688	192.168.0	192.168.0.	55084	53			UDP	60	32	12	10.73688					0.000343	128	UDPFlood
10.75151	192.168.0	192.168.0.	55084	53			UDP	60	32	12	10.75151					0.014628	128	UDPFlood
10.75151	192.168.0	192.168.0.	55085	53			UDP	60	32	12	10.75151					0.000002	128	UDPFlood
10.75151	192.168.0	192.168.0.	55103	53			UDP	60	32	12	10.75151					0.000001	128	UDPFlood
10.75199	192.168.0	192.168.0.	55102	53			UDP	60	32	12	10.75199					0.000478	128	UDPFlood
10.75199	192.168.0	192.168.0.	55101	53			UDP	60	32	12	10.75199					0.000002	128	UDPFlood
10.75199	192.168.0	192.168.0.	55100	53			UDP	60	32	12	10.75199					0.000001	128	UDPFlood
10.752	192.168.0	192.168.0.	55099	53			UDP	60	32	12	10.752					0.000001	128	UDPFlood

0.000505	192.168.0	192.168.0.2				ICMP	60	28		0.000505					0	255	ICMPflood
1.016029	192.168.0	192.168.0.2				ICMP	60	28		1.016029					1.015524	255	ICMPflood
2.031582	192.168.0	192.168.0.2				ICMP	60	28		2.031582					1.015553	255	ICMPflood
3.047242	192.168.0	192.168.0.2				ICMP	60	28		3.047242					1.01566	255	ICMPflood
4.069339	192.168.0	192.168.0.2				ICMP	60	28		4.069339					1.022097	255	ICMPflood
5.085956	192.168.0	192.168.0.2				ICMP	60	28		5.085956					1.016617	255	ICMPflood
6.086859	192.168.0	192.168.0.2				ICMP	60	28		6.086859					1.000903	255	ICMPflood
7.088861	192.168.0	192.168.0.2				ICMP	60	28		7.088861					1.002002	255	ICMPflood
8.110338	192.168.0	192.168.0.2				ICMP	60	28		8.110338					1.021477	255	ICMPflood
9.109667	192.168.0	192.168.0.2				ICMP	60	28		9.109667					0.999329	255	ICMPflood
10.11917	192.168.0	192.168.0.2				ICMP	60	28		10.11917					1.009503	255	ICMPflood
11.13363	192.168.0	192.168.0.2				ICMP	60	28		11.13363					1.014464	255	ICMPflood
12.1349	192.168.0	192.168.0.2				ICMP	60	28		12.1349					1.001264	255	ICMPflood
13.13581	192.168.0	192.168.0.2				ICMP	60	28		13.13581					1.000914	255	ICMPflood
14.13691	192.168.0	192.168.0.2				ICMP	60	28		14.13691					1.001098	255	ICMPflood
15.14962	192.168.0	192.168.0.2				ICMP	60	28		15.14962					1.012706	255	ICMPflood
16.14933	192.168.0	192.168.0.2				ICMP	60	28		16.14933					0.999714	255	ICMPflood
17.14941	192.168.0	192.168.0.2				ICMP	60	28		17.14941					1.000078	255	ICMPflood
18.14845	192.168.0	192.168.0.2				ICMP	60	28		18.14845					0.999039	255	ICMPflood
19.14779	192.168.0	192.168.0.2				ICMP	60	28		19.14779					0.999344	255	ICMPflood
20.15967	192.168.0	192.168.0.2				ICMP	60	28		20.15967					1.011879	255	ICMPflood
