NESNELERİN İNTERNETİ SİSTEMLERİNE YAPILAN SALDIRILARIN ANALİZİ ÜZERİNE TUZAK SİSTEMLER İLE BİR DURUM ÇALIŞMASI

A CASE STUDY ON THE ANALYSIS OF ATTACKS AGAINST INTERNET OF THINGS SYSTEMS WITH HONEYPOT SYSTEMS

Ferhat DAL

Doç.Dr. Harun ARTUNER

Tez Danışmanı

Hacettepe Üniversitesi
Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin
Adli Bilimler Anabilim Dalı İçin Öngördüğü
YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2019

FERHAT DAL'ın hazırladığı "**Nesnelerin İnterneti Sistemlerine Yapılan Saldırıların Analizi Üzerine Tuzak Sistemler İle Bir Durum Çalışması**" adlı bu çalışma aşağıdaki jüri tarafından **ADLİ BİLİMLER ANABİLİM DALI**'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Doç. Dr. Bünyamin CİYLAN

Başkan .....

Doç. Dr. Harun ARTUNER

Danışman .....

Doç. Dr. Kayhan İMRE

Üye .....

Doç. Dr. Sevil ŞEN

Üye .....

Doç. Dr. Ahmet Burak CAN

Üye .....

Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından **Yüksek Lisans TEZİ** olarak ..... / ..... /..... tarihinde onaylanmıştır.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU

Fen Bilimleri Enstitüsü Müdürü

ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içerisindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başka kaynaklardan yararlanılması durumunda ilgili kaynaklara bilimsel normlara uygun olarak atıfta bulunulduğunu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede bir tez çalışması olarak sunmadığımı beyan ederim.

30/09/2019

Ferhat DAL

YAYINLANMA FİKRİ MÜLKİYET HAKKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kâğıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan "***Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge***" kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- ☐ Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir.
- ☐ Enstitü / Fakülte yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren .... ay ertelenmiştir.
- ☐ Tezim ile ilgili gizlilik kararı verilmiştir.

30/09/2019

(İmza)

Ferhat DAL

ÖZET

NESNELERİN İNTERNETİ SİSTEMLERİNE YAPILAN SALDIRILARIN ANALİZİ ÜZERİNE TUZAK SİSTEMLER İLE BİR DURUM ÇALIŞMASI

Ferhat DAL
Yüksek Lisans, Adli Bilimler Anabilim Dalı
Tez Danışmanı: Doç.Dr. Harun ARTUNER
Eylül 2019 67 Sayfa

Teknolojinin gelişimi ile beraber İnternete bağlanabilen cihaz sayısı giderek yaygınlaşmaktadır. Günümüzde neredeyse her alanda İnternetin sağladığı tüm imkânlardan faydalanılmaktadır. Bu cihazların bilinirliğinin artmasıyla birlikte siber saldırganlar da bu sistemleri daha çok hedef tahtasına koymuşlardır. Bu sistemler bilinen bilgisayar ya da sunucu gibi sistemlerin sahip olduğu savunma mekanizmalarına sahip değildirler. Bu yüzden savunma amaçlı yöntemlerde klasik teknikler uygulanamamaktadır. Nesnelerin İnterneti (IoT) en basit anlamada, internet bağlı olan herhangi bir fiziksel cihaz olabilir. Nesnelerin İnternetinde kullanılan sensör cihazların kaynakları, yani bellek (ROM ve RAM), CPU ve depolama alanı sınırlıdır. Genellikle, veri toplayan sensörler, makineden makineye (M2M) veri iletimi yaparak topladıkları verileri belirli bir merkeze gönderirler. Bu tür aygıtlar bir ağa bağlandığında Nesnelerin İnternetinin (IoT) bir parçasını oluştururlar. Bu sistemlere erişimi sağlamak ve yönetmek için sistemlerinde yönetim protokollerinden biri ya da birkaçı aktif olmalıdır. Bu çalışmada, Nesnelerin İnterneti yönetim protokolü Telnet ve özellikle yaygın olarak kullanılan MQTT protokolü için bir saldırı benzetim ortamı ve tuzak sistem tasarlanmıştır. Bu çalışma söz konusu protokollere yapılan ve yaygın olarak bilinen

saldırı yöntemlerini kapsamaktadır. Çalışmada her bir protokol için bilinen ataklar denenmiş ve sonuçları paylaşılmıştır. Ayrıca gerçek bir IP kamera sistemine çeşitli saldırılar yapılarak sonuçları paylaşılmıştır.

Anahtar Kelimeler: IoT Güvenliği, Telnet, MQTT, IoT Saldırıları, Saldırı Benzetimi, Nesnelerin İnterneti, Bilgi Güvenliği, Tuzak Sistemler, Siber Saldırılar

ABSTRACT

A CASE STUDY ON THE ANALYSIS OF ATTACKS AGAINST INTERNET OF THINGS SYSTEMS WITH HONEYPOT SYSTEMS

Ferhat DAL
Master of Science, Department of Forensic Science,
Supervisor: Dr. Lecturer Harun ARTUNER
September 2019, 67 Pages

With the development of technology, devices connected to the Internet are becoming increasingly widespread. Almost every field in the present day is making use of the opportunities provided by the Internet. With the increasing awareness of these devices, cyber attackers have also put these systems on the target board. These systems do not have the defense mechanisms that systems like computers or servers have. Therefore, classical techniques can't be applied in defensive methods. In the simplest sense, the internet of things (IoT) can be any physical device connected to the internet. The resources of the sensor devices used on the Internet of things, namely memory (ROM and RAM), CPU and storage space are limited. Generally, sensors that collect data transmit data from the machine to the machine (M2M) and send the data to a specific center. Such devices form part of the Internet of Things (IOT) when connected to a network. One or more of the management protocols must be active in their systems to provide and manage access to these systems. In this work, an attack simulation environment and honeypot system is designed for the Telnet protocol of internet of things and the commonly used MQTT protocol. This study covers commonly known attack methods made by the aforementioned protocols. In the study, known attacks for

each protocol were tried and the results were shared. In addition, various attacks were implemented to a real IP camera system and the results were shared.

Key words: IoT Security, Telnet, MQTT, IoT Attacks, IoT Attack Simulation, Internet of Things, Information Security, Secure Communication, Cyber Attacks

TEŞEKKÜR

Başta tez danışmanım Doç.Dr. Harun ARTUNER olmak üzere, bu çalışmada ki katkı ve desteklerinden dolayı Doç.Dr. Bünyamin CİYLAN, Doç.Dr. Kayhan İMRE, Doç.Dr. Ahmet Burak CAN ve Doç.Dr. Sevil Şen AKGÜNDÜZ'e teşekkürlerimi sunarım.

Ayrıca, sonsuz destekleri ve gösterdikleri sabır dolayısı ile sevgili eşim Ebru ve kızım Aysima'ya sonsuz sevgilerimi sunarım.

İÇİNDEKİLER

TABLOLAR

ŞEKİLLER

SİMGELER VE KISALTMALAR

Simgeler

EB	Exabyte
MB	Megabit

Kısaltmalar

IoT	Internet of Things
TCP	Transmission Control Protokol
RFID	Radio-Frequency Identification
QoS	Quality of Service
Telnet	Telemetry Network
SSH	Secure Shell
HTTP	Hyper Text Transfer Protocol
MQTT	Message Queuing Telemetry Transport
IP	Internet Protocol
GPS	Global Positioning System
DoS	Denial of Service
DdoS	Distributed Denial of Service
CoAP	Constrained Application Protocol
PKI	Public Key Infrastructure
IEEE	Institute of Electrical and Electronics Engineers
MAC	Media Access Control
Wi-Fi	Wireless Fidelity
TDMA	Time-division Multiple Access
AES	Advances Encryption System
CSMA/CA	Carrier Sense Multipla Access / Collision Avoidance
ISM	Industrial Scientific Medical

OFDMA	Orthogonal Frequency-division Multiple Access
CN	Core Network
RAN	Radio Access Network
AMQP	Advanced Message Queuing Protocol
OMG	Object Management Group
DTK	Deception Toolkit
TLS	Transport Layer Security
SFTP	Secure File Transfer Protocol
HTML	Hyper Text Markup Language
CWMP	CPE WAN Management Protocol
XML	eXtensible Markup Language
JSON	JavaScript Object Notation
SMTP	Simple Mail Transfer Protocol

1.GİRİŞ

İnternetin ve teknolojinin gelişmesi ile beraber, internete bağlanabilen cihaz sayısı her geçen gün artmaktadır. Nesnelerin interneti (Internet of Things)'nin kesin tanımı hala bir tartışma konusudur ve IoT'yi tanımlamak için çeşitli tanımlar yapılmıştır. Fakat en basit tanımıyla nesnelerin interneti, internete bağlanabilen her şey anlamına gelmektedir[1]. Nesnelerin interneti ilk olarak 1999 yılında Kevin Ashton tarafından RFID (Radio-Frequency Identification) kullanarak tedarik zinciri tanımını yapmak için kullanılmıştır[2]. Bu kullanım geçen zaman diliminde sürekli artış göstermiş ve farklı alanlarda kullanılmaya devam etmiştir. Nesnelerin internetinin insanların yaşamlarına birçok kolaylık kazandırmasıyla birlikte çeşitli tehditlerde ortaya çıkarmıştır. Bu sistemlerin geleneksel güvenlik mekanizmalarına sahip olmayışı saldırganların bu sistemleri hedefe koymasına sebebiyet vermiştir.

4.Sanayi Devrimi ile birlikte akıllı evlerin ve sürücüsüz araçların giderek yaygınlaştığı dünyada insanlar hızla teknolojiye bağımlı hale gelmişlerdir. Bunun sonucunda bireylerin iş yerinde çalışırken ya da evinde dinlenirken her an bir siber saldırıya maruz kalabilmeleri yüksek ihtimal dâhilindedir. Bu sistemlerin yeterli kaynaklara sahip olamayışı onları daha da savunmasız hale getirmiştir. Bu sistemlerin fiziksel ve ağ güvenliğinin sağlanması konusunda bazı çalışmalar yapılmaktadır. Fakat sistemlerin kaynak yetersizliği bu konuda çeşitli problemleri ortaya çıkarmaktadır. Bunlardan en önemlisi kullanılan nesnelerin kısıtlı bant genişliği, hafıza ve yeterli hesaplama gücünün olmayışından dolayı bilinen ve yüksek işlem gücü gerektiren savunma mekanizmalarının uygulanmasının imkânsız olmasıdır. Bu sistemlerin sürekli kontrolü ve izlenmesi de yeterli seviyede olmadığı için olası bir aykırılık (anomali) tespiti yada saldırı engellemesinin yapılması oldukça zordur.

Tuzak sistemler, bilgi teknolojileri içerisinde saldırıların tespit edilmesi konusunda uygulanan önemli sistemler arasında yer almaktadır. Tuzak sistemler de temel amaç gerçek bir sistem gibi davranarak saldırganın ilgisini çekip daha sonra yapılan tüm aktiviteleri kayıt altına almaktır. Günümüzde birçok sistemin benzetimini yapan tuzak sistem bulunmaktadır. Nesnelerin interneti uygulamalarının yaygınlaşması bu alanda

da tuzak sistemlerin gelişimini hızlandırmıştır. Güncel olarak Telnet, SSH ve Http gibi protokollerin benzetimini yapan nesnelerin interneti tuzak sistemleri bulunmaktadır.

Bu tez çalışması kapsamında, Nesnelerin İnterneti yönetim protokollerinden olan Telnet (Telemetry Network) ve özellikle yaygın olarak kullanılan MQTT (Message Queuing Telemetry Transport) protokolü için bir saldırı benzetim ortamı ve tuzak sistem tasarlanmıştır. Aynı zamanda gerçek bir senaryo ile IP kamera sistemine çeşitli saldırılar yapılmış ve sonuçları analiz edilmiştir. Bu çalışma söz konusu protokollere yapılan ve yaygın olarak bilinen saldırı tekniklerini kapsamaktadır. Çalışmada her bir protokol için bilinen ataklar denenmiş ve sonuçları paylaşılmıştır.

2.NESNELERİN İNTERNETİ

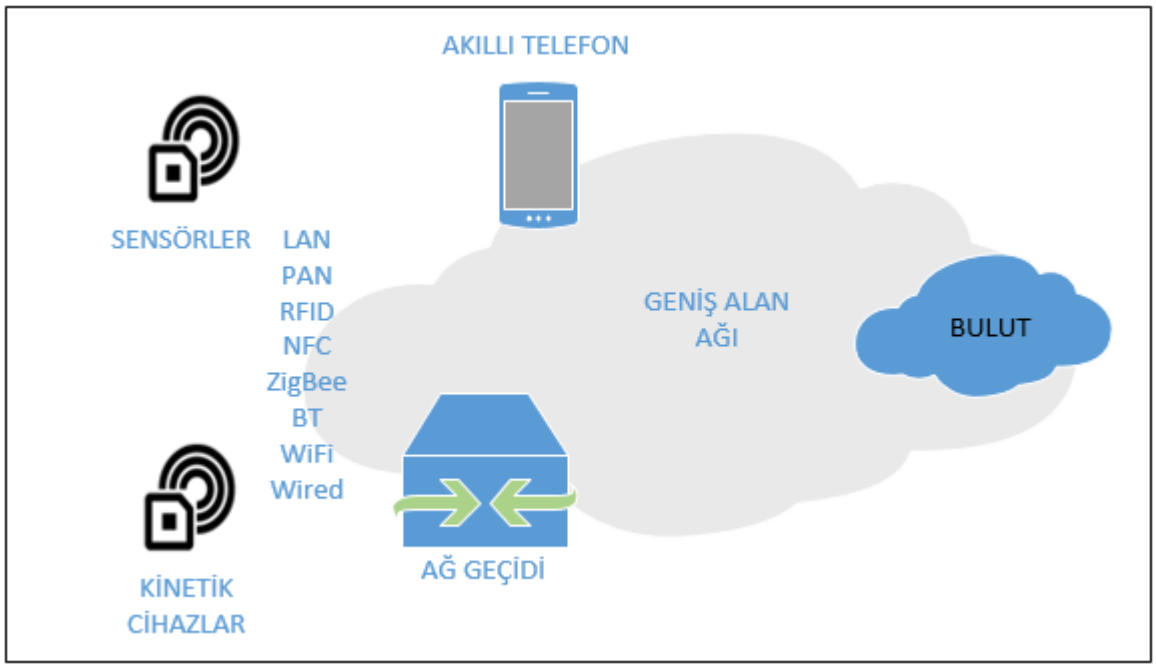
2.1 Giriş

Kesin tanımı hala bir tartışma konusu olsa da en basit tanımıyla nesnelerin interneti, internete bağlanabilen her şey anlamına gelmektedir[1]. Bunun dışında yapılan diğer bazı nesnelerin interneti tanımları aşağıda verilmiştir.

- Sistemlerin en az insan müdahalesiyle veri üretmesini, değiştirmesini ve tüketmesini sağlayan bilgisayarların iletişim ve hesaplama kabiliyetini artırma yeteneğidir[3].
- Nesnelerin interneti temelde, hesaplama, iletişim kurma, anlamlandırma ve harekete geçme yeteneğidir[4].
- IoT, her yerden, her zaman ve her şeyden bilgi almak veya durumunu değiştirmek için erişilebilen, benzersiz olarak tanımlanabilen “nesnelerden” oluşan bir ağdır[5].

Bu terimin çeşitli tanımları olsa da burada dikkat çeken konu ismini veren ana bileşenlerdir, “İnternet” ve “Nesneler”. Nesneleri çeşitli bilgileri toplayan sensörler olarak düşündüğümüzde, sensörlerin internet altyapısını kullanarak bu bilgileri işlemek üzere bir merkeze göndermesi ile bu tanımları genel olarak kapsamış olur. İstenilen yerden ve istenilen zamanda bilginin elde edilmesini amaçlayan bu kavram ile birçok

farklı tipte cihaz bir araya gelir ve bilgi paylaşımı yaparak nesnelerin internetini oluşturur. Nesnelerin interneti genel topolojisi Şekil 2.1'de gösterilmiştir.

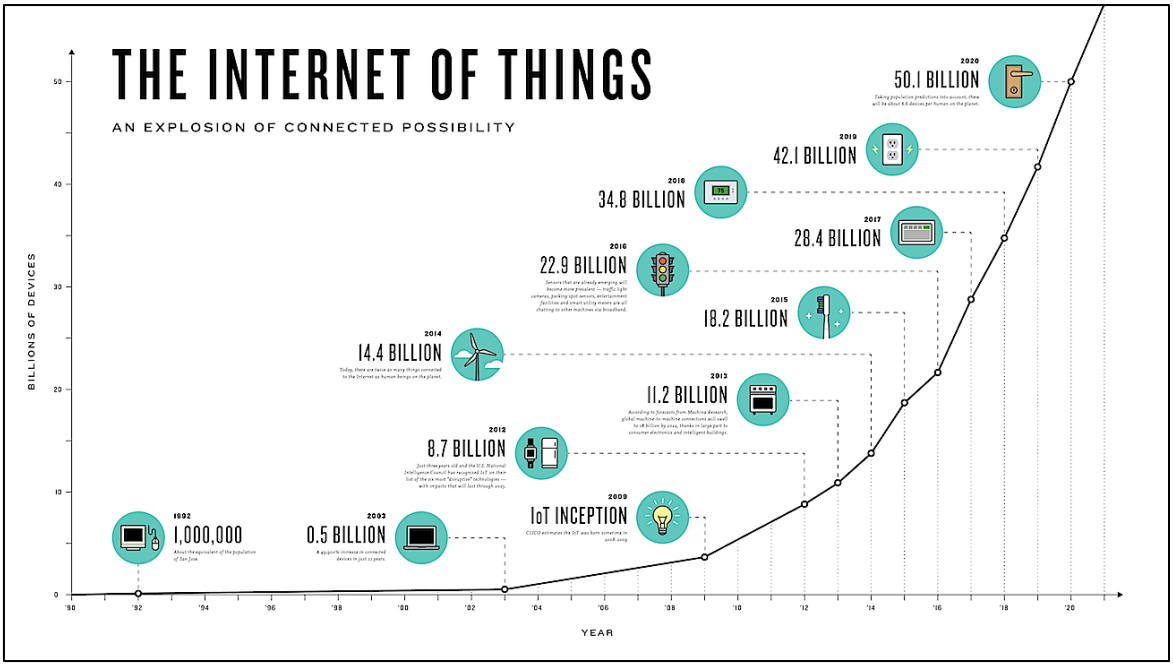


Şekil 2.1 Nesnelerin İnterneti Genel Topolojisi [5]

2.2 Tarihsel Gelişimi ve Gelecek Öngörüsü

Nesnelerin interneti kavramı ilk olarak 1999 yılında MIT Auto-ID Center kurucularından Kevin Ashton tarafından Procter & Gamble (P&G) şirketinde tedarik zinciri yönetimini konu aldığı bir sunumda başlık olarak kullanılmıştır [6]. MIT Auto-ID laboratuvar RFID altyapısı üzerinde çalışmakta olduğundan bu tarihle birlikte nesnelerin interneti RFID teknolojisi kapsamında geliştirilmeye devam edilmiştir. Bu tarihten itibaren birçok firma ve araştırma kurumu tarafından bu kavram kullanılmaya devam etmiş ve kullanımına ilişkin öngörülerde bulunmuşlardır[7, 8, 9].

Gartner'a göre, dünyadaki bağlantılı nesnelerin sayısının 2009 ve 2020 arasında otuz kat artışa sahip olduğu ve bu nedenle 2020'de İnternet'e bağlı 26 milyar nesne olacağı öngörülmektedir[7]. Cisco tarafından yapılan bir araştırmaya göre 2020 yılında internete bağlı nesnelerin 50 milyar olacağı öngörülmektedir. Bu artış öngörüsü Şekil 2.2'de gösterilmiştir.



Şekil 2.2 Nesnelerin İnterneti Gelişim Öngörüsü[8]

Nesnelerin internetinin bu kadar büyük hale gelmesinin nedeni kısmen bir şeye bağlıdır: Moore yasası. Moore yasası, bir yonga üzerindeki transistor sayısının yaklaşık iki yılda bir ikiye katlandığını belirtiyor[10]. Bu durum, insanların aynı boyuttaki yonga üzerinde daha güçlü bilgisayarlar geliştirmelerini sağlamaktadır. 1971'de tanınmış bir yarı iletken yonga üreticisi olan Intel, bir işlemci üzerinde 2300 transistor içerirken ve 2012'de mevcut işlemcileri 1,4 milyar transistor içeriyordu [11]. Bu yaklaşık %610 000'lik bir artıştır ve bu eğilimin devam edeceği beklenmektedir.

Nesnelerin internetine yapılan yatırım ve büyüme hızına bakıldığında gelişimi net olarak görülmektedir. İnternet üzerindeki toplam IP trafiği 2012 yılında aylık 43,57 EB (Exabyte) iken, 2014 yılında bu değer 62,47 EB olmuştur. Bu değerlerin artışında daha fazla cihaz kullanılmasını sağlayan Ipv6 teknolojisinin etkisi görülmektedir. Bu alana yapılan yatırımlar ise 2012'den 2014'e ürün geliştiren eleman sayısı 122 binden 300 bine, sermaye kullanımı 738 milyon dolardan 960 milyar dolara çıktığı görülmüştür[12]. Yapılan araştırmalar ile elde edilen veriler ışığında nesnelerin internetinin günümüz ve geleceğin en önemli teknolojileri arasında yer alacağı öngörülmektedir.

2.3 Güvenlik

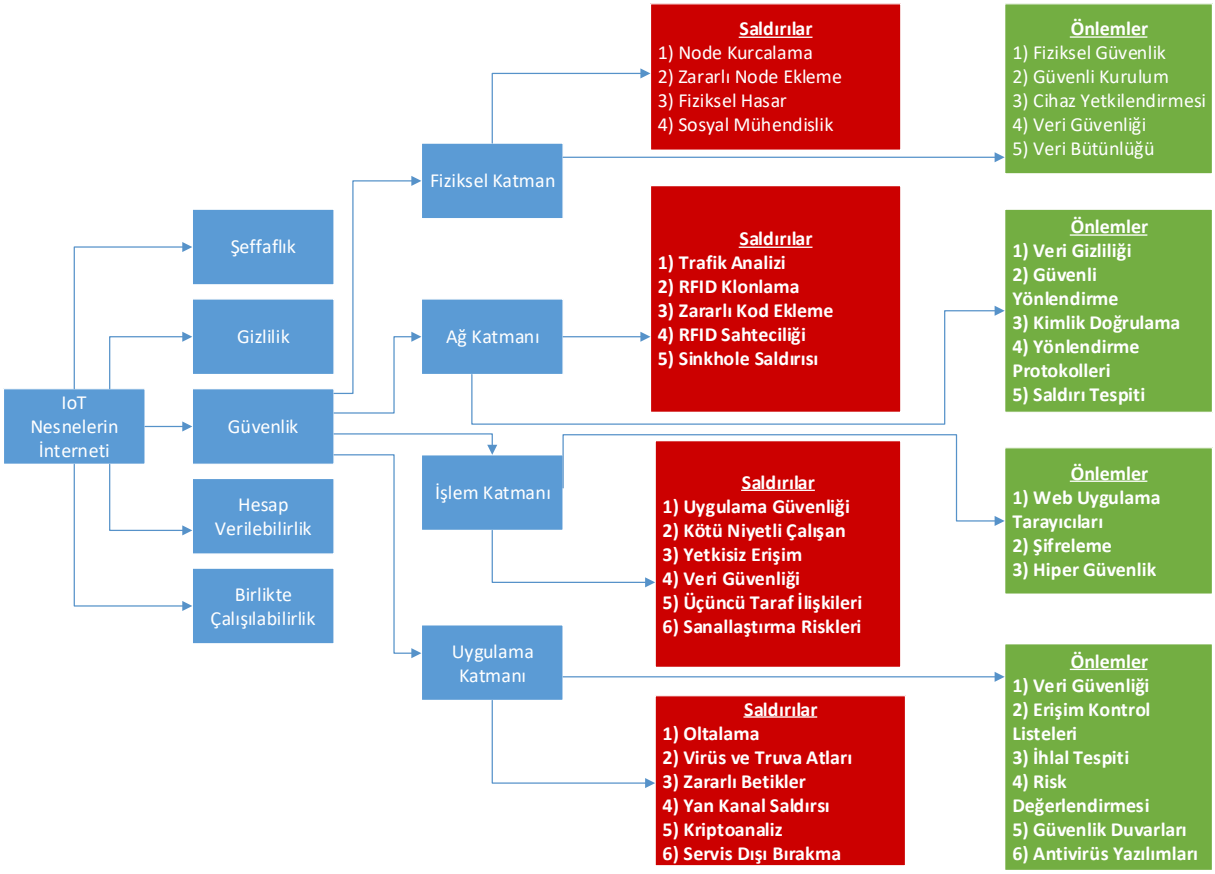
Nesnelerin interneti sistemlerindeki kaynak yetersizliğinden dolayı geleneksel savunma ve tespit sistemlerinin kullanılamaması, onu daha çok tercih edilen bir hedef haline getirmiştir. Bu sistemlerin güvenliğinin sağlanması için tüm bilişim sistemlerinde olduğu gibi bazı güvenlik parametrelerinin uygulanması gerekmektedir. Gizlilik, bütünlük, erişilebilirlik, kimlik doğrulama, yetkilendirme vb. özelliklerin uygulanması güvenlik için önemlidir. Dolayısı ile aşağıda yer alan geleneksel güvenlik parametrelerinin nesnelerin interneti sistemlerinde bulunması beklenmektedir.

- **Güvenilirlik**
  Bir kullanıcı tarafından alınan bilgi, kimliği doğrulanmış elektronik kaynaktan gönderilmiş olsun veya olmasın tanınabilir(kimlik) olmalıdır.
- **Gizlilik**
  Bilgilerin sadece yetkili kullanıcılar tarafından okunmasını ve yetkisiz kullanıcılar tarafından okunmamasını sağlamak.
- **Bütünlük**
  Bilgileri iletirken, veri bütünlüğü bilginin özgünlüğünü sağlamasıdır. Bilgi aktarımının, saldırgan tarafından yeniden yazılmamasına, kopyalanmamasına veya değiştirilmemesine izin verilmemelidir.
- **Mahremiyet**
  Bireysel bir kullanıcının kimliği veya ticari tercihleri gibi bilgilerinin sistem tarafından korunmasıdır.
- **Erişilebilirlik**
  Sistem tarafından sağlanan hizmetler her zaman ulaşılabilir olmalıdır. Güvenliği sağlarken hizmet kesintisinin yaşanmaması esastır.

SANS enstitüsü tarafından yapılan bir araştırmaya göre katılımcıların %48'i nesnelerin internetinde yaşanan güvenlik sorunlarının diğer sistemlerle aynı olduğunu belirtmişlerdir[13].

Birçok farklı cihazın bir araya gelmesiyle çok geniş ölçekli bir nesnelerin interneti altyapısı oluşmuştur[14]. Bundan dolayı gelecekte bu sistemlerin büyük bir tehlike ile

karşı karşıya kalacağı tahmin edilmektedir[15]. Bu sistemlerde servis dışı bırakma, şifre kırma saldırıları, ortadaki adam gibi birçok siber saldırının olacağı öngörülmektedir. Bu saldırıların birçoğu zayıf parola kullanımı, şifrelemenin olmayışı ve diğer eksikliklerden kaynaklanmaktadır. Nesnelerin internetinde yer alan saldırılar bulunduğu katmana göre farklılık gösterebilmektedir. Bu katmanlarda yer alan saldırı vektörleri ve önlemler Şekil 2.3'te gösterilmiştir.



Şekil 2.3 Nesnelerin İnterneti Saldırıları ve Önlemler [15]

2.3.1 Fiziksel Katman

Fiziksel katman fiziksel çevre ile ilgilenir ve gerçek dünyadan elde edilen tüm verileri sensör düğümleri ve diğer fiziksel cihazlar yardımıyla toplar. Bu katman çeşitli fiziksel cihazlar arasındaki iletişimden sorumludur. Bu katmanın amacı, ağa hizmet vermek ve cihazların kimlik doğrulanmasını sağlamaktır. Fiziksel katman, Bluetooth, GPS ve Zigbee gibi bazı saldırılara karşı savunmasız olan sensörlerden meydana gelmektedir. Bu tip saldırılar nesnelerin interneti ağının donanım parçalarına yapılabilmektedir[16]

ve saldırganın hedef sistemler ile aynı ortamda olması gerekmektedir. Fiziksel katman saldırıları Tablo 2.1'de gösterilmiştir.

Saldırı Adı	Etkileri	Önlemler
Node Kurcalama	Fiziksel hasar yolu ile hassas verilere erişim	Fiziksel Güvenlik
RF Arayüzüne Saldırı	Sinyal bozma ile iletişim kesilmesi	Yetkilendirme
Zararlı Node Ekleme	İletim sürecinde kesinti oluşturma	Güvenli Önyükleme
Fiziksel Hasar	Sensörlere fiziksel hasar verilmesi	Risk Değerlendirmesi
Sosyal Mühendislik	Özel bilgilerin sızması	Veri Gizliliği
Uyku Yoksunluğu Saldırısı	Nodeların kapatılması	Kimlik Doğrulama
Etiketlere İzinsiz Erişim	Tüm bilgileri değiştirilmesi veya silinmesi	Kimlik Doğrulama

Tablo 2.1 Fiziksel Katman Saldırıları

- **Düğümün Ele Geçirilmesi**
  Saldırganın fiziksel olarak düğüm noktasını ele geçirmesi ile yapılan bir saldırıdır. Bu saldırıda düğümün bir parçası ya da tamamı kullanılarak hassas bilgiler elde edilebilir ya da değiştirilebilir.
- **Kablosuz Ağda Sinyal Karıştırma**
  Bu saldırı tipi daha çok kablosuz sensör ağlarda yaygın olarak kullanılmaktadır. Bu tür bir saldırgan kablosuz sensör düğümlerinin radyo frekanslarına karıştırma uygular [17] ve daha sonra düğümlerin iletişimini sağlayan sinyalleri engeller.
- **RF Servis Dışı Bırakma**

Servis dışı bırakma (Denial of Service) saldırısı RFID'nin herhangi bir arayüzüne yapılabilir. DoS saldırısı hedefe karıştırma radyo sinyalleri gönderilerek RFID'nin iletişimini durdurmayı amaçlar.

- **Kötücül Düğüm Eklenmesi**

  Bu saldırı türü ortada ki adam saldırısı olarak ta bilinmektedir. Saldırgan gönderici ve alıcı arasına yeni bir düğüm ekleyerek oluşan bütün trafiği dinleyebilir ya da değiştirebilir.

- **Fiziksel Saldırı**

  Saldırgan hedef sisteme fiziksel olarak hasar verebilir. Bu saldırı türü sistemin bulunduğu ortamın güvenlik sistemleri ile ilgilidir.

- **Sosyal Mühendislik**

  Bu saldırıda saldırgan sistem kullanıcılarını aldatarak onlardan hassas bilgileri elde etmeyi amaçlar. Saldırgan hedef sistemle direk iletişime geçtiği için bu saldırı türü fiziksel saldırılar arasında gösterilebilir.

- **Uyku Yoksunluğu Saldırısı**

  Birçok sensör güç ihtiyacını değiştirilebilir piller ile sağlamaktadır. Bu sensörler, pil ömrünün uzatılması için uyku rutinleri gibi bazı işlevleri takip edecek şekilde programlanmıştır. Bu tür bir saldırıda, saldırgan sensör düğümlerini yoğun olarak meşgul eder ve daha fazla pil tüketimine neden olur. Bunun sonucunda hedef sensör çalışamaz hale gelir.

- **Kötücül Kod Eklenmesi**

  Bu saldırıda, saldırgan fiziksel olarak bir düğüme kötü amaçlı bir program ekleyebilir ve bu saldırıyı bir düğüme uygulayarak tüm IoT sistemine erişebilir [18]. Örneğin zararlı yazılım içeren bir USB ile sistemi ele geçirebilir.

- **Yetkisiz Erişim:**

  Yeterli seviyede uygulanmayan yetkilendirme ve kimlik doğrulama politikalarının eksik olmasından dolayı saldırgan hedef sistemde verilere erişebilir ve verileri değiştirebilir.

2.3.2 Ağ Katmanı

Ağ katmanı, farklı fiziksel cihazlar arasındaki iletişimden, ağ yönetiminden ve ayrıca bir IoT sisteminde birçok iletişim protokolü aracılığıyla bilgilerin muhafazasından sorumludur. Günümüzde IoT sistemlerde çoğunlukla MQTT ve CoAP (Constrained Application Protocol) protokolleri kullanılmaktadır. Kablosuz sensörler aracılığı ile elde edilen sensör bilgileri ağ katmanı vasıtası ile bilgilerin işleneceği merkeze iletilir. IoT sistemde yer alan her cihaz elde ettiği özel bilgileri kablosuz sensörler yardımı ile gönderir[19]. Bu nedenle sistemde verilerin güvenli ve güvenilir bir şekilde aktarılma görevi bu katman tarafından yerine getirilir. Ağ katmanında yer alan saldırılar Tablo 2.2'de gösterilmiştir.

Saldırı Adı	Etkileri	Önlemler
Sinkhole Saldırısı	Veri sızıntısı	Güvenli yönlendirme
Trafik Analizi	Ağ ile ilgili gizli bilgi sızıntısı	Yönlendirme güvenliği
RFID Klonlama	RFID benzetimi ile verilerin çalınması	Kimlik doğrulama
RFID Sahteciliği	Veri iletişiminin kontrol edilmesi	GPS sistemi tekniği
Hello Flood Saldırısı	Trafik sıkışıklığı ve kanal tıkanması	Hello flood önleme
Yönlendirme Saldırısı	Yönlendirme döngüleri ile ağın imhası	Yönlendirme tablolarının şifrelenmesi

Tablo 2.2 Ağ Katmanı Saldırıları

- **Trafik Analizi Saldırısı**

Bu saldırıda, saldırgan kablosuz özelliği nedeniyle RFID teknolojisinden gelen gizli bilgilere ve diğer yararlı verilere erişebilir. Saldırıya başlamadan önce saldırganın hedef ağ hakkında bilgi toplaması gerekmektedir[20]. Bu işlem, port tarama uygulamaları,

paket dinleyici uygulamaları vb. bazı dinleme (sniffing) işlemleri kullanılarak gerçekleştirilmektedir.

- **RFID Klonlama**

Bu tür bir saldırıda, saldırgan RFID'yi taklit ederek ve geçerli RFID'den başka bir RFID etiketine veri kopyalayarak faydalı verilere erişebilir [21].

- **Kötücül Kod Enjeksiyonu**

Bu saldırı IoT ağında ciddi bir etkiye sebebiyet verebilir ve hatta ağı kesintiye uğratabilir. Bu saldırıda saldırgan hedef sistemde yeni bir düğüm noktası oluşturur ve zararlı kodu bu düğüme enjekte eder[22].

- **RFID Sahteciliği**

Bu saldırıda, saldırgan RFID sinyallerini taklit ederek verilerin iletimini yakalar. Daha sonra gerçek bir veri gibi kendi verilerini elde ettiği orijinal RFID[23] etiketi ile iletir. Saldırgan sonuç olarak gerçek bir kimliği kullanarak hedef sisteme dâhil olur.

- **RFID Yetkisiz Erişim**

RFID sistemlerinde, etiketlere erişiminin sağlanması herkes için çok kolaydır, çünkü çoğunlukla RFID sisteminde yerleşik politika veya herhangi bir kimlik doğrulama sistemi yoktur[24]. Böylece, saldırgan hedef sistemde yer alan sensör düğümlerinin bilgilerini değiştirebilir, okuyabilir veya silebilir.

- **Sinkhole Saldırısı**

Saldırgan bir sinkhole oluşturur ve kablosuz sensör ağının düğümlerinden gelen tüm trafiği üzerine alır. Bu saldırıda trafik gerçek hedefe iletilmez ve iletişim durdurulur. Böylece verilerin mahremiyetine ve gizliliğine zarar verilir.

- **Ortadaki Adam Saldırısı**

Saldırganın iletişim kuran iki cihaz arasına gizlice girerek iletişimi ele geçirip değiştirdiği bir saldırıdır. Fiziksel saldırı türlerinden farklı olarak, saldırganın fiziksel olarak hedefe yakın olması gerekmemektedir, fakat saldırgan ağ katmanında bir düğümün diğeri ile olan ağ protokolünün iletişimi üzerinde yoğunlaşmalıdır.