21.17507	192.168.0	192.168.0.2				ICMP	60	28		21.17507					1.015404	255	ICMPflood
22.19657	192.168.0	192.168.0.2				ICMP	60	28		22.19657					1.021496	255	ICMPflood
23.21214	192.168.0	192.168.0.2				ICMP	60	28		23.21214					1.01557	255	ICMPflood
24.22979	192.168.0	192.168.0.2				ICMP	60	28		24.22979					1.017646	255	ICMPflood
25.24344	192.168.0	192.168.0.2				ICMP	60	28		25.24344					1.013654	255	ICMPflood
26.24424	192.168.0	192.168.0.2				ICMP	60	28		26.24424					1.0008	255	ICMPflood
27.24525	192.168.0	192.168.0.2				ICMP	60	28		27.24525					1.001011	255	ICMPflood
28.24654	192.168.0	192.168.0.2				ICMP	60	28		28.24654					1.001286	255	ICMPflood
29.24764	192.168.0	192.168.0.2				ICMP	60	28		29.24764					1.001105	255	ICMPflood
30.24956	192.168.0	192.168.0.2				ICMP	60	28		30.24956					1.00192	255	ICMPflood
31.25139	192.168.0	192.168.0.2				ICMP	60	28		31.25139					1.001827	255	ICMPflood
32.2528	192.168.0	192.168.0.2				ICMP	60	28		32.2528					1.001409	255	ICMPflood
33.26899	192.168.0	192.168.0.2				ICMP	60	28		33.26899					1.016195	255	ICMPflood
34.28416	192.168.0	192.168.0.2				ICMP	60	28		34.28416					1.015163	255	ICMPflood
35.30999	192.168.0	192.168.0.2				ICMP	60	28		35.30999					1.025829	255	ICMPflood
36.32152	192.168.0	192.168.0.2				ICMP	60	28		36.32152					1.011537	255	ICMPflood
37.35458	192.168.0	192.168.0.2				ICMP	60	28		37.35458					1.03306	255	ICMPflood
38.35239	192.168.0	192.168.0.2				ICMP	60	28		38.35239					0.997811	255	ICMPflood
39.35418	192.168.0	192.168.0.2				ICMP	60	28		39.35418					1.001786	255	ICMPflood
40.37052	192.168.0	192.168.0.2				ICMP	60	28		40.37052					1.016338	255	ICMPflood
41.37322	192.168.0	192.168.0.2				ICMP	60	28		41.37322					1.002698	255	ICMPflood
42.37397	192.168.0	192.168.0.2				ICMP	60	28		42.37397					1.000752	255	ICMPflood
43.39038	192.168.0	192.168.0.2				ICMP	60	28		43.39038					1.016417	255	ICMPflood
44.41254	192.168.0	192.168.0.2				ICMP	60	28		44.41254					1.022157	255	ICMPflood
45.42868	192.168.0	192.168.0.2				ICMP	60	28		45.42868					1.01614	255	ICMPflood
46.44372	192.168.0	192.168.0.2				ICMP	60	28		46.44372					1.01504	255	ICMPflood
47.46995	192.168.0	192.168.0.2				ICMP	60	28		47.46995					1.026231	255	ICMPflood
48.47505	192.168.0	192.168.0.2				ICMP	60	28		48.47505					1.005097	255	ICMPflood
49.49098	192.168.0	192.168.0.2				ICMP	60	28		49.49098					1.015926	255	ICMPflood