- **Servis Dışı Bırakma**

Bu saldırıda saldırgan hedef sisteme çok sayıda paket göndererek sistemi hizmet veremez duruma getirebilir. Sistemin kaynakları kullanılamaz hale geldiği için vermesi gereken hizmetleri yerine getiremez.

- **Yönlendirme Saldırısı**

Bu saldırı türünde saldırgan, yönlendirme hakkındaki bilgileri taklit eder ve değiştirir. Hedef sisteme çok sayıda hatalı ve yanlış paketler gönderilerek oluşturulan karışıklıktan dolayı gerçek paketlerde asıl hedeflerine ulaşamazlar.

2.3.3 İşlem Katmanı

İşlem katmanı ağ ve fiziksel katmanlarının birlikte çalışmasından sorumludur. IoT sistemlerde yer alan çok büyük miktardaki verinin veri tabanı ile ilişkilendirilmesi ve işlenmesi çok önemlidir. İşlem katmanı, bilgiyi değerlendirme ve verileri işleme süreçlerini akıllı hesaplama temelinde otomatik olarak yapabilir. Bu nedenle, bu katmandaki teknolojilerin yeniliği, IoT sisteminin gelişimi için yararlı olacaktır. İşleme katmanı, veri depolama ve veri işleme gibi farklı teknolojilerden meydana gelir. Verilerin işlendiği ya da depolandığı bulut sistemlerine yapılan saldırılar IoT için en önemli saldırıdır. Bu saldırılar ile ağ katmanı savunmasız hale getirilebilir. İşlem katmanında yer alan saldırılar Tablo 2.3'de gösterilmiştir.

Saldırı Adı	Etkileri	Önlemler
Sanallaştırma Tehditleri	Kaynaklara zarar verilmesi	Hiper güvenlik
Paylaşılan Kaynaklar	Yetkisiz kişilerin kaynaklara erişimi	Homomorfik şifreleme
Uygulama Güvenliği	Veri çalınması	Web uygulama tarayıcı kullanımı
Veri Güvenliği	Verinin bulutta olmasından dolayı gizli bilgilerin ifşası	Yedekli yapı
Temel Altyapı Güvenliği	Alt tabakanın korunmasız olması	Yedekli yapı
Üçüncü Taraf İlişkileri	Veri sızıntısı	Şifreleme

Tablo 2.3 İşlem Katmanı Saldırıları

- **Yetkisiz Erişim**

  İşlem katmanı, veri depolama ve işleme gibi kritik görevleri yerine getirir[25]. Bu saldırıda saldırgan yetersiz veya hatalı yapılandırılmış yetkilendirmeden dolayı hedef sisteme erişerek verileri değiştirebilir ya da silebilir.
- **Kötü Niyetli Çalışan**

  Yetkili bir kimse tarafından sisteme yapılan saldırıdır. Burada saldırgan kendi amaçları için verileri değiştirerek ya da silerek saldırıyı gerçekleştirebilir[26].
- **Uygulama Güvenliği**

  Verilerin işlenmesi ve depolanması süreçlerinde yaşanan eksikliklerden kaynaklanan tehditlerdir. Yazılımların hizmet olarak tedarik edilmesi ile kullanılan dış kaynak hizmetlerinden kaynaklanan zafiyetlerin sebebiyet verdiği saldırılardır. Özellikle web yönetim platformlarında ve bulut bilişim hizmetlerinde yer alan açıklıklar bu saldırıların oluşmasına sebebiyet vermektedir.
- **Veri Güvenliği**

  Kullanıcı verilerinin güvenliğinin sağlanması hizmet sağlayıcı için büyük bir sorumluluktur. Verilerin servis sağlayıcı tarafından yedeklenmesi birçok güvenlik problemine neden olabilmektedir.
- **Altyapı Güvenliği**

  Altyapı olarak hizmet alan sistemler de, geliştiriciler alt katmanlara erişemez ve bu katmanın güvenliği servis sağlayıcıların sorumluluğundadır. IoT sistem yöneticileri bu hizmet kullanımında servis sağlayıcısına güvenmek zorundadır. Fakat uygulamada altyapı kökenli birçok güvenlik problemi meydana gelebilir.
- **Sanallaştırma Güvenliği**

  Sanal makinelerin güvenliği diğer makineler kadar önemlidir ve makineye herhangi bir zarar gelmesi tüm sistemi etkileyebilir. Bu katmanda sanallaştırma teknolojileri birçok saldırıya karşı güvensiz olabilir.
- **Kaynak Paylaşımı**

  Sanal makinelerde aynı kaynağın kullanılması IoT ağı için birçok zafiyete yol açabilir. Paylaşılan kaynakların yetkilendirme politikaları olmadan paylaşılması veri hırsızlığına sebep olabilir.

2.3.4 Uygulama Katmanı

Uygulama katmanı, ağa bağlı cihazlar arasında son kullanıcılara yönelik yaygın bir şekilde kullanıcı etkileşimli hizmetler sağlayan hizmet odaklı katmandır. İşlem katmanında yer alan işlenmiş bilgiye erişmek için, kullanıcı ihtiyaçlarını ulaşım, iletişim ve akıllı evler gibi çeşitli şekillerde kolaylaştıran IoT uygulamalarına bir erişim ara yüzü sağlar.

Uygulama saldırıları, IoT sistem güvenliği içerisinde önemli bir yere sahiptir. Uygulama saldırıları, zararlı virüsler ve truva atı, solucanlar, casus yazılımlar vb. saldırı vektörleri kullanılarak gizli verilere zarar verilebilir, veriler değiştirilebilir, IoT cihazlarına zarar verilebilir ve faydalı bilgilere erişilebilir. Bu katmanda yer alan saldırılar Tablo 2.4'de gösterilmiştir.

Saldırı Adı	Etkileri	Önlemler
Sanallaştırma Tehditleri	Kaynaklara zarar verilmesi	Hiper güvenlik
Paylaşılan Kaynaklar	Yetkisiz kişilerin kaynaklara erişimi	Homomorfik şifreleme
Uygulama Güvenliği	Veri çalınması	Web uygulama tarayıcı kullanımı
Veri Güvenliği	Verinin bulutta olmasından dolayı gizli bilgilerin ifşası	Yedekli yapı
Temel Altyapı Güvenliği	Alt tabakanın korunmasız olması	Yedekli yapı
Üçüncü Taraf İlişkileri	Veri sızıntısı	Şifreleme

Tablo 2.4 Uygulama Katmanı Saldırıları

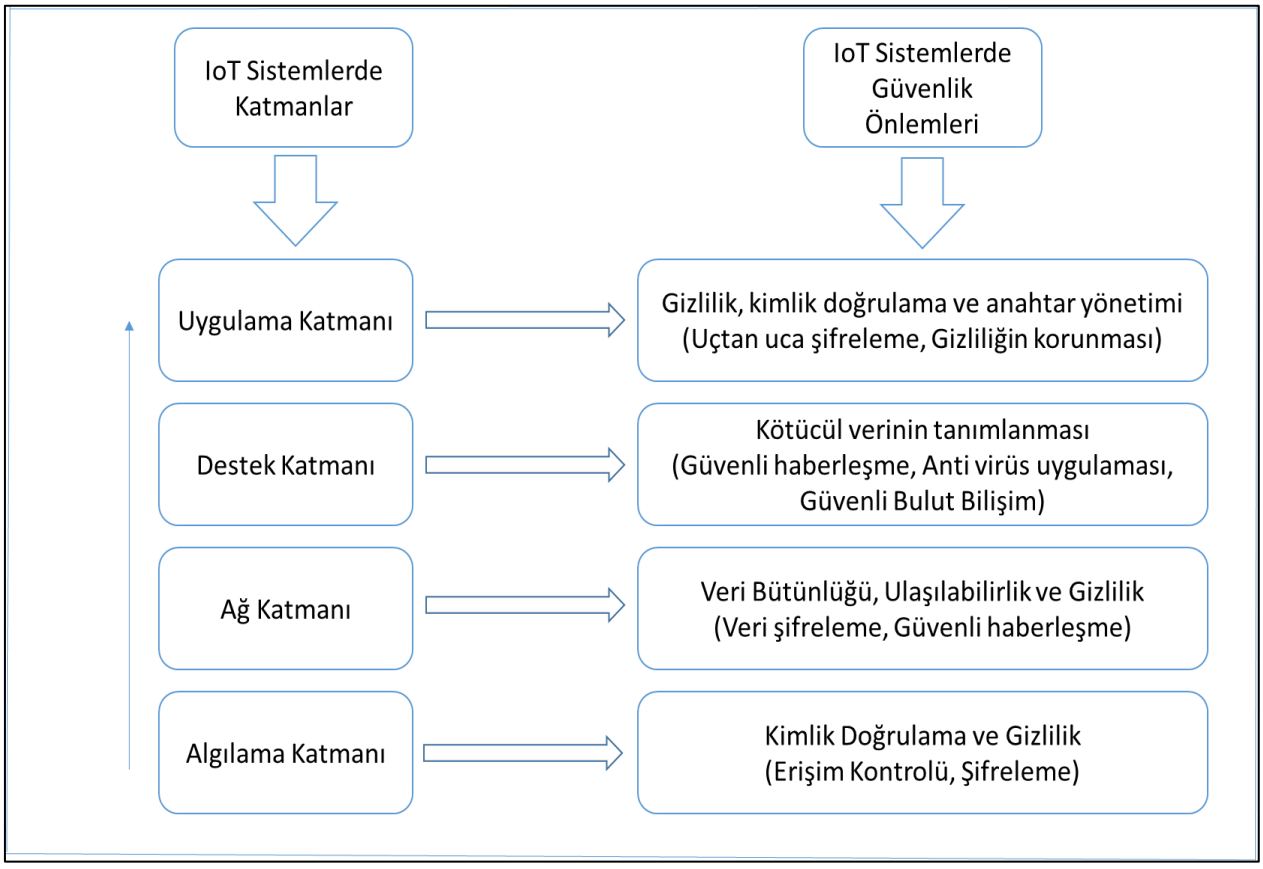
- **Virüsler, Solucanlar, Truva Atı ve Casus Yazılımlar:** Saldırgan, sistemlere zararlı yazılımları enjekte ederek IoT sistemini etkileyebilir. Bu tür saldırılarda

kullanılan casus yazılımlar vasıtası ile hedef sistemden kritik bilgiler çalınabilir, sistemlere zarar verilebilir ve daha birçok atak vektörü harekete geçirilebilir.

- **Zararlı Betikler:** IoT sistemlerin kontrol edilmesi ve izlenmesi için oluşturulan sistemlerde kullanılan tarayıcılar üzerinde çalıştırabilen bu zararlı betikler ile sistem ele geçirilebilir.
- **Servis Dışı Bırakma:** Bu katmanda ağ katmanın da yer alan saldırılılara ek olarak kullanılan uygulamada tespit edilen kodlama zafiyetleri ile hedef sistem hizmet veremez hale getirilebilir.

2.4 Güvenlik Önlemleri

Nesnelerin interneti sistemlerinde bulunan güvenlik sorunlarına yönelik bazı önlemler geliştirilmiştir. Bunlar; erişim kontrolü, veri şifreleme, bulut bilişim ve sertifikasyon olarak aşağıda anlatılmıştır. Şekil 2.4'te her bir katman için güvenlik önlemleri gösterilmiştir.



Şekil 2.4 Nesnelerin İnterneti Güvenlik Önlemleri

2.4.1 Veri Şifreleme

Şifreleme, açık metnin şifreli metin olarak bilinen anlaşılamaz bir formata dönüştürme işlemidir. Bilginin bütünlüğünün ve gizliliğini korumak için kullanılır. Veriler bir saldırgan tarafından ele geçirildiğinde, şifreleme bu verilerin açığa çıkmasını önler. İki tür şifreleme yöntemi vardır; cihazdan cihaza (hop by hop) ve uçtan uca (peer to peer) şifreleme.

Cihazdan cihaza şifrelemede bilgi iletişim aşamasında şifrelenerek iletilir. Her bir cihazda verinin şifresi çözülür ve tekrar şifrelenerek bir sonraki cihaza gönderilir. IoT sistemlerde bu yöntem ağ katmanı için uygun olan şifreleme yöntemidir. Uçtan uca şifreleme yönteminde sadece alıcı ve gönderici noktalarında veri şifrelenir ya da şifresi çözülür. Sistem ihtiyaçlarına göre bu şifreleme yöntemlerinden herhangi biri uygulanabilir. Güvenli bir anahtar değişimi ve anahtar yönetim sistemi kullanmak dinleme, sahtecilik ve veri ifşası gibi çok sayıda saldırıyı önleyebilir[27].

2.4.2 Açık Anahtar Sertifikaları

Sertifika, birbiriyle iletişim kuran tarafların gerçek kimliğini doğrulamanın güvenli bir yoludur. Açık anahtar altyapısını (PKI) kullanarak, bir IoT sisteminin güvenilirliğini ve gizliliğini korumak için açık anahtar sertifikası ile kimlik doğrulaması yapılabilir. Aynı zamanda bilgi aktarımına dâhil olan tarafların kimliğini tespit etmenin de güvenli bir yoludur.

2.4.3 Erişim Kontrolü

Erişim kontrolü, cihazlara, nesnelere veya bir nesnelerin interneti sisteminin kaynaklarına erişmek için illegal olan bir kişiye erişimi sınırlandırarak güvenli bir IoT ortamı sağlamak için kullanılır. Sertifikasyon ve erişim kontrol mekanizması birbiriyle ilişkilidir. Doğru erişim kontrolü politikasını uygulamak için, sistemde doğru kimlik doğrulama yönteminin kullanılması önemlidir.

2.4.4 Bulut Bilişim

Bulut bilişim, büyük veri depolama kapasitesi, düşük maliyetle yüksek performans elde etmek için kullanılan bir teknolojidir. IoT sistemlerde çok büyük miktarda verinin toplanması ve analiz edilmesinden dolayı bulut bilişim ile birlikte kullanılması çok verimli olacaktır. Bulut bilişimin başka bir kullanımı da üçüncü taraf güvenliğini sağlamaktır. Bulut bilişimde kullandığın kadar öde prensibinin uygulanmasından dolayı nesnelerin interneti sistemlerde kullanımı düşük maliyetle sağlanabilir.

2.5 Protokoller

Nesnelerin interneti sistemlerde kullanılan sistemler bulunduğu katmana göre farklılıklar göstermektedir. Başlıca kullanılan protokoller Şekil 2.5'te gösterilmiştir.

<table>
<tr><td colspan="2">Oturum</td><td>MQTT, SMQTT, CoRE, DDS, AMQP, XMPP, CoAP, …</td><td rowspan="4">Güvenlik
TCG, Oath 2.0, SMACK, SASL, ISASecure, ace, DTLS, Dice, …</td><td rowspan="4">Yönetim
IEEE 1905, IEEE 1451, …</td></tr>
<tr><td rowspan="2">Ağ</td><td>Kapsülleme</td><td>6LowPAN, 6TiSCH, 6Lo, Thread, …</td></tr>
<tr><td>Yönlendirme</td><td>RPL, CORPL, CARP, …</td></tr>
<tr><td colspan="2">Veri Bağı</td><td>WiFi, Bluetooth Low Energy, Z-Wave, ZegBee Smart, DECT/ULE, 3G/LTE, NFC, Weightless, HomePlug GP, 802.11ah, 805.15.4e, G.9959, WirelessHART, DASH7, ANT+, LTE-A, LoRaWAN, …</td></tr>
</table>

Şekil 2.5 Nesnelerin İnterneti Protokolleri[28]

2.5.1 Veri Bağı Katmanı Protokolleri

2.5.1.2 IEEE 802.15.4e

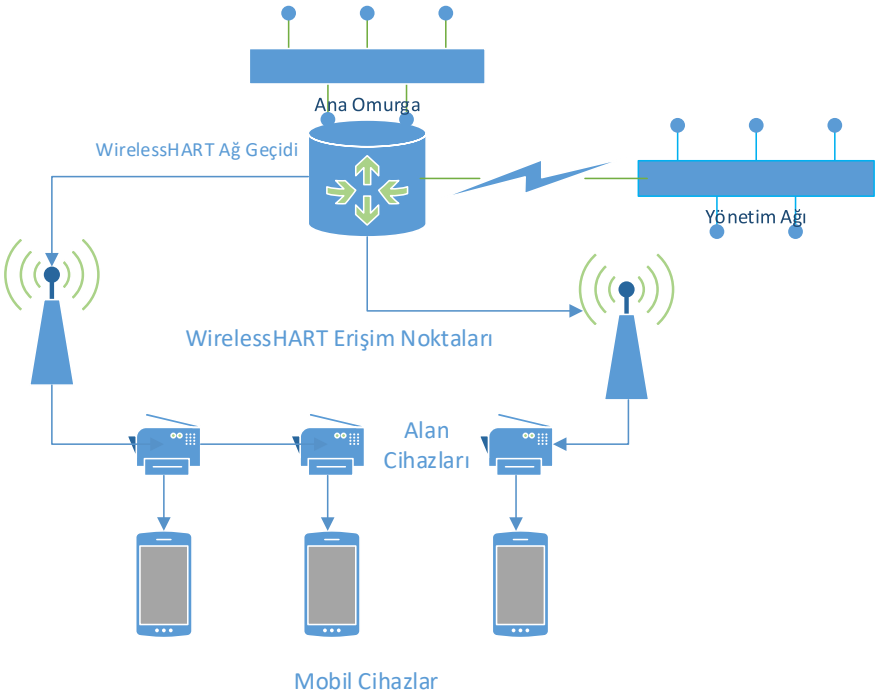
IEEE 802.15.4, MAC katmanında yaygın olarak kullanılan bir veri bağı standardıdır. Bu standart, çerçeve biçimini, üstbilgileri, hedef ve kaynak adreslerini belirtir ve düğümler arasındaki iletişimin nasıl gerçekleşebileceğini tanımlar. Ağda kullanılan geleneksel çerçeve formatları, güç kısıtlı IoT cihazları için uygun değildir. 2008 yılında IEEE 802.15.4e standardı, IEEE 802.15.4'ü genişletmek ve düşük güç iletişimini desteklemek için oluşturuldu. IoT veri bağlantılarında yüksek güvenilirlik ve düşük maliyetli iletişim sağlamak için zaman eşleme ve kanal atlamayı kullanır[29].

2.5.1.3 IEEE 802.11ah

IEEE 802.11ah, IEEE 802.11 standartlarının IoT ihtiyaçlarını karşılamak için düşük yük tüketimine sahip olan versiyonudur. IEEE 802.11 standartları (Wi-Fi), geleneksel ağda en çok kullanılan kablosuz standartlardır. Dizüstü bilgisayarlar, cep telefonları, tabletler ve dijital TV'ler de dâhil olmak üzere birçok dijital cihazlar için yaygın olarak kullanılmaktadırlar. Fakat orijinal Wi-Fi standartları, çerçeve yükleri ve yüksek güç tüketimi nedeniyle IoT uygulamaları için uygun değildirler. Bu nedenle, IEEE 802.11 çalışma grubu, sensörlere ve nesnelere uygun düşük bant genişliği, düşük güç tüketimi ve iletişimi destekleyen bir standart geliştirmek için 802.11ah görev grubu çalışmasını başlatmıştır[30].

2.5.1.4 WirelessHART

WirelessHART, IEEE 802.15.4 PHY'nin üzerinde çalışan bir MAC katman standardıdır ve zaman bölmeli çoklu erişim (TDMA) yöntemini kullanır. Mesajları şifrelemek ve bütünlüğü kontrol etmek için gelişmiş şifreleme standardı algoritmasını (AES) kullanır. Bu nedenle, diğerlerinden daha güvenli ve güvenilirdir. Şekil 2.6'da görüldüğü gibi, mimarisi, bir ağ yöneticisi, bir güvenlik yöneticisi, kablosuz ağı kablolu ağlara bağlamak için bir ağ geçidinden, saha cihazları, erişim noktaları, yönlendiriciler ve adaptörler gibi kablosuz aygıtlardan oluşmaktadır[30].



Şekil 2.6 WirelessHART Mimarisi [30]

2.5.1.5 Z-Wave

Z-Wave, ev otomasyonu için tasarlanmış, ancak son zamanlarda akıllı evler ve küçük ticari alanlar dâhil olmak üzere birçok IoT uygulamasında kullanılan düşük güç tüketen bir MAC standardıdır. 30 metreye kadar olan mesafeler, noktadan noktaya iletişim ve küçük mesajlar için uygundur. Güvenilir iletim için medya erişimi için CSMA/CA kullanır. Yöneticinin istemcileri kontrol ettiği, onlara komutlar gönderdiği ve tüm ağın zamanlamasını ele aldığı bir mimariyi izler [31].

2.5.1.6 Bluetooth Smart

IoT'de yaygın olarak kullanılan veri bağı katmanı için kısa mesafeli iletişim standardıdır. Çoğunlukla araç içi ağlarda kullanılır. Orijinal Bluetooth standartlarından 15 kat daha düşük bir gecikme süresi vardır. Enerji tüketimi orijinal Bluetooth'dan 10 kat kadar daha azdır[32].

2.5.1.7 ZigBee

ZigBee, akıllı evlerde, uzaktan kumandalarda ve sağlık sistemlerinde orta mesafeli iletişim için kullanılan IoT'deki en yaygın standartlardan biridir. Ağ topolojileri yıldız,

eşler arası (peer-to-peer) veya küme ağacını (cluster-tree) içerir. Merkezi bir kontrol mekanizması ile ağ kontrol edilir. ZigBee standardının iki türü vardır: ZigBee ve ZigBee Pro. ZigBee Pro, simetrik anahtar değişimi, ölçeklenebilirlik ve verimli bire-bir yönlendirme mekanizmaları kullanarak daha iyi performans ve daha fazla özellik sunuyor [33].

2.5.1.8 DASH7

DASH7, RFID cihazları için kullanılan ve dünya çapında mevcut olan endüstriyel bilimsel tıbbi (ISM) bantta çalışan yeni bir kablosuz iletişim protokolüdür. Temel olarak, ZigBee'ye kıyasla daha yüksek veri hızına sahip, ölçeklendirilebilir, uzun menzilli dış mekân alanı için tasarlanmıştır. Şifreleme ve Ipv6 adreslemeyi destekleyen düşük maliyetli bir çözüm sunmaktadır[34].

2.5.1.9 HomePlug

HomePlug Green PHY (HomePlug GP), HomePlug Powerline Alliance tarafından geliştirilen ve çoğunlukla ev otomasyon uygulamalarında kullanılan bir veri bağı protokolüdür. HomePlug-AV, HomePlug-AV2 dâhil olmak üzere HomePlug paketi, ağ yığınının hem PHY hem de MAC katmanlarını kapsar. HomePlugGP, akıllı ev ve akıllı şebeke uygulamaları gibi IoT uygulamaları için tasarlanmıştır. Temel olarak HomePlug-AV'nin birlikte çalışabilirliğini, güvenilirliğini ve kapsamını korurken maliyetini ve güç tüketimini azaltmak için tasarlanmıştır. HomePlug GP, MACP tekniği olarak sadece CSMA'yı kullanırken HomePlug GP hem CSMA hem de TDMA kullanıyor. Ayrıca HomePlugGP, uyku zamanlarını senkronize ederek ve gerektiğinde uyanarak düğümlerin uyumasına izin veren bir güç tasarrufu moduna sahiptir[35].

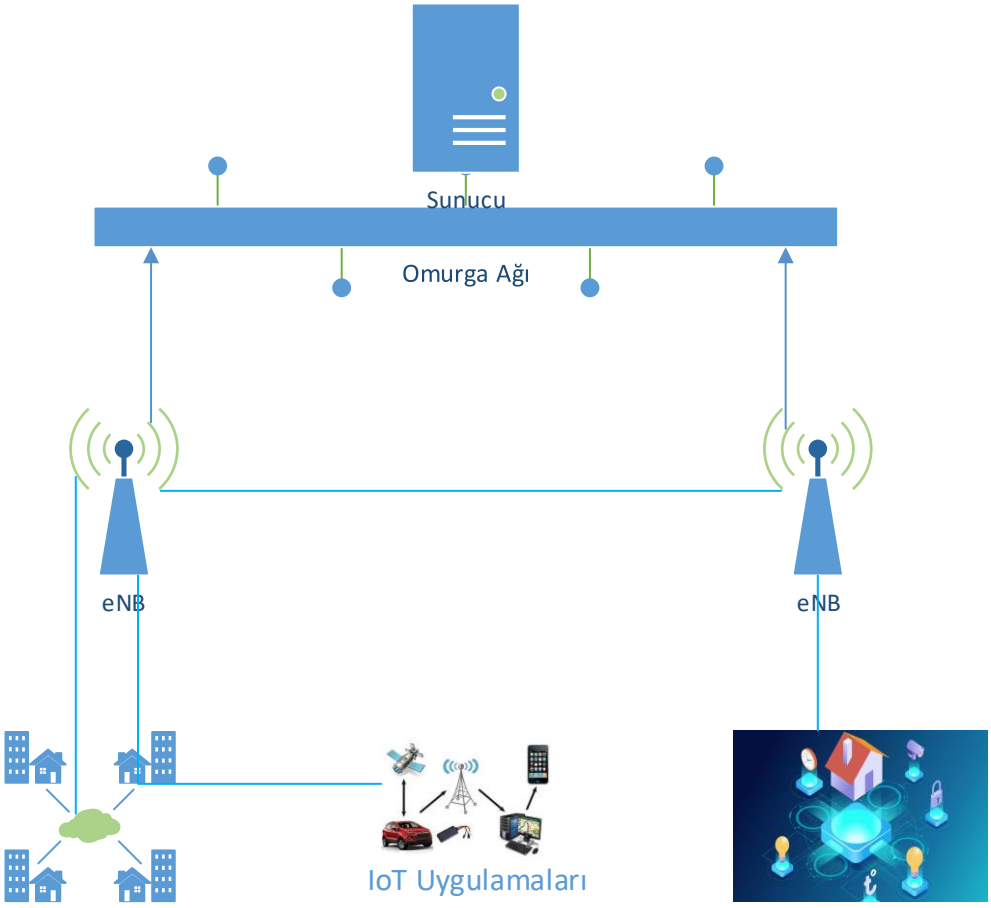
2.5.1.10 G.9959

Düşük bant genişliği, düşük maliyetli ve yarım çift yönlü güvenilir kablosuz iletişim için tasarlanmış bir ITU MAC katman standardıdır. Zamanın kritik olduğu, güvenilirliğin önemli olduğu ve düşük güç tüketiminin gerekli olduğu gecikme duyarlı uygulamalar için tasarlanmıştır. G9959 protokolü, benzersiz kanal erişimi, çerçeve doğrulama ve yeniden iletim özelliklerini içerir[36].

2.5.1.11 LTE-A

LTE-A, M2M ve IoT gereksinimlerini karşılamak için tasarlanmış bir hücresel ağ standartları topluluğudur. Diğer hücresel protokollere kıyasla en ölçeklenebilir ve uygun

maliyetli standartlardan biridir. 2009 yılında mobil teknolojiler için geliştirilmeye başlanmıştır. Geleneksel olarak, frekansın birden fazla alt taşıyıcıya bölünmüş olduğu bir orta erişim teknolojisi olarak dikey frekans bölmeli çoklu erişim (OFDMA) kullanır. LTE-A'nın mimarisi bir çekirdek ağ (CN), bir radyo erişim ağı (RAN) ve mobil düğümlerden oluşur. Çekirdek ağ (CN), mobil cihazları kontrol etmek ve IP'lerini takip etmekten sorumludur. Radyo erişim ağı (RAN), kontrol ve veri düzlemlerini kurmak ve kablosuz bağlantı ve radyo erişim kontrolünü ele almaktan sorumludur. Radyo erişim ağı ve çekirdek ağ, Şekil 2.7'de gösterildiği gibi S1 bağlantısını kullanarak iletişim kurar, burada RAN, diğer mobil düğümlerin kablosuz olarak bağlandığı eNB'lerden oluşur [37].



Şekil 2.7 LTE-A Mimarisi [37]

2.5.1.12 LoRaWAN (long-range wide-area network)

LoRaWAN, güç tasarrufu, düşük maliyet, mobilite, güvenlik ve çift yönlü iletişim gereksinimleriyle IoT uygulamaları için tasarlanmış uzun menzilli geniş alanlı bir kablosuz ağ teknolojisidir. Milyonlarca cihaza sahip, ölçeklenebilir kablosuz ağlar için tasarlanmış düşük güç tüketimi için optimize edilmiş bir protokoldür. Hareketlilik ve kullanım kolaylığı özellikleri sağlarken, IoT'nin gelecekteki ihtiyaçlarını desteklemek için yedeklilik, ortam bağımsız, düşük maliyetli, düşük güç tüketimi teknolojilerini destekler[38].

2.5.1.12 DECT/ULE

DECT (dijital gelişmiş kablosuz telekomünikasyon), kablosuz telefonlar için tasarlanmış evrensel bir Avrupa standardıdır. IoT uygulamaları için kullanılabilecek düşük güç ve düşük maliyetli bir kablosuz ara yüzü teknolojisini belirten DECT/ULE (ultra düşük enerji) adlı yeni bir sürüm piyasaya sürüldü. DECT/ULE, orijinal DECT protokolünde desteklenmeyen FDMA, TDMA ve zaman bölmeli çoklamayı destekler[39].

2.5.1.13 EnOcean

EnOcean, öncelikli olarak otomasyon için geliştirilen, daha sonra diğer IoT uygulamaları için de uygulanabilen, enerji tasarrufu sağlayan bir kablosuz teknolojidir. Temel amaç, hareketin verimli bir şekilde toplanmasını veya herhangi bir çevre enerjisinin kullanılması ve dönüştürücüler kullanılarak kullanılabilir enerjiye dönüştürülmesidir. Bu protokolün düşük bir paket boyutu vardır ve çoğunlukla ısıtma, havalandırma ve klima IoT uygulamalarında kullanılır[40].

2.5.2 Veri Aktarım Protokolleri

2.5.2.1 MQTT

MQTT (Message Queuing Telemetry Transport) protokolü 4. Bölümde detaylı olarak ele alınmıştır.

2.5.2.2 SMQTT

MQTT protokolünün güvenli olan bir modelidir. Hafif nitelikte şifreleme sağlamak için tasarlanmıştır. Bu tür bir şifreleme, bir mesajın şifrelenmiş ve çok sayıda başka düğümlere iletildiği ve IoT uygulamalarında oldukça yaygın olan çok noktaya yayın özelliğini kullanır. Genellikle, algoritma dört ana aşamadan oluşur: kurulum, şifreleme,

yayınlama ve şifre çözme. Kurulum aşamasında, aboneler ve yayıncılar kendilerini geliştiriciye kaydeder ve geliştiricilerin anahtar oluşturma algoritması seçimine göre bir ana gizli anahtar alırlar. Daha sonra, veriler yayınlandığında, abonelere gönderen aracı tarafından şifrelenir ve yayınlanır. Son olarak, aynı ana gizli anahtara sahip olan abonelerde şifresi çözülür. Anahtar oluşturma ve şifreleme algoritmaları standartlaştırılmamıştır.

2.5.2.3 AMQP

Gelişmiş ileti kuyruklandırma protokolü (AMQP), finans endüstrisi için tasarlanmış, TCP üzerinden çalışan ve MQTT'ye benzer yayın / abone mimarisini kullanan başka bir OASIS standardıdır. Bu standartlardaki temel fark, broker'ın iki ana bileşene ayrılmasıdır: değişim ve kuyruklar. Değişim bileşeni, yayıncı mesajlarının alınmasından ve önceden belirlenmiş rollerin ardından kuyruklara dağıtılmasından sorumludur. Aboneler temel olarak konuları temsil eden ve mevcut olduğunda sensörsel verileri alan kuyruklara bağlanır[41].

2.5.2.4 CoAP

Kısıtlı Uygulama Protokolü (CoAP) kısıtlı düğümlerle ve Nesnelerin İnternet'indeki kısıtlı ağlarda kullanım için geliştirilen özel bir web transfer protokolüdür. Protokol, akıllı enerji ve bina otomasyonu gibi makineden makineye (M2M) uygulamalar için tasarlanmıştır. UDP protokolü üzerinde geliştirilen bu protokol güvenirliği sağlamak için hafif bir mekanizma kullanmaktadır. CoAP mimarisi iki ana alt gruba ayrılır: mesajlaşma ve istek-yanıt. Mesajlaşma alt katmanı, mesajların güvenilirliğinden ve kopyalanmasından sorumludur ve istek-yanıt alt katmanı iletişimden sorumludur. CoAP dört mesajlaşma türüne sahiptir: onaylanabilir, onaylanamayan, geri alma ve ayrı[42].

2.5.2.5 XMPP

Bu protokol orijinal olarak sohbet ve mesaj alışverişi uygulamaları için tasarlanmıştır. Protokol XML dilini kullanır ve on yıldan fazla bir süre önce IETF tarafından standartlaştırılmıştır. İnternet üzerinden kullanıldığında oldukça verimlidir. Son zamanlarda, kullanımı kolay genişletilebilir hale gelen XML'in standart kullanımı nedeniyle IoT ve SDN uygulamaları için kullanım alanları genişletilmiştir. XMPP, hem yayınlama/abone olma hem de istek/yanıt mimarisini destekler ve hangi mimarinin

kullanılacağını seçmek geliştiriciye bağlıdır. Yakın zamanlı gerçek zamanlı uygulamalar için tasarlanmıştır ve böylece düşük gecikmeli küçük mesajları etkili bir şekilde destekler. Herhangi bir servis garantisi sağlamamaktadır ve dolayısıyla M2M iletişimleri için kullanışlı değildir. Dahası, XML mesajları, IoT uygulaması için kritik olan güç tüketimini artıran çok sayıda başlık ve etiket biçimi nedeniyle ek yük oluşturur. Bu nedenle, XMPP IoT'de nadiren kullanılmaktadır. Fakat IoT uygulamalarını desteklemek amacıyla mimari güncelleme çalışmaları devam etmektedir[43].

2.5.2.6 DDS

Veri dağıtım hizmeti (DDS), Object Management Group (OMG) tarafından tasarlanan mesajlaşma standardıdır. Yayınlama/abone mimarisi kullanır ve çoğunlukla M2M iletişimleri için kullanılmaktadır. Bu protokolün en faydalı özelliği, IoT ve M2M iletişimine uygun ve daha merkezi bir mimarinin kullanımı ile hizmet seviyelerinin ve güvenilirliğinin kaliteli olmasıdır. Güvenlik, ivedilik, öncelik, dayanıklılık, güvenilirlik, vb. olmak üzere çeşitli kalite kıstasları sunmasını sağlayan 23 kalite seviyesine sahiptir. İki alt katmanı vardır: veri merkezli yayınlama-abone ve veri-yerel yeniden yapılandırma alt katmanları. Birincisi, aboneye mesaj tesliminin sorumluluğunu alırken, ikincisi tercihe bağlıdır ve uygulama katmanında DDS'nin bütünleşmesine izin verir. Yayım katmanı, sensör verilerinin dağıtımından sorumludur. Veri yazarı, abonelere gönderilecek olan veriler ve değişiklikler hakkında hemfikir olmak için yayıncılarla etkileşim halindedir. Aboneler, IoT uygulamasına teslim edilecek sensör verilerinin alıcılarıdır. Veri okuyucular temel olarak yayınlanan verileri okur ve bunları abonelere iletir. Konular temel olarak yayınlanmakta olan veri türleridir.

3.TUZAK SİSTEMLER

3.1 Giriş

Tuzak sistem (Balküpü), saldırganları cezbetmek ve yakalamak için bir tuzak olarak tasarlanan sistemdir[44]. Genel olarak tuzak sistemlerde kasıtlı olarak açıklıklar barındırılır ve bu sayede saldırganların ilgisini çekebilir. Tuzak sistem içerisine saldırganın kullanarak farklı atakları deneyeceği bilgiler bırakılarak onun izlenmesi amaçlanır. Sistem içerisinde yer alan mekanizmalar sayesinde saldırganın bütün

hareketleri kayıt altına alınır. Saldırganın kayıt altına alınan aktivite kayıtları istihbarat analizi için kullanılabilir.

3.2 Tarihçe

Balküpü kavramı 1990 öncesine dayanmasına rağmen, kullanım alanı yaygın değildi. Bilinen en eski sistemlerden biri 1997 yılında geliştirilen DTK (Deception Toolkit)'dır[45]. DTK, bir sistem üzerinde çok çeşitli Unix zafiyetlerini simüle edebilme yeteneğine sahiptir ve birçok farklı ana bilgisayar olarak da maskeleme yapabilmektedir. Bu nedenle bir bal küpü ağı oluşturma yeteneğine sahiptir. Perl'de geliştirilmiştir ve gelen servis isteklerini işlemek için TCP paketleyicileri kullanılmıştır. Basit olduğu için saldırgan tarafından kolaylık fark edilebilir.