28.87455	192.168.0.	192.168.0.2				ICMP	98	84		28.87455					0	64	Normal
29.87517	192.168.0.	192.168.0.2				ICMP	98	84		29.87517					1.000623	64	Normal
30.8772	192.168.0.	192.168.0.2				ICMP	98	84		30.8772					1.002026	64	Normal
31.87925	192.168.0.	192.168.0.2				ICMP	98	84		31.87925					1.002054	64	Normal
32.88195	192.168.0.	192.168.0.2				ICMP	98	84		32.88195					1.002695	64	Normal
33.88469	192.168.0.	192.168.0.2				ICMP	98	84		33.88469					1.002744	64	Normal
34.88581	192.168.0.	192.168.0.2				ICMP	98	84		34.88581					1.001118	64	Normal
35.88871	192.168.0.	192.168.0.2				ICMP	98	84		35.88871					1.0029	64	Normal
36.89087	192.168.0.	192.168.0.2				ICMP	98	84		36.89087					1.002158	64	Normal
37.89364	192.168.0.	192.168.0.2				ICMP	98	84		37.89364					1.002769	64	Normal
38.89606	192.168.0.	192.168.0.2				ICMP	98	84		38.89606					1.00242	64	Normal
39.89785	192.168.0.	192.168.0.2				ICMP	98	84		39.89785					1.001796	64	Normal
40.90034	192.168.0.	192.168.0.2				ICMP	98	84		40.90034					1.002484	64	Normal
41.90237	192.168.0.	192.168.0.2				ICMP	98	84		41.90237					1.002032	64	Normal
42.90359	192.168.0.	192.168.0.2				ICMP	98	84		42.90359					1.001217	64	Normal
43.90583	192.168.0.	192.168.0.2				ICMP	98	84		43.90583					1.002244	64	Normal
44.90813	192.168.0.	192.168.0.2				ICMP	98	84		44.90813					1.002303	64	Normal
45.9103	192.168.0.	192.168.0.2				ICMP	98	84		45.9103					1.002165	64	Normal
46.91298	192.168.0.	192.168.0.2				ICMP	98	84		46.91298					1.002687	64	Normal
47.91519	192.168.0.	192.168.0.2				ICMP	98	84		47.91519					1.002209	64	Normal
48.91766	192.168.0.	192.168.0.2				ICMP	98	84		48.91766					1.002468	64	Normal
49.91985	192.168.0.	192.168.0.2				ICMP	98	84		49.91985					1.002191	64	Normal
50.92141	192.168.0.	192.168.0.2				ICMP	98	84		50.92141					1.001562	64	Normal
51.92425	192.168.0.	192.168.0.2				ICMP	98	84		51.92425					1.002839	64	Normal
52.92625	192.168.0.	192.168.0.2				ICMP	98	84		52.92625					1.001994	64	Normal
53.92921	192.168.0.	192.168.0.2				ICMP	98	84		53.92921					1.002964	64	Normal
54.93132	192.168.0.	192.168.0.2				ICMP	98	84		54.93132					1.002111	64	Normal
55.9337	192.168.0.	192.168.0.2				ICMP	98	84		55.9337					1.002382	64	Normal
56.9353	192.168.0.	192.168.0.2				ICMP	98	84		56.9353					1.001595	64	Normal
57.93737	192.168.0.	192.168.0.2				ICMP	98	84		57.93737					1.002072	64	Normal
58.93919	192.168.0.	192.168.0.2				ICMP	98	84		58.93919					1.001823	64	Normal
59.10405	192.168.0.	192.168.0.	67	68		UDP	342	328	308	59.10405					0.164858	255	Normal
59.94126	192.168.0.	192.168.0.2				ICMP	98	84		59.94126					0.837209	64	Normal
60.94433	192.168.0.	192.168.0.2				ICMP	98	84		60.94433					1.003066	64	Normal
61.94679	192.168.0.	192.168.0.2				ICMP	98	84		61.94679					1.002464	64	Normal
62.94807	192.168.0.	192.168.0.2				ICMP	98	84		62.94807					1.001274	64	Normal
63.95037	192.168.0.	192.168.0.2				ICMP	98	84		63.95037					1.0023	64	Normal
64.95214	192.168.0.	192.168.0.2				ICMP	98	84		64.95214					1.001775	64	Normal
65.95363	192.168.0.	192.168.0.2				ICMP	98	84		65.95363					1.00149	64	Normal
66.95542	192.168.0.	192.168.0.2				ICMP	98	84		66.95542					1.001786	64	Normal
67.95773	192.168.0.	192.168.0.2				ICMP	98	84		67.95773					1.002311	64	Normal
68.95985	192.168.0.	192.168.0.2				ICMP	98	84		68.95985					1.002125	64	Normal
69.96236	192.168.0.	192.168.0.2				ICMP	98	84		69.96236					1.002506	64	Normal
70.96456	192.168.0.	192.168.0.2				ICMP	98	84		70.96456					1.002199	64	Normal
71.96738	192.168.0.	192.168.0.2				ICMP	98	84		71.96738					1.00282	64	Normal
72.96929	192.168.0.	192.168.0.2				ICMP	98	84		72.96929					1.00191	64	Normal
73.97145	192.168.0.	192.168.0.2				ICMP	98	84		73.97145					1.002161	64	Normal
74.97285	192.168.0.	192.168.0.2				ICMP	98	84		74.97285					1.001405	64	Normal
75.9743	192.168.0.	192.168.0.2				ICMP	98	84		75.9743					1.001448	64	Normal
76.97611	192.168.0.	192.168.0.2				ICMP	98	84		76.97611					1.001805	64	Normal