Daha sonra CyberCop Sting bal küpü ilk kez ticari olarak geliştirilen balküpü olmuştur[46]. DTK'nın aksine Windows işletim sistemlerini simüle eden bir sisteme sahipti. CyberCop Sting yönlendiricileri, Solaris ve Windows makinelerini simüle edebilir. Sistem ayrıca Telnet gibi bazı hizmetleri de simüle edebilme kabiliyetine sahiptir. Sistem saldırgana karşı bir ağın parçasıymış gibi görünebilir.

Gelişmeler 1999 yılında Honeynet Projesi'nin kurulmasıyla birlikte yeni bir boyut kazandı. Proje ekibinin ana hedefleri, internetteki mevcut tehditler hakkında farkındalık yaratmak ve aynı zamanda genel kamuoyuna gerekli araçları ve yöntemleri sağlamaktı[47].

3.3 Tuzak Sistemlerin Sınıflandırılması

3.3.1 Etkileşim Tabanlı Sınıflandırma

Düşük etkileşimli sistemler genellikle sınırlı işlevselliklere sahip bir simüle hizmet kümesine sahip özellikler sunar. Bu tür sistemler, genellikle saldırılarda kullanılan yöntemlerden ziyade, saldırıların kaynağı hakkında veri toplamak için kullanılır.

Yüksek etkileşimli sistemler, tam kapsamlı bir tuzak sistem sağlama kabiliyetine sahiptirler. Bu sistemlerde saldırganlar sistemle daha fazla etkileşime girebilirler. Bu durumda saldırganın komutları yürütmek ve sisteme dosya yüklemek için izni olduğundan; saldırgan hakkında daha fazla bilgi toplamak mümkündür.

Orta etkileşimli tuzak sistemler, düşük etkileşimli ve yüksek etkileşimli sistemler arasında yer alır.

3.3.2 Kurulum Tabanlı Sınıflandırma

Üretim bal küplerinin kullanımı daha basittir. Daha az bilgi bulundurulur. Üretim bal küpleri çoğunlukla üretim sunucuları ile birlikte üretim ağına konumlandırılırlar. Üretim bal küpleri genelde, kurulumu ve yapılandırması daha kolay olan, düşük etkileşimli bal küpleridir. Saldırılar ve saldırganlar hakkında araştırma bal küplerine nazaran daha az bilgi toplarlar.

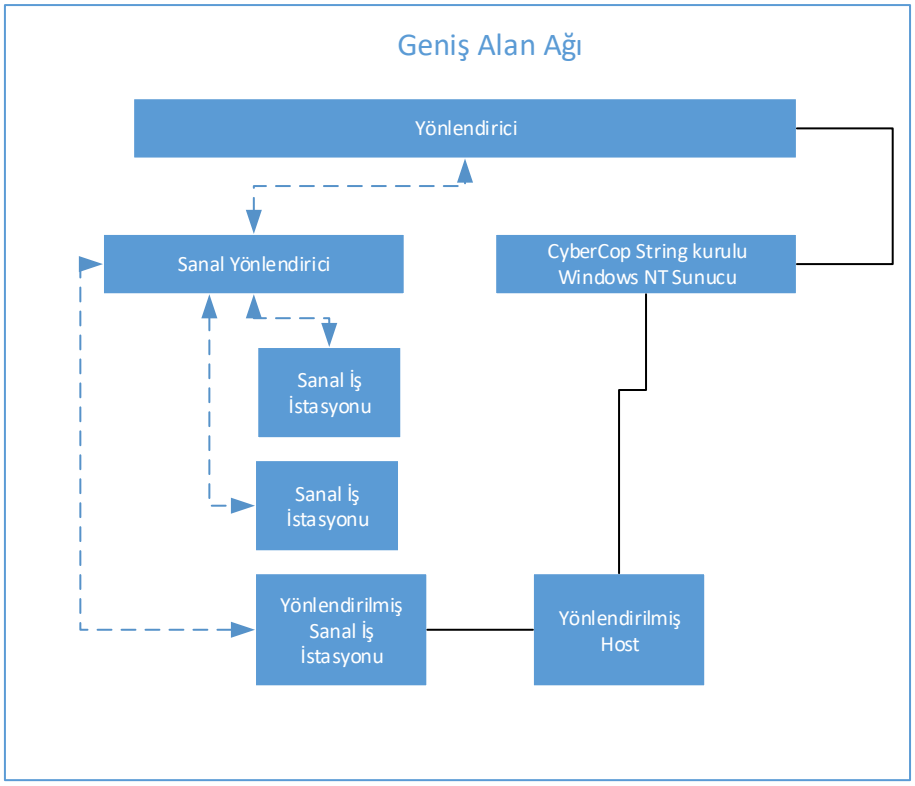
Araştırma bal küpleri, farklı ağları hedef alan saldırgan gruplarının amaçları ve saldırı yöntemleri hakkında bilgi elde etmek için ve kuruluşların karşılaştıkları tehditleri araştırmak ve kuruluşların bu risklere karşı daha iyi nasıl korunabileceklerini öğrenmek maksadı ile kullanılırlar. Araştırma bal küplerinin kurulumu ve idamesi daha zordur. Daha detaylı bilgi barındırdılar ve bilhassa askeri kurumlar, araştırma ve devlet kurumları tarafından tercih edilirler.

3.4 Popüler Tuzak Sistemler

Günümüzde birçok açık kaynak tuzak sistem mevcuttur. Ücretsiz olarak indirilerek kullanılabilirler. Genel olarak kullanılan tuzak sistemler aşağıda anlatılmıştır.

3.4.1 Kippo

Orta etkileşimli bir tuzak sistemdir. Parola kırmak için kullanılan kaba kuvvet saldırılarını ve komut satırı aktivitelerini kaydetmek için kullanılan SSH tabanlı bir tuzak sistemdir. Python dili ile twisted kütüphanesi kullanılarak geliştirilmiştir. Yaygın olarak kullanılan bir modele sahiptir ve bu yapı kullanılarak farklı tuzak sistemler oluşturulabilir. Komut satırı benzetimi yapabilir ve gerçek olamayan bir dosya sistemine sahiptir. Saldırgan sistemde komut koşturabilme ve dosya indirme imkânına sahiptir[48].



Şekil 3.1 CyberCop Sting Tuzak Sistem [48]

3.4.2 Cowrie

Kippo türevi bir tuzak sistemdir. Bu sistem hem Telnet hem de SSH protokollerini desteklemektedir. Linux komutlarının çalıştırılmasına imkân tanır. SFTP ve SCP protokollerini simüle edebilmektedir.

3.4.3 Dionaea

Python dili kullanılarak geliştirilmiştir. Libemu kütüphanesini kullanarak kabuk kod tespiti yapabilir. Ipv6 ve TLS protokollerini destekler. Sistemde zafiyetler barındırır ve zararları yazılımları yakalamak için kullanır[49].

3.4.4 Glastopf

Düşük etkileşimli olan bu sistem web uygulaması olarak hizmet verir. Web saldırıları hakkında bilgi toplamak için kullanılır. Saldırgana gerekli yanıtları vererek bir web servisi işlevi görür. Dosya yükleme ve post metodu ile HTML enjeksiyonu gibi zafiyetleri barındırır.

3.4.5 Thug

Thug istemci tafaında yer alan bir tuzak sistemdir. Bir saldırıya maruz kalmayı beklemek yerine, bir istemci gibi davranır zararlı fonksiyonlara sahip sunucuları bulmaya çalışır. Thug, python dili ile geliştirilmiş bir web tarayıcısıdır.

3.4.6 HonSSH

HonSSH yüksek etkileşimli bir tuzak sistem çözümüdür. Bir sunucuyu simüle etmekten ziyade, daha çok Proxy gibi davranır. Sistem saldırgan ve tuzak sistem arasında bulunuru ve bir SSH Proxy gibi çalışır. Saldırgandan gelen bağlantıları kabul eder ve tuzak sistem ile bağlantı kurmasını sağlar. Üzerinden geçen tüm veriyi kayıt altına alır. HonSSH kippo'dan esinlenerek geliştirilmiştir.

3.5 Nesnelerin Interneti Tuzak Sistemler

3.5.1 Telnet IoT

Telnet saldırılarını yakalamak için geliştirilen bir tuzak sistemdir. Python da yazılmıştır ve daha çok zombi ağları(botnet) zararlılarını tespit etmek için kullanılmaktadır. Cowrie da olduğu gibi saldırgana bir Telnet oturumu sunulur. Elde edilen dosyalar VirusTotal'e yüklenerek daha kapsamlı analiz elde edilebilir[50].

3.5.2 HoneyThing

HoneyThing, Honeynet GsoC projesinin bir parçası olarak oluşturulmuştur. Bu sistem TR-069 nesnelerinin benzetimi için kullanılmaktadır. Sistem hem web ara yüzünü hem de CWMP protokolünü simüle edebilmektedir. IoT sistemlerde bulunan ve yaygın olarak bilinen zafiyetlerin benzetimini yapabilir[51].

3.5.3 MTPot

Cymmetria Research tarafından açık kaynak olarak geliştirilmiştir[52]. Mirai zararlı yazılımının etkilediği makinaları tespit etmek için kullanılan bu sistemin kullanımı oldukça kolaydır. Diğer Telnet sistemlerde olduğu gibi Telnet servisini simüle eder ve ayarları tespit edilmek istenen mirai zararlısının sürümüne göre değiştirilebilir.

3.5.4 IoTPOT

IoTPOT, Alman ve Japon üniversitelerindeki araştırmacılar arasında işbirliğine dayalı bir çalışmadır. IoTPOT, Telnet saldırıları için bir IoT tuzak sistem ve kum havuzundan oluşur. Çeşitli cihazların Telnet servislerinin benzetimini yapabilir. İki bölümden oluşur; düşük etkileşimli bir önyüz ve IoTBOX olarak adlandırılan yüksek etkileşimli bir sanal

arka yüzden meydana gelir. IoTBOX MIPS ve ARM mimarilerinin de arasında bulunduğu sekiz farklı işlemci mimarisini destekler. Önyüz saldırgandan gelen bağlantı isteklerini ve komutları arka yüze iletir. Daha sonra arka yüzden gelen yanıtları saldırgana iletir[53].

4.MQTT

4.1 Giriş

MQTT, veri aktarımı için TCP/IP iletim katmanı protokolünün üzerinde çalışan bir uygulama katmanı protokolüdür. Kaynak kısıtlı cihazlara uygun bir protokoldür ve 1999 yılında Dr. Andy Stanford-Clark ve Arlen Nipper tarafından geliştirilmiştir. Hafif ve kullanımı kolay bir protokoldür, bu da IoT gibi kaynak kısıtlı sistemlerde iletişimin sağlanması için elverişli bir aktarım imkânı sunmaktadır. MQTT'nin tasarım amacı, kaynak kısıtlı sistemler için düşük bant genişliği ve güvenilmez ağlar gibi ortamlarda güvenilir mesaj iletimini sağlamaktır. Verilerin teslimatı istemci-sunucu (client-server) ve yayın-abone(Publish/Subscribe) mekanizmaları kullanılarak yapılır[54].

4.2 Protokol Mimarisi

4.2.1 Yayın/Abone Mekanizması

İstemci-sunucu modelinde, iletişim kurucu istemci ve sunucu arasında doğrudan bir bağlantı vardır. Buna karşılık, yayın-abone (pub/sub) modelinde hem yayıncı hem de abone iletişim halinde olurlar. Mesajı gönderen bir yayıncının abonenin varlığını bilmesi gerekmemektedir. Yayımcı-abone mekanizmasında taraflar birbirinin varlığından haberdar olmadığı için iki tarafta istedikleri zaman mesaj alıp gönderebilmektedirler. Mesajların yayınlanması ve abone olunması, broker olarak adlandırılan merkez tarafından yapılır. Broker, yayınlanmış mesajı ilgili abonelere veya abone grubuna filtreler ve yayınlar. Broker, mesajların filtrelenmesine bakarak hangi mesajın hangi mesajı alacağına karar verir[55].

4.2.2 Mesaj Filtreleme

Broker, her abonenin abonelik durumlarına göre mesajları alması için mesaj filtrelemeyi kullanır. Yayım/Abone sistemleri, konu tabanlı filtreleme, içerik tabanlı filtreleme ve tür tabanlı filtreleme gibi mesaj filtreleme seçeneklerine sahiptir.

- Konu tabanlı filtreleme, burada konuya bakılarak filtreleme yapılır. Her mesajda konu kısmı yer almaktadır. Mesajı almak isteyen abone burada sadece konuyu temel alarak üye olacaktır.
- İçerik tabanlı filtreleme, burada mesaj içeriğine bakılarak filtreleme yapılır. Örneğin, 25 dereceden daha yüksek sıcaklık verisi içeren mesajın filtrelenmesi buna örnek verilebilir.
- Tür tabanlı filtreleme, burada mesajın konusunun türüne göre filtreleme yapılır. Bu tür bilgisi her bir mesaja eklenmiştir. Burada tür sıcaklı, nem vs. olabilir.

4.2.3 MQTT Konu Tabanlı Filtreleme

MQTT protokolü konu tabanlı filtrelemeyi kullanmaktadır. Bir konu başlığı metinsel ifadelerden oluşur ve hiyerarşik bir yapıya sahiptir. Birden fazla konu kesme işareti ile ayrılarak hiyerarşik bir yapı oluşturulur.

Örneğin: /sıcaklık/fabrika/giriskat/ofis1

Burada ‘sıcaklık’ birinci seviye konudur, fabrika ikinci seviyedir ve hiyerarşi bu şekilde devam eder. Konu için kullanılan metinler konunun amacını gösterdiği için bu metinler açıklayıcı olmalıdır. Abone, yukarıdaki gibi tek bir konuya abone olabilir veya joker karakterlerini kullanarak aynı anda birden çok konuya abone olabilir.

Örneğin: /sıcaklık/fabrika/+/ofis

Yukarıdaki konu tek seviye joker karakterinin kullanımını göstermektedir. Bu konu ile ‘giriskat’ başlığı altındaki ofislere ait veriler alınabilir.

/sıcaklık/fabrika/#

Bu konu çok düzeyli joker karakterinin kullanımını göstermektedir. Bu konu kullanılarak ‘fabrika’ konusu altındaki bütün veriler alınabilir. Konu, istemci (client) tarafından yayınlanan ve abone olunan her bir mesaja eklenmiştir. Broker, bir abonenin mesajı alıp almayacağına karar vermek için bu konuları kullanır[56].

4.2.4 MQTT İstemci ve Broker

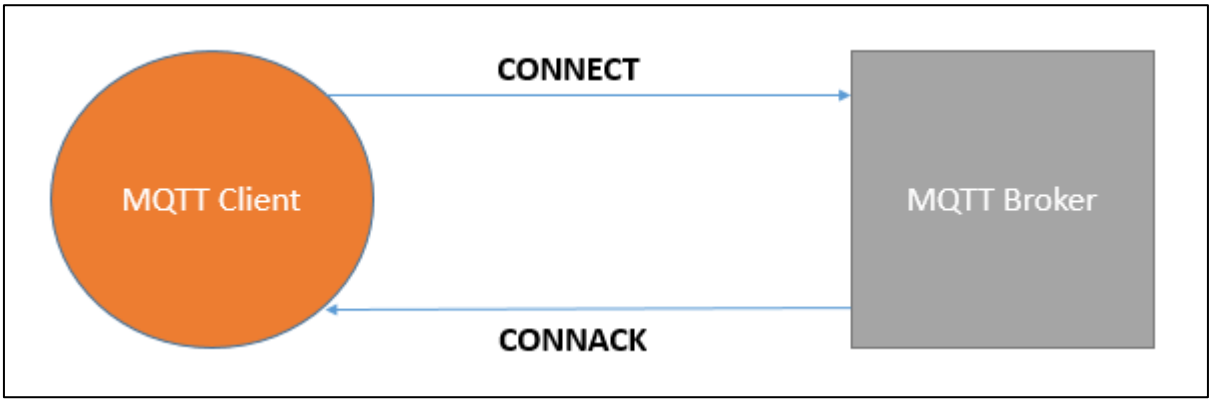
MQTT istemcisi, belirli konularla ilgili aynı anda mesaj yayınlayabilir ve broker’a abone alabilir. Bu genellikle kaynak kısıtlı olan bir sensör ya da grafik ara yüze sahip bir istemci olabilir.

MQTT broker yayım/abone mekanizmasının merkezinde yer almaktadır. Broker, yayınlanan ve abone olunan tüm mesajların alınmasından, bu mesajların

filtrelenmesinden ve mesajın gönderilmesi için uygun abonelere karar verilmesinden sorumludur. Yayımcı tarafından gönderilen ve abone tarafından alınan her veri broker üzerinden geçmek zorundadır.

4.2.5 MQTT Bağlantısı

MQTT bağlantısı her zaman istemci ile broker arasında kurulur, iki istemci asla birbiriyle doğrudan bağlantı kurmazlar. İstemci, broker ile CONNECT mesajını kullanarak bağlantı kurar. Buna yanıt olarak broker CONNACK mesajını gönderir. CONNACK mesajındaki dönüş kodu, bağlantının başarılı olup olmadığını belirler. MQTT bağlantısı Şekil 4.1'de gösterilmiştir. İstemci broker ile başarılı bir şekilde bağlantı kurduktan sonra, artık broker'a mesaj yayınlayabilir veya abone olabilir.



Şekil 4.1 MQTT Bağlantısı

4.2.6 Yayınla ve Üye Ol (Publish ve Subscribe)

Her yayınlama ve abone olma mesajı, mesajların filtrelenmesi için aracı tarafından kullanılan konuyu içerir. Publish mesajı konu ile birlikte yüküde içermektedir. Bu yük iletilecek olan gerçek veridir. Yük ikili veri, metin verileri ve hatta JSON veya XML verilerini içerebilir. Publish mesajının içeriği Tablo 4.1'de gösterilmiştir.

PUBLISH	
topic	/oda1/sıcaklık
payload	"sıcaklık: 25-derece"

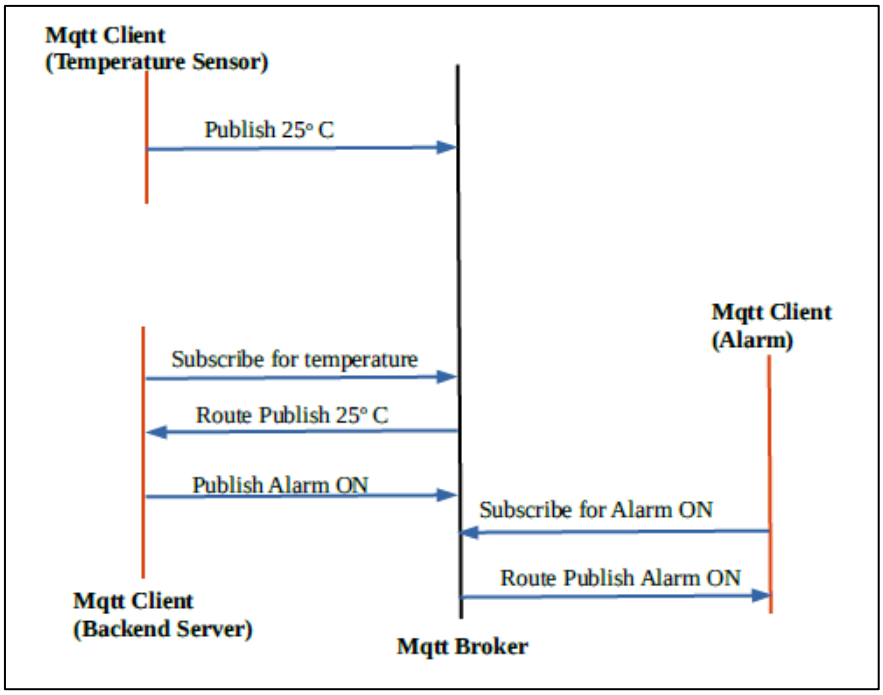
Tablo 4.1 Publish Mesajı

SUBSCRIBE mesajı, abonenin ilgilendiği konu adlarının listesini içerir. Bu nedenle abone, bir veya daha fazla konuya abone olabilir. SUBSCRIBE mesaj formatı Tablo 4.2'de gösterilmiştir.

SUBSCRIBE	
topic1	/oda1/sıcaklık
topic2	/oda2/sıcaklık
topic3	/oda3/sıcaklık

Tablo 4.2 Subscribe Mesajı

MQTT yayınlama / abone olma mesajlaşma sisteminde, yayımcı belirli konulara ve istemcilere ilgili konulara ilişkin verileri yayınlar ve ilgili mesajı brokerdan alır.



Şekil 4.2 MQTT Broker ve Client Yapısı

Şekil 4.2'de MQTT istemcilerin ve broker'ın çalışma yapısı gösterilmiştir.

4.2.7 MQTT Kalite Servisi (Quality of Service)

Hizmet kalitesi, kısaca QoS, ağ iletişimi hizmet kalitesi olarak bilinmektedir. Ağ üzerindeki iletişimi önceliklendirerek zaman kaybını aza indirmeyi hedefleyen bir ağ

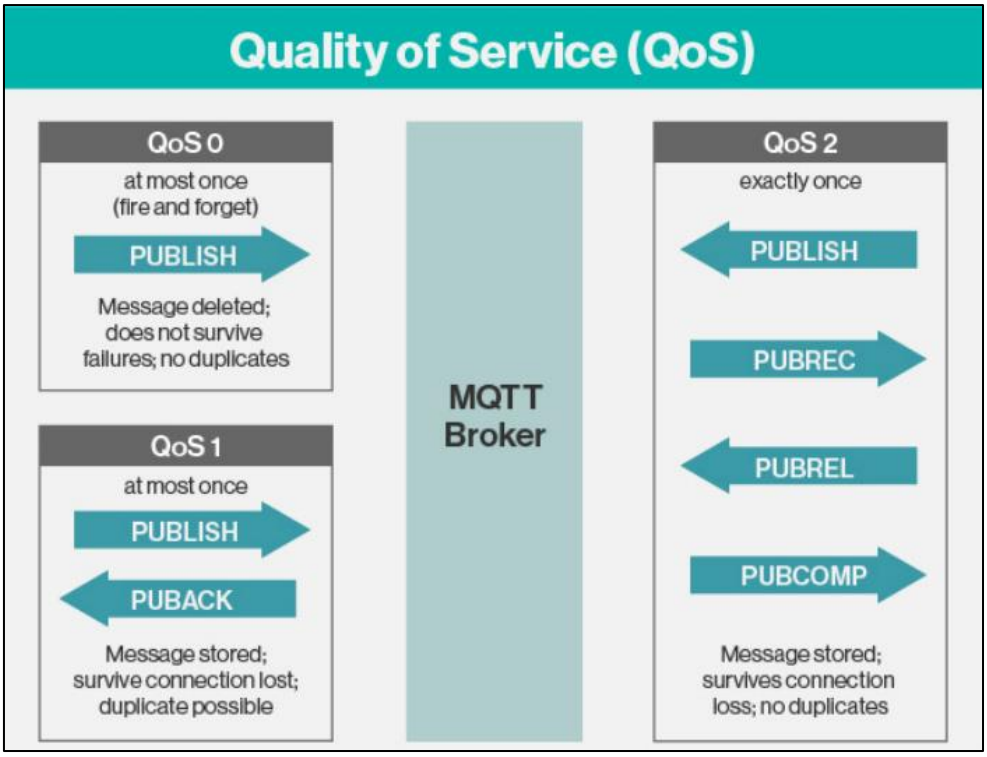
hizmetidir. Bir ağ bağlantısı üzerinden gönderilen bir trafik veya program türüne öncelik veren çeşitli teknikler olarak tanımlanabilir.

Üç farklı kalite hizmeti seviyesi vardır. Bu seviyeler MQTT protokolünün içerikleri nasıl yöneteceğini ortaya koyar. Yüksek seviyeli kalite hizmetleri daha güvenilir olsa da daha fazla gecikme olabilir ve bant genişliği ihtiyaçları olmaktadır. Bu nedenle üye birimler kendileri ne almak istiyorlarsa ona göre yüksek kalite hizmeti seviyelerini seçebilirler. Kalite hizmeti seviyeleri üç bölümden oluşmaktadır.

Bunlardan birincisi “Unacknowledged Service” yani onaylanmamış servistir. Bu kalite hizmeti seviyesi PUBLISH (Yayın) paket sıralamasını kullanır. Yayıncı, broker birimine bir kez gönderi yollar ve broker da bu gönderiyi bir kez üye birime yollar. Gönderinin ulaşıp ulaşmadığından emin olunacak bir yapı yoktur ve gönderi kaydedilmez. Bu kalite seviyesi “QoS0” olarak da adlandırılır. Özetlemek gerekirse bu kalite seviyesinde gönderi iletilmeyebilir ama en alt sınırda trafik vardır.

Kalite hizmeti seviyelerinden ikincisi ise “Acknowledged Service” yani kabul edilmiş servistir. Bu kalite hizmeti seviyesi PUBLISH/PUBACK paket sıralamasını kullanır. Onaylı bir paket, içeriğin alındığını ve eğer alınmadıysa belirli zaman aralıklarında içeriğin tekrar yollandığını kontrol eder ve doğrular. Bu durumda üye birim aynı gönderiden birden çok kez alabilir. Bu kalite hizmeti seviyesi “QoS1” olarak da tanımlanır. Özet olarak bu kalite hizmeti seviyesinde gönderi kesin gönderilir fakat gönderi birden fazla kez iletilebilir.

Hizmet kalitesi seviyelerinden üçüncüsü “Assured Service” yani garanti edilmiş hizmet denilebilir. Bu hizmet kalite seviyesi gönderiyi iki çift paket halinde gönderir. İlk çift PUBLISH/PUBREC olarak isimlendirilirken ikinci gönderi paketi çifti PUBREL/PUBCOMP olarak isimlendirilir. Bu hizmet kalitesi seviyesinde gönderi kesin iletilir. Tek seferde iletilen bu hizmet kalitesi seviyesinde maksimum trafik vardır.[57] Kalite hizmet seviyeleri Şekil 4.3’te gösterilmiştir.



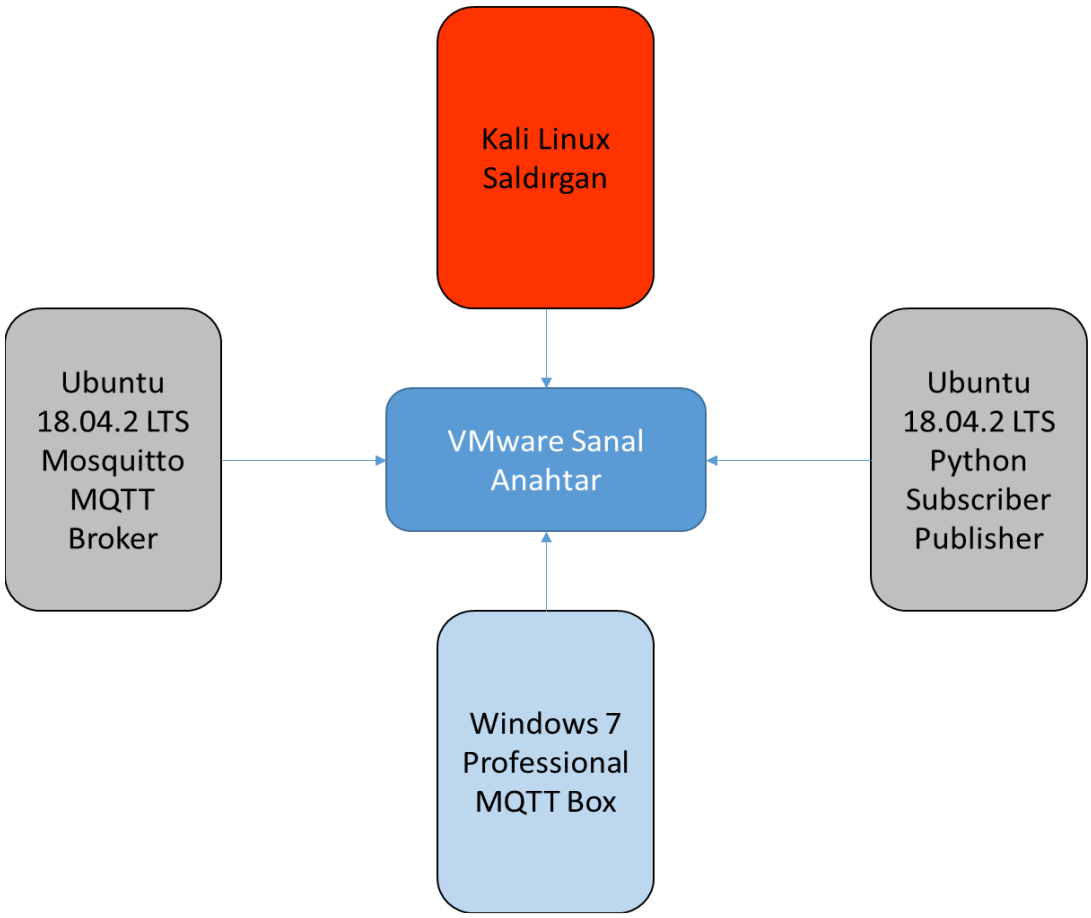
Şekil 4.3 MQTT Kalite Servisi[57]

5.SALDIRI BENZETİMİ

5.1 Saldırı Ortamı

Benzetim işlemleri Microsoft Windows 7 hostu üzerinde bulunan Vmware Workstation 11.1.4 build-3848939 üzerinde gerçekleştirilmiştir. Ubuntu 18.04.2 LTS üzerinde broker yazılımı olarak Mosquitto 1.5.4 kullanılmıştır. Broker'a publish ve subscribe paketleri göndermek için Python dili ile geliştirilen betikler kullanılmıştır. Sistemlerin birlikte çalışması için Vmware Sanal Anahtar yapısı kullanılmıştır.

Ubuntu üzerinde çalıştırılan python betikleri ile publish ve subscribe paketleri broker'a gönderilmiş ve aynı zamanda gönderilen paketler MQTT Box üzerinden de izlenebilmiştir. Aynı ağ üzerinde yer alan Kali Linux saldırgan tarafından saldırılar gerçekleştirilmiştir. Benzetim ortamı topolojisi Şekil 5.1'de gösterilmiştir.



Şekil 5.1 Benzetim Ortamı Topolojisi

5.1.2 Python Publisher

Broker'a mesajları göndermek için Python dilinde geliştirilen betik kullanılmıştır. Python içerisinde yer alan Paho Kütüphanesi kullanılarak geliştirilen betik Şekil 5.2' de gösterilmiştir.

```
import paho.mqtt.client as mqttistemci
import time
 def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Broker Bağlandı")
        global Connected
       Connected = True
     else:
       print("Bağlantı Hatası")
```

```
Connected = False
broker_address= “192.168.241.141”
port = 1883
user = “admin”
password = “admin”
client = mqttClient.Client(“Python”)
client.username_pw_set(user, password=password)
client.on_connect= on_connect
callback
client.connect(broker_address, port=port)
client.loop_start()
while Connected != True:
   time.sleep(0.1)
 try:
   while True:
       mesaj = raw_input(‘Mesaj Giriniz:’)
      client.publish(“Oda1/Sıcaklık”,mesaj)
except KeyboardInterrupt:
   client.disconnect()
   client.loop_stop()
```

Şekil 5.2 Python Publisher

5.1.3 Python Subscriber

Broker’dan mesajları almak için Python dilinde geliştirilen betik kullanılmıştır. Python da yer alan paho kütüphanesi kullanılarak geliştirilen betik Şekil 5.3’ de gösterilmiştir.

```
import paho.mqtt.client as mqttistemci
import time
def on_connect(client, userdata, flags, rc):
    if rc == 0:
       print(“Broker’a Bağlandı”)
       global Connected              #Use global variable
      Connected = True               #Signal connection
```

```
    else:
        print("Bağlantı Hatası!")
 def on_message(client, userdata, message):
   print "Mesaj Alındı: "  + message.payload
 Connected = False   #global variable 36owrie36 state of the connection
 broker_address= "192.168.241.141"   #Broker adresi
port = 1883
user = "admin"
password = "admin"
client = mqttClient.Client("Python")
client.username_pw_set(user, password=password)
client.on_connect= on_connect
client.on_message= on_message
client.connect(broker_address, port=port
client.loop_start()
while Connected != True:
time.sleep(0.1)
client.subscribe("Oda1/Temp")
try:
   while True:
      time.sleep(1)
except KeyboardInterrupt:
   print "exiting"
   client.disconnect()
   client.loop_stop()
```

Şekil 5.3 MQTT Subscriber

5.1.4 Saldırı Araçları

Kullanılan saldırı araçları Tablo 5.1 de gösterilmiştir.

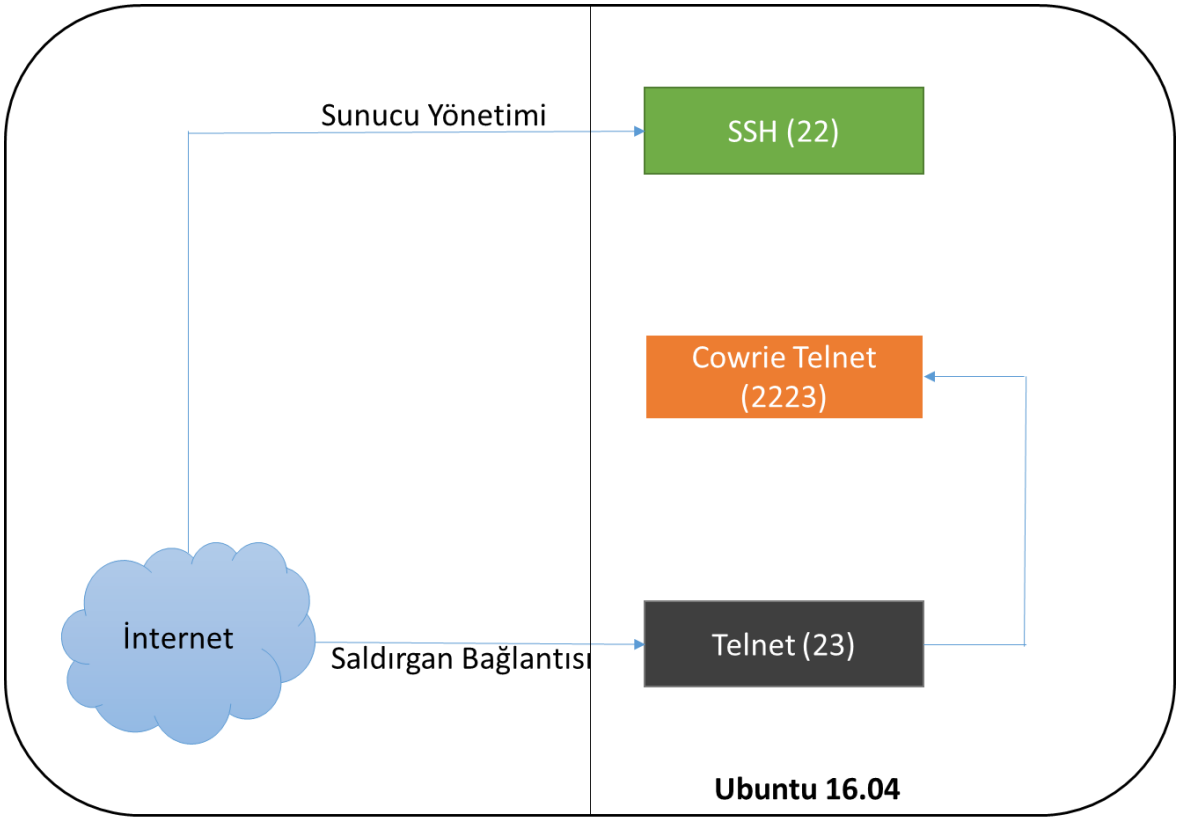
Araç	Kullanım Amacı
Nmap	Ağ Keşfi
Metasploit	Exploit Çalıştırma
Wireshark/Tcpdump	Ağ Trafiğinin Dinlenmesi
Cowrie	Telnet Tuzak Sistem
Mosquitto	MQTT Broker
Python Betikleri	MQTT İstemci
MQTT Box	MQTT İstemci
Hping3	DDOS Saldırı Aracı
Mqtt-benchmark	Broker Performans Analizi

Tablo 5.1 Saldırı Araçları

5.2 Telnet Tuzak Sistemi

Telnet protokolüne yapılan saldırıların benzetimi için Cowrie[58] tuzak sistemi kullanılmıştır. Cowrie, kaba kuvvet saldırılarını ve saldırgan tarafından gerçekleştirilen kabuk etkileşimini kayıt altına almak için tasarlanmış orta seviyede bir Telnet bal küpüdür. Cowrie mimari yapısı şekil 5.4'te gösterilmiştir. Tuzak sistemin önemli özellikleri şunlardır;

- Dosya ekleme ve kaldırma özelliğine sahip sahte dosya sistemi.
- Saldırganın /etc/passwd gibi dosyaları yakalayabilmesi için sahte dosya içeriği ekleme kabiliyeti.
- Oturum günlükleri bin / playlog yardımcı programını kullanarak orijinal zamanlama ile tekrar oynatmak için UML Uyumlu bir formatta saklanır.
- Dosya yükleme için SFTP ve SCP desteği vardır.
- SMTP bağlantı kayıtlarını SMTP Honeypot'a iletebilir.
- Günlük yönetimi çözümlerinde kolay işlem yapmak için JSON biçiminde kayıt tutabilmektedir.