EK 2

HAZIRLANAN VERISETLERINDE ÖN İSLEM UYGULANMADAN ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%0.0	%0.0	%0.0
	ICMP flood	%0.0	%0.0	%0.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0
RF	HTTP flood	%0.0	%0.0	%0.0
	ICMP flood	%0.0	%0.0	%0.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%1.0	%1.0	%1.0
KNN	HTTP flood	%1.0	%0.05434	%0.10308
	ICMP flood	%0.0	%0.0	%0.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%1.0	%0.032051	%0.06211
	UDP flood	%0.0	%0.0	%0.0
SVC	HTTP flood	%0.0	%0.0	%0.0
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%0.0	%0.0	%0.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%1.0	%1.0	%1.0

EK 3

HAZIRLANAN VERISETLERINDE ÖLÇEKLENDIRME ÖN İSLEMI UYGULANDIKTAN SONRA ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0	%0.00201	%0.00402
	ICMP flood	%1.0	%0.04000	%0.07692
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%1.0	%0.00040	%0.00080
RF	HTTP flood	%0.0	%0.0	%0.0
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%0.20008	%0.28195
	TCP SYN flood	%1.0	%0.01782	%0.03503
	UDP flood	%0.0	%0.0	%0.0
KNN	HTTP flood	%0.0	%0.0	%0.0
	ICMP flood	%1.0	%0.10008	%0.18195
	Normal	%1.0	%0.10008	%0.38195
	TCP SYN flood	%1.0	%0.027265	%0.05308
	UDP flood	%0.0	%0.0	%0.0
SVC	HTTP flood	%1.0	%0.06040	%0.06080
	ICMP flood	%1.0	%0.10008	%0.38195
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0

EK 4

HAZIRLANAN VERISETLERINDE ÖZELLIK AZALTMA ÖN İSLEMI UYGULADIKTAN SONRA ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0)	%1.0	%1.0
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%1.0	%1.0	%1.0
	UDP flood	%1.0	%1.0	%1.0
RF	HTTP flood	%1.0	%1.0	%1.0
	ICMP flood	%0.0	%0.0	%0.0
	Normal	%1.0	%0.618778	%0.68287
	TCP SYN flood	%1.0	%0.81877	%0.82877
	UDP flood	%1.0	%1.0	%1.0
KNN	HTTP flood	%1.0	%0.96942	%0.98447
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%0.96942	%0.98447
	TCP SYN flood	%1.0	%0.89351	%0.94376
	UDP flood	%1.0	%1.0	%1.0
SVC	HTTP flood	%0.0	%0.0	%0.0
	ICMP flood	%0.0	%0.0	%0.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%1.0	%1.0	%1.0
	UDP flood	%0.0	%0.0	%0.0

EK 5

HAZIRLANAN VERISETLERINDE ÖZELLIK AZALTMA VE ÖLÇEKLENDIRME ÖN İSLEMLERI UYGULADIKTAN SONRA ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0)	%0.18518	%0.3125
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%0.28518	%0.3725
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0
RF	HTTP flood	%1.0	%0.0032154	%0.00641
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0
KNN	HTTP flood	%1.0	%0.00240	%0.004803
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%0.0	%0.0	%0.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0
SVC	HTTP flood	%1.0	%1.0	%1.0
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%1.0	%1.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%1.0	%1.0	%1.0

EK 6

NSL-KDD VERISETLERINDE ÖN İSLEM UYGULANMADAN ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0	%0.994572	%0.997278
	ICMP flood	%1.0	%0.99969	%0.99984
	Normal	%1.0	%0.99785	%0.99892
	TCP SYN flood	%1.0	%0.999956	%0.99997
	UDP flood	%0.0	%0.0	%0.0
RF	HTTP flood	%1.0	%0.99592	%0.99796
	ICMP flood	%1.0	%0.992147	%0.99605
	Normal	%1.0	%0.999948	%0.99997
	TCP SYN flood	%1.0	%0.999978	%0.99998
	UDP flood	%0.0	%0.0	%0.0
KNN	HTTP flood	%1.0	%0.957937	%0.9785169
	ICMP flood	%1.0	%0.997885	%0.998941
	Normal	%1.0	%0.9926803	%0.996326
	TCP SYN flood	%1.0	%0.9993681	%0.99968
	UDP flood	%0.0	%0.0	%0.0
SVC	HTTP flood	%0.9993681	%0.999368	%0.9993681
	ICMP flood	%0.593367	%0.593367	%0.593367
	Normal	%0.0	%0.0	%0.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0

EK 7

NSL-KDD VERISETLERINDE ÖLÇEKLENDIRME ÖN İSLEMI UYGULANDIKTAN SONRA ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0	%0.987788	%0.993856
	ICMP flood	%1.0	%00.95983	%0.979503
	Normal	%1.0	%0.999896	%0.999948
	TCP SYN flood	%1.0	%0.999782	%0.99989
	UDP flood	%0.0	%0.0	%0.0
RF	HTTP flood	%1.0	%0.451831	%0.622429
	ICMP flood	%1.0	%0.345514	%0.513580
	Normal	%1.0	%0.999961	%0.99998
	TCP SYN flood	%1.0	%0.292450	%0.452551
	UDP flood	%0.0	%0.0	%0.0
KNN	HTTP flood	%1.0	%0.770691	%0.8704980
	ICMP flood	%1.0	%0.905164	%0.950221
	Normal	%1.0	%0.950221	%0.999980
	TCP SYN flood	%1.0	%0.999869	%0.999934
	UDP flood	%0.0	%0.0	%0.0
SVC	HTTP flood	%0.999825	%0.9998256	%0.999825
	ICMP flood	%0.32700729	%0.327007	%0.3270072
	Normal	%1.0	%0.9999610	%0.99998
	TCP SYN flood	%0.99982568	%0.9998256	%0.9998256
	UDP flood	%0.0	%0.0	%0.0