Şekil 5.4 Cowrie Mimari Yapısı

Tuzak sisteme ait önemli dosyalar aşağıda gösterilmiştir;

- 38owrie.cfg: Cowrie yapılandırma dosyası. Varsayılan değerler etc/38owrie.cfg.dist dosyası içerisinde yer almaktadır.
- share/38owrie/fs.pickle: Sahte dosya sistemi.
- etc/userdb.txt: Tuzak sisteme erişim için kullanılan hesap bilgileri.
- honeyfs/: Sahte dosya sistemi için kullanılan diğer dosyalar.
- honeyfs/etc/issue.net: Giriş öncesi görülen mesaj bilgisini tutar.
- honeyfs/etc/motd: Sisteme giriş yaptıktan sonra görülen uyarı mesajıdır.
- var/log/38owrie/38owrie.json – transaction output in JSON format
- var/log/38owrie/38owrie.log – log/debug output
- var/lib/38owrie/tty/: Oturum kayıtlarını tutmaktadır.
- var/lib/38owrie/downloads/: Saldırgan tarafından sisteme yüklenen dosyaların tutulduğu alandır.

- bin/playlog: Kaydedilen oturum kayıtlarını tekrar oynatmak için kullanılan yardımcı program.

5.3 Telnet Saldırıları

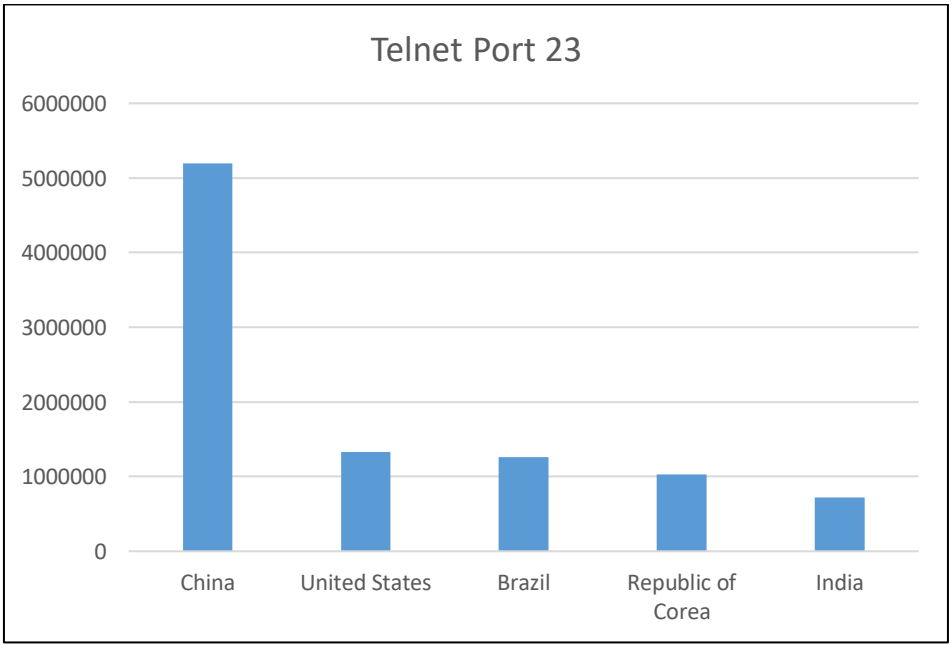
Telnet[59], İnternet üzerindeki çok kullanıcılı bir sisteme uzaktaki başka bir bilgisayardan bağlanmak için geliştirilen bir TCP/IP protokolü ve bu işi yapan programlara verilen genel bir isimdir. Bu protokol, İnternet üzerindeki en eski uzaktan erişim protokollerinden biridir. 1969'da IP ağının ilk zamanlarında kullanıma sunuldu ve uzun süredir uzak ağdaki bilgisayarlara erişmek için varsayılan olarak kullanıldı. Telnet programı ile sanal sunucunuza bağlandığınızda, uzak işletim sistemine bağlanmış olursunuz. Telnet güvensiz bir protokoldür ve tüm veriler açık metin olarak gönderilir. Bu yüzden telnet oturumundan, koklayıcı (sniffer) kullanılarak kolay bir şekilde hassas bilgilere erişilebilir.

IBM Managed Security Services tarafından yapılan bir araştırmada, müşterilerinin 2016 yılının ikinci çeyreğinde yaptığı bağlantılardan, telnet protokolü kullanımı (TCP bağlantı noktası 23), tüm bağlantıların yüzde 79'una karşılık gelerek en fazla bağlantı yüzdesine sahip olmuştur.[60] Elde edilen bağlantı bilgileri ile ilgili detaylar tablo 5.2'de gösterilmiştir.

Rank	Destination TCP port	Sweeps	Internet Assigned Numbers Authority (IANA)-assigned service description and popular use[22]
1	23	78.65%	telnet
2	1433	2.61%	Microsoft SQL Server
3	8080	2.14%	HTTP alternate for port 80
4	3306	1.59%	MySQL
5	3389	1.54%	MS WBT Server, Windows Remote Desktop
6	3128	1.00%	Active API Server Port, some proxy servers (squid-http, 3proxy)
7	443	0.90%	http protocol over TLS/SSL
8	5900	0.61%	Remote framebuffer, VNC (virtual network computing), Apple Remote Desktop
9	9200	0.56%	WAP connectionless session service, EMC2 (Legato) Networker or Sun Solstice Backup
10	21320	0.54%	N/A
	All other	9.87%	All other TCP ports combined

Tablo 5.2 Bağlantı Bilgileri[61]

İnternete bağlı olan sistemleri aramak için oluşturulan arama motoru Shodan tarafından 4 Nisan 2016'da yapılan bir araştırmada[61], telnet protokolünün aktif olarak birçok sistemde kullanıldığı bildirilmiştir. Yapılan arama sonuçları şekil 5.5'de gösterilmiştir.



Şekil 5.5 Shodan Arama Sonuçları

Her uygulamada yer alan başlık bilgisi (banner) sayesinde saldırganlar hedef sistemlerde çalışan hizmetler hakkında bilgi elde edebilmektedirler. Elde edilen servis bilgilerini kullanarak daha gelişmiş saldırılar için hazırlık yapabilirler. Bir saldırgan açık bir telnet portunu keşfettiğinde şunları yapabilir;

- Hedef sistemde çalışan telnet servisi hakkında detaylı bilgiler elde edebilir.
- Telnet kimlik doğrulaması olmayan bir yapılandırmaya sahip ise direk olarak hedef sisteme erişebilir.
- Varsayılan olarak bırakılan kullanıcı adı ve parola bilgilerini (root, admin) kullanarak hedef sistemde erişim elde edebilir.
- Sistem yöneticisinin telnet protokolüne yaptığı bağlantıları sniffer ile dinleyerek trafik bilgisini kayıt altına alabilir.
- Telnet servis hesabına kaba kuvvet saldırıları yaparak servis kullanıcı bilgilerini elde edebilir.
- Servis dışı bırakma saldırısı yaparak hedef servisi hizmet veremez hale getirebilir.