EK 8

NSL-KDD VERISETLERINDE ÖZELLIK AZALTMA ÖN İSLEMI UYGULANDIKTAN SONRA ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0)	%0.99864	%0.99932
	ICMP flood	%1.0	%0.99969	%0.99984
	Normal	%1.0	%0.9974173	%0.998707
	TCP SYN flood	%1.0	%0.999803	%0.999901
	UDP flood	%1.0	%0.59980	%0.599901
RF	HTTP flood	%1.0	%0.9972862	%0.998641
	ICMP flood	%1.0	%1.0	%1.0
	Normal	%1.0	%0.9999351	%0.999967
	TCP SYN flood	%1.0	%1.0	%1.0
	UDP flood	%1.0	%0.5498039	%0.54990194
KNN	HTTP flood	%1.0	%0.9579375	%0.9785169
	ICMP flood	%1.0	%0.997885	%0.9989417
	Normal	%1.0	%0.99281014	%0.9963921
	TCP SYN flood	%1.0	%0.9993245	%0.99966215
	UDP flood	%0.5041467	%0.5041467	%0.5041467
SVC	HTTP flood	%0.999697	%0.999697	%0.99969797
	ICMP flood	%0.574146	%0.57414678	%0.574146
	Normal	%0.0	%0.0	%0.0
	TCP SYN flood	%0.0	%0.0	%0.0
	UDP flood	%0.0	%0.0	%0.0

EK 9

NSL-KDD VERISETLERINDE ÖZELLIK AZALTMA VE ÖLÇEKLENDIRME ÖN İSLEMLERI UYGULANDIKTAN SONRA ALINAN RAPORLAR

Algoritmalar	Saldırı Çeşitleri	Kesinlik	Duyarlılık	F1-skor
MLP	HTTP flood	%1.0)	%0.99728629	%0.99864130
	ICMP flood	%1.0	%0.9948655	%0.9974261
	Normal	%1.0	%0.99988319	%0.9999415
	TCP SYN flood	%0.9998910	%0.99989105	%0.9998910
	UDP flood	%0.5141467	%0.5141467	%0.51414678
RF	HTTP flood	%1.0	%0.4097693	%0.5813282
	ICMP flood	%1.0	%0.93748112	%0.9677318
	Normal	%1.0	%0.99994808	%0.9999740
	TCP SYN flood	%1.0	%0.999978211	%0.99998910
	UDP flood	%0.514146	%0.5141467	%0.5141467
KNN	HTTP flood	%1.0	%0.83446404	%0.90976331
	ICMP flood	%1.0	%0.9051646	%0.95022194
	Normal	%1.0	%0.9996236	%0.9998117
	TCP SYN flood	%1.0	%0.9998692	%0.999934
	UDP flood	%0.5041467	%0.504146	%0.504146
SVC	HTTP flood	%1.0	%0.9999782	%0.999989
	ICMP flood	%0.6141467	%0.6141467	%0.6141467
	Normal	%1.0	%0.9099782	%0.9099891
	TCP SYN flood	%1.0	%0.994978	%0.9949891
	UDP flood	%0.534146	%0.53414	%0.5341467

ÖZGEÇMİŞ

Adı Soyadı : Dilek BAŞKAYA

Doğum Yeri : Eskişehir

Doğum Tarihi : 21/08/1994

Medeni Hali : Bekar

Yabancı Dili : İngilizce

Eğitim Durumu (Kurum ve Yıl):

Lise : Tayfur Bayar Lisesi (2012)

Lisans : Ankara Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü (2017)

Yüksek Lisans : Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı (09/2018- 11/2020)

Çalıştığı Kurum/Kurumlar ve Yıl: Sekom iletişim Sistemleri

Yayınları: DDoS Attacks Detection by Using Machine Learning Methods on Online Systems. 5. Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı.9-11 Eylül, Dicle Üniversitesi, Bildiri Özetleri Kitabı, 63-68, Diyarbakır.