5.3.1 Bilgi Toplama

Bu bölümde nmap aracı kullanılarak hedef sistemde çalışan telnet protokolü hakkında bilgiler elde edilmiştir. Kali Linux işletim sistemi üzerinde çalıştırılan komut seti ekran görüntüsü şekil 5.6'de gösterilmiştir.

```
root@kali:~# nmap -sS -sV 192.168.241.138 -p 23
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-23 05:18 PDT
Nmap scan report for 192.168.241.138
Host is up (0.00026s latency).

PORT   STATE SERVICE VERSION
23/tcp open  telnet  Linux telnetd
MAC Address: 00:0C:29:0E:16:09 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

Şekil 5.6 Nmap Port Taraması

Telnet servisinin hedef sistemde 23 portu üzerinden hizmet verdiği ve Linux işletim sistemi üzerinde çalıştığı tespit edilmiştir.

5.3.2 Kaba Kuvvet Saldırısı

Telnet kullanıcı adı ve şifre bilgilerini elde etmek için kaba kuvvet saldırısı (brute force) yapılmıştır. Bu saldırı hedef sistemde kullanılan kimlik bilgilerinin tahmin edilmesine dayalı bir tekniği barındırmaktadır. Daha önce oluşturulan listeler kullanılarak hedef sistem kimlik bilgileri tahmin edilmeye çalışılmıştır. Saldırı için Kali Linux üzerinde yer alan metasploit framework içerisinde bulunan *auxiliary/scanner/telnet/telnet_login* aracı kullanılmıştır. Saldırı için kullanılan komutlar aşağıda listelenmiştir.

use auxiliary/scanner/telnet/telnet_login

msf auxiliary(telnet_login) > set rhosts 192.168.241.138

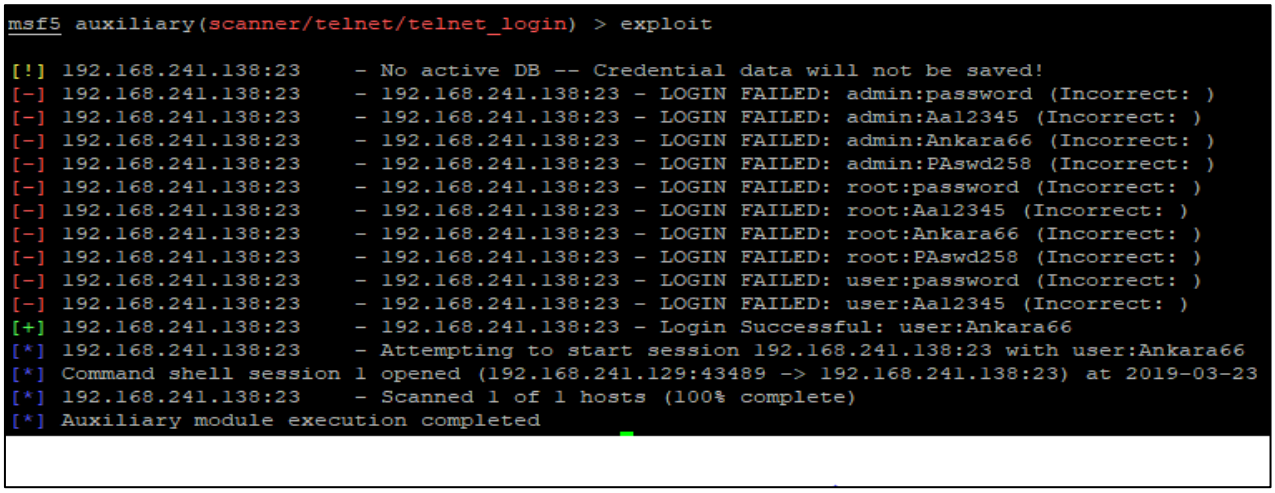
msf auxiliary(telnet_login) > set user_file /root/Desktop/user.txt

msf auxiliary(telnet_login) > set pass_file /root/Desktop/pass.txt

msf auxiliary(telnet_login) > set stop_on_success true

msf auxiliary(telnet_login) > exploit

Yapılan saldırı ve sonucuna ait ekran görüntüsü şekil 5.7'de gösterilmiştir.

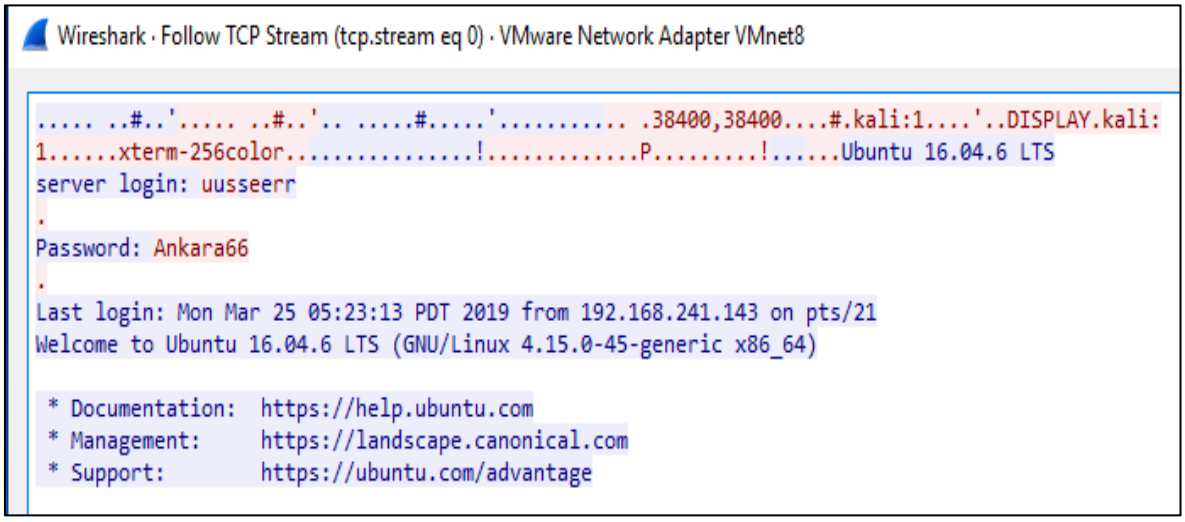


Şekil 5.7 Kaba Kuvvet Saldırısı

Kaba kuvvet parola saldırılarına karşı kırılganlığı sınırlandırmak için yapabileceğiniz en temel şey, parolalarınızı sık sık değiştirmek ve güçlü parolalar kullanmaktır. Minimum 10 karakter uzunluğunda, büyük harf, küçük harf, rakam ve özel karakter kullanılarak parola üretilmelidir.

5.3.3 Trafik Dinleme

Paket dinleme, belirli bir ağdan geçen tüm veri paketlerini izleme ve yakalama işlemidir. Paket dinleyicileri genellikle ağ trafiğini izlemek ve gidermek amacıyla ağ ve sistem yöneticisi kişiler kullanmaktadır. Ayrıca saldırganlar da şifre, hesap bilgileri vb. hassas bilgileri içeren veri paketlerini yakalamak için paket dinleyici kullanabilirler. Bu yazılımlara örnek olarak Wireshark, Tcpdump ve Network Miner araçları verilebilir. Kötü niyetli bir davetsiz misafir, ağ üzerindeki bir paket dinleyiciyi ağa yerleştirerek tüm ağ trafiğini yakalayabilir ve analiz edebilir. Bu saldırıda telnet protokolü iletişimini yakalamak için Wireshark yazılımı kullanılmıştır. Hedef sistem ile yapılan telnet bağlantısına ait kullanıcı bilgileri ağ trafiği dinlenerek elde edilmiştir. Trafik dinleme sonucu elde edilen bilgiler şekil 5.8'de gösterilmiştir. Trafik dinleme saldırılarını engellemek için iletişim kurulan bağlantı altyapısı için şifreleme teknolojileri kullanılabilir. Telnet protokollünün birçok zafiyet içermesinden dolayı alternatif olarak şifreli haberleşme imkânı sunan SSH protokolü kullanılmalıdır. Buna ek olarak farklı konumlarda yer alan ağ bölümlerinin bağlantısı için IPSec (Internet Protocol Security) protokolü kullanılabilir. IPSec kullanılarak uçtan uca hat şifrelemesi ile trafik dinleme saldırıları engellenebilir.



Şekil 5.8 Wireshark Trafik Bilgileri

5.3.4 Servis Dışı Bırakma

Hizmet reddi (DoS) veya servis dışı bırakma saldırısı, bir saldırganın veya saldırganların bir hizmetin sunulmasını engellemeye çalıştığı saldırı türüdür. Bu saldırı, neredeyse her türden dijital ortama erişimi engelleyerek başarılabilir: sunucular,

cihazlar, hizmetler, ağlar, uygulamalar ve hatta uygulamalar içindeki belirli işlemler. Saldırı belirli bir kaynaktan geliyor ise burada DoS (Denial of Service) terimi; birden çok kaynaktan geliyor ise DDoS (Distributed Denial of Service) terimi kullanılmaktadır. Genellikle, bu saldırılar hedef sistem kaynaklarını ya da bant genişliğini tüketerek, veri talep eden bir sistemi isteklere cevap veremeyecek hale getirir.

Bu saldırı benzetiminde Kali Linux işletim sistemi üzerinde bulunan Hping3 aracı kullanılarak Telnet servisinin hizmet verdiği sunucuya yoğun bir trafik gönderimi yapılacak ve sistemde oluşan olası kaynak tüketimleri gözlemlenecektir. Hping3 aracı kullanılarak yapılan saldırıda kullanılan komut seti aşağıda gösterilmiştir.

root@kali: hping 192.168.241.142 –S –flood –p 23 –rand-source

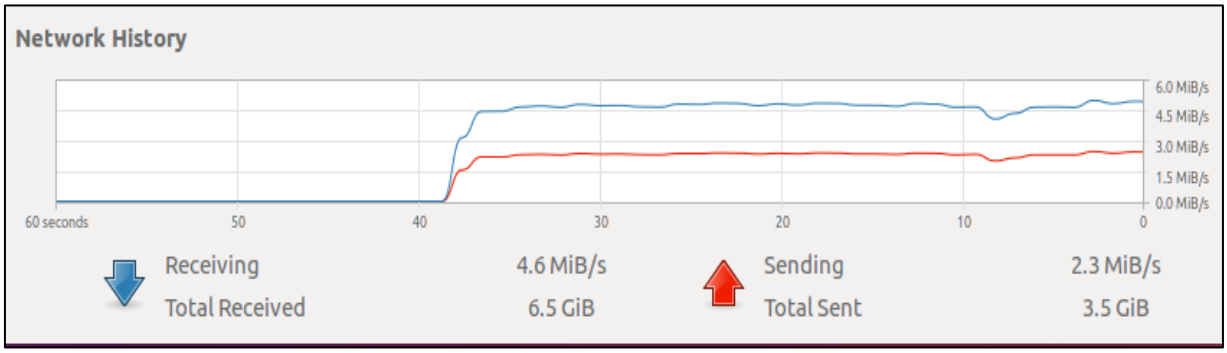
Burada hedef sisteme SYN Flood olarak adlandırılan DdoS saldırısı gerçekleştirilmiştir. Hedef sisteme sürekli olarak SYN bayrağı ile oluşturulmuş TCP paketleri farklı kaynak IP adreslerinden gönderilmiştir. Yapılan saldırıda elde Wireshark aracı ile elde edilen trafik görüntüsü şekil 5.9'da gösterilmiştir.

No.	Time	Source	Destination	Protocol	Length	Info
4595…	7.222869	40.115.140.227	192.168.241.142	TCP	54	12470 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.222881	50.24.173.42	192.168.241.142	TCP	54	12471 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.222936	4.55.84.221	192.168.241.142	TCP	60	12490 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.222943	17.22.56.99	192.168.241.142	TCP	60	12491 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.222982	115-36-245-196.chub…	192.168.241.142	TCP	60	12492 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.222987	174-135-61-17.res.b…	192.168.241.142	TCP	60	12493 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223022	136.236.216.161	192.168.241.142	TCP	60	12494 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223027	39.91.131.101	192.168.241.142	TCP	60	12495 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223062	171.106.232.84	192.168.241.142	TCP	60	12496 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223067	228.95.100.106	192.168.241.142	TCP	60	12497 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223079	192.168.241.142	157.33.158.176	TCP	60	1883 → 12472 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
4595…	7.223096	157.33.158.176	192.168.241.142	TCP	54	12472 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.223110	192.168.241.142	184.135.16.80	TCP	60	1883 → 12473 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
4595…	7.223116	232.132.133.112	192.168.241.142	TCP	60	12498 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223118	184.135.16.80	192.168.241.142	TCP	54	12473 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.223122	103.239.183.214	192.168.241.142	TCP	60	12499 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223179	205.161.135.224	192.168.241.142	TCP	60	12500 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223180	192.168.241.142	40.9.213.37	TCP	60	1883 → 12474 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
4595…	7.223192	123.158.58.177	192.168.241.142	TCP	60	12501 → 1883 [SYN] Seq=0 Win=512 Len=0
4595…	7.223199	40.9.213.37	192.168.241.142	TCP	54	12474 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.223219	192.168.241.142	100.173.75.251	TCP	60	1883 → 12475 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
4595…	7.223233	100.173.75.251	192.168.241.142	TCP	54	12475 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.223263	192.168.241.142	131.220.207.42	TCP	60	1883 → 12476 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
4595…	7.223271	192.168.241.142	46-150-206-16.gecom…	TCP	60	1883 → 12477 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
4595…	7.223276	131.220.207.42	192.168.241.142	TCP	54	12476 → 1883 [RST] Seq=1 Win=32767 Len=0
4595…	7.223292	46-150-206-16.gecom…	192.168.241.142	TCP	54	12477 → 1883 [RST] Seq=1 Win=32767 Len=0

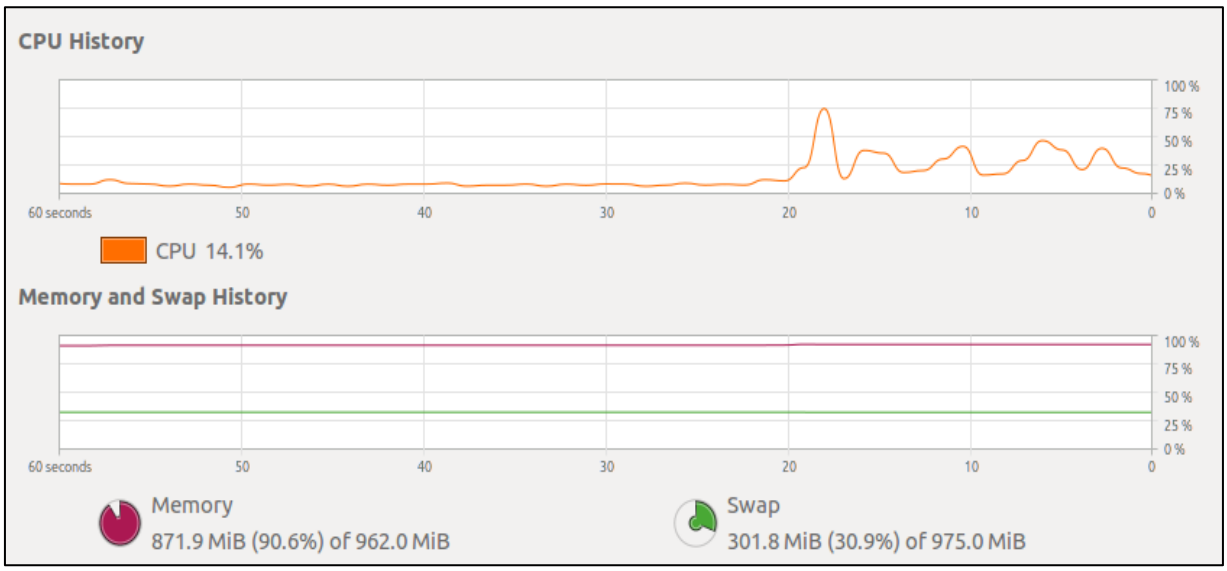
Şekil 5.9 SYN Flood Trafiği

Saldırı yapılan hedef sistem incelenmiş, saldırı öncesi ve sonrası kullanılan sistem kaynaklarına ait veriler paylaşılmıştır. Saldırı sırasında ağ trafiği 4.6 MB/s seviyesine ulaşmıştır. Sistem kaynaklarında yoğun bir tüketim artışı olduğu tespit edilmiştir. İşlemci

kullanımı saldırı öncesi %0,7 kullanılırken saldırı sırasında kullanım oranı %14,1 olmuştur. Sistemde bellek kullanımında değişim olmadığı gözlemlenmiştir.



Şekil 5.10 SYN Flood Esnasında Trafik Kullanımı

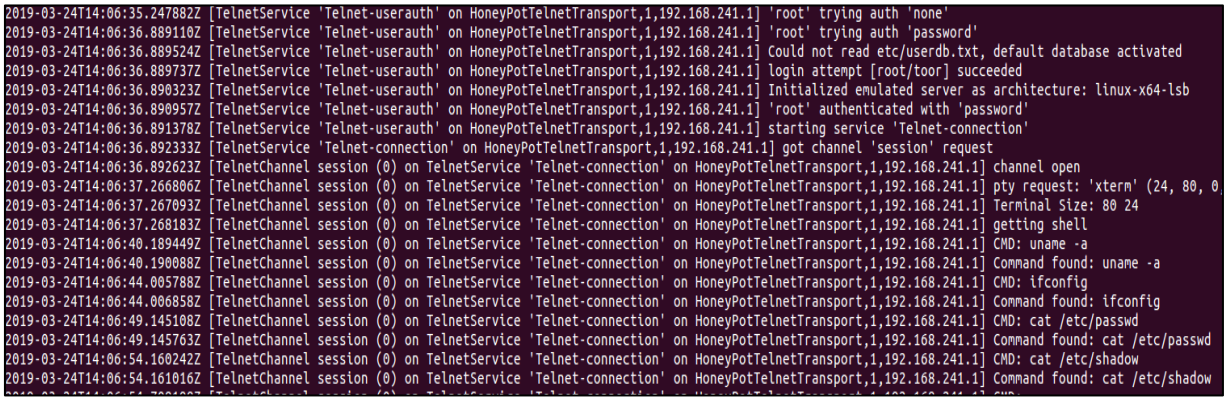


Şekil 5.11 SYN Flood Esnasında Sistem Performansı

5.4 Tuzak Sistem Kayıt Bilgileri

Hedef sisteme gerçekleştirilen tüm saldırılara ait aktiviteler Cowrie telnet tuzak sistemi tarafından kayıt altına alınmıştır. */home/cowrie/cowrie/var/log/cowrie$* dizini altında günlük olarak oluşturulan kayıt dosyaları log ve json uzantılı olarak tutulmaktadır. Kayıt dosyası içerisinde bağlantı zaman bilgisi, saldırgan IP adresi ve komut satırında çalıştırdığı komutlar yer almaktadır. Kayıt dosyasına ait içerik görüntüsü şekil 5.10'da gösterilmiştir. Ayrıca saldırganın komut satırında yaptığı tüm işlemler

/home/cowrie/cowrie/bin$ dizinde yer alan *playlog* uygulaması ile yeniden izlenebilmektedir.



Şekil 5.12 Cowrie Kayıt Dosyası İçeriği

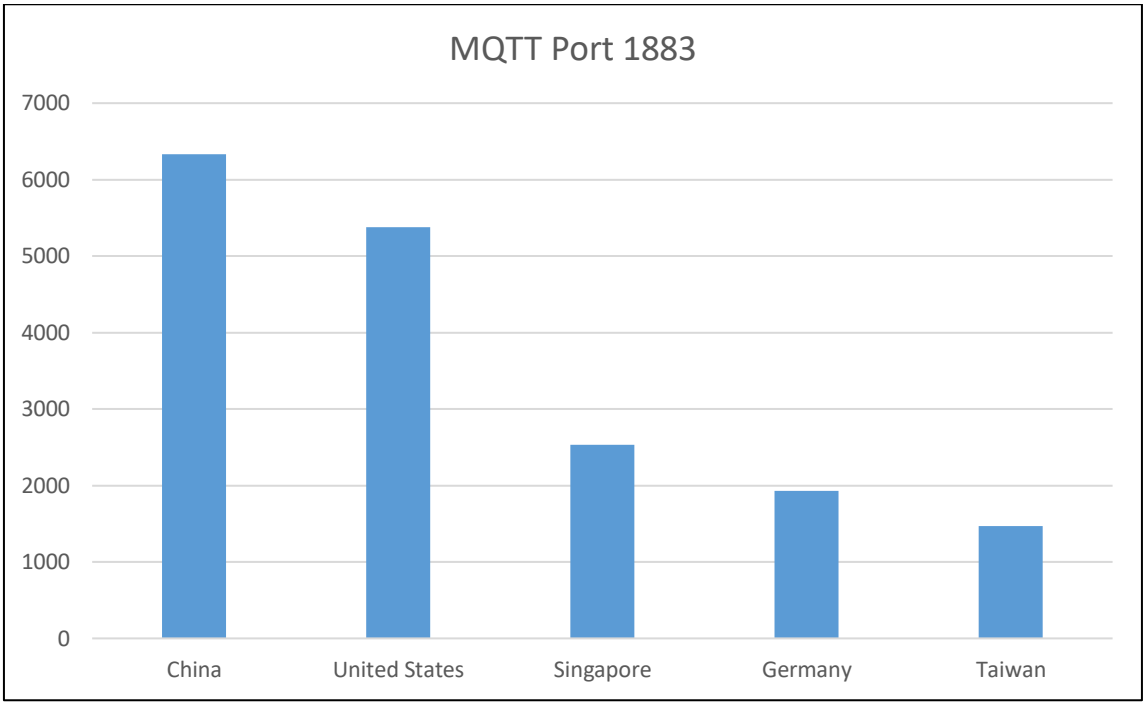
5.5 MQTT Protokolüne Yapılan Saldırılar

MQTT, kaynak kısıtlı cihazlar ve düşük bant genişliği, yüksek gecikme süresi veya güvensiz ağlar için oluşturulmuş yayınlama/abone olma mimarisine dayalı çok basit ve hafif bir mesajlaşma protokolüdür. Tasarım ilkeleri, ağ bant genişliğini ve cihaz kaynak ihtiyaçlarını en aza düşürürken aynı zamanda güvenilirliği ve teslimatın bir dereceye kadar güvencesini sağlamayı hedefler. Ayrıca bu tasarım ilkeleri, bu protokolü yaygınlaşmakta olan “makineden makineye” (M2M) veya “Nesnelerin İnterneti” teknolojileri için ideal bir iletişim altyapısı sağlamaktadır.

MQTT V3.1’de kullanıcı adı ve parola ile doğrulama yapılabilmektedir. Ağ üzerinde şifreleme, MQTT protokolünün kendisinden bağımsız olarak SSL kullanılarak gerçekleştirilebilir (SSL protokolünün hafif olmadığı ve önemli bir ek yükü getirdiğine dikkat etmek gerekir). Ayrıca protokol üzerinden gönderilen ve alınan verileri şifreleyen bir uygulama ile ek bir güvenlik uygulaması sağlanabilir, ancak bu yük artıcı ve performans düşürücü bir etkiye sahip olduğu için protokolde yerleşik değildir.

5.5.1 Bilgi Toplama

İnternette kaç tane MQTT sunucusu bulunduğunu görmek için Shodan arama motoru kullanılmıştır. Bağlantı noktası 1883 olarak yapılan kısa bir arama ile 30 binden fazla hedef sistem tespit edilmiştir. MQTT cihazlarının çoğu Çin ve ABD’de bulunmaktadır (tüm sayının yaklaşık% 40’ı). Arama sonuçları şekil 5.11’de gösterilmiştir.



Şekil 5.13 Shodan Arama Sonuçları

Ayrıca tespit edilen hedef sistemde çalışan servis bilgileri nmap aracı ile detaylı olarak tespit edilmiştir. Kali Linux üzerinde bulunan nmap aracı ile hedef sistemde çalışan MQTT broker servisine ait bilgiler toplanmıştır. Çalıştırılan komut ve elde edilen sonuç ekran görüntüsü şekil 5.12'de gösterilmiştir. Sistem üzerinde 1883 portunda mosquitto 1.4.8 broker servisinin çalıştığı tespit edilmiştir.

Şekil 5.14 MQTT Broker Bilgileri

5.5.2 Trafik Dinleme

MQTT protokolü üzerinde yerleşik olarak bir şifreleme uygulaması bulunmamaktadır. Be nedenle protokol üzerinden yapılan tüm iletişim açık metin (clear text) olarak iletilmektedir. Kimlik doğrulama tamamen isteğe bağlıdır. Kimlik doğrulama yapılsa bile, şifreleme varsayılan olarak kullanılmamaktadır. Bu nedenle iletişim üçüncü kişiler tarafından dinlenebilir ve hassas bilgiler ifşa olabilmektedir.

MQTT protokolünde kimlik doğrulama işlemi şu şekilde yapılmaktadır; İstemciler CONNECT paketiyle bir kullanıcı adı ve şifre göndererek MQTT Broker ile kimlik doğrulaması yapabilirler. CONNACK paketi, istemciden alınan bir CONNECT paketine cevaben MQTT broker tarafından gönderilen pakettir. CONNACK paket başlığı, kimlik doğrulama sonucunu temsil eden bir "dönüş kodu" (return code) alanı içerir. Tüm dönüş kodları tablo 5.3'te detaylı olarak gösterilmiştir.

Value	Return Code Response	Description
0	0x00 Connection Accepted	Connection accepted
1	0x01 Connection Refused, unacceptable protocol version	The Server does not support the level of the MQTT protocol requested by the Client
2	0x02 Connection Refused, identifier rejected	The Client identifier is correct UTF-8 but not allowed by the Server
3	0x03 Connection Refused, Server unavailable	The Network Connection has been made but the MQTT service is unavailable
4	0x04 Connection Refused, bad user name or password	The data in the user name or password is malformed
5	0x05 Connection Refused, not authorized	The Client is not authorized to connect
6-255		Reserved for future use

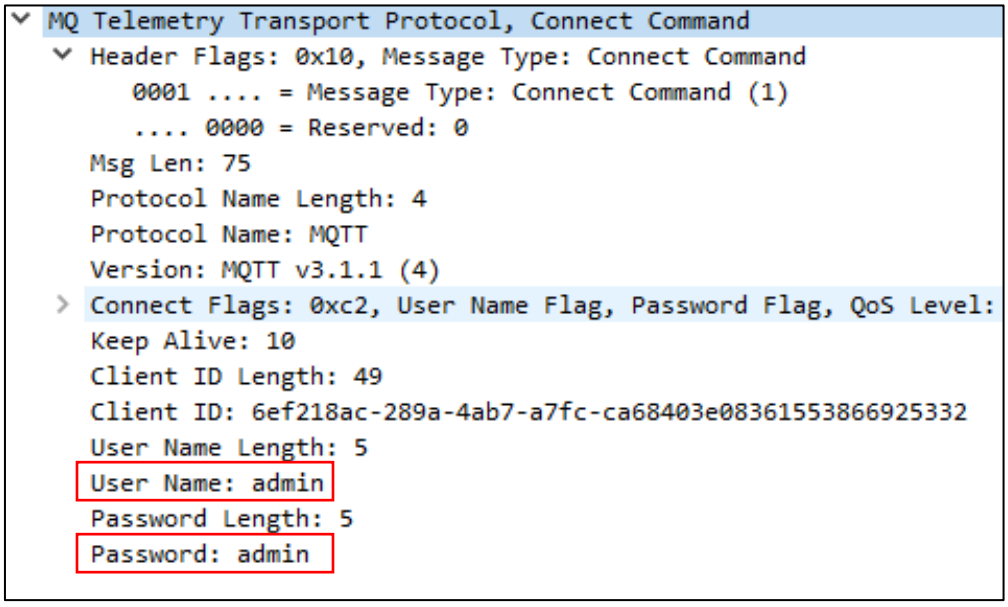
Tablo 5.3 MQTT Dönüş Kodları (CONNACK) [62]

Bu saldırıda Wireshark aracı kullanılarak mosquitto broker ile bir istemci arasında oluşan trafik dinlenmiş ve kimlik bilgileri elde edilmiştir. Python istemci tarafından gönderilen CONNECT paketine cevaben Mosquitto broker tarafından CONNACK paketi gönderilmiştir. Bu iletişim içerisinde kimlik doğrulamak için kullanılan kullanıcı adı ve parola bilgisi açık olarak elde edilmiştir. Wireshark ekran görüntüsü 5.13'te gösterilmiştir.

No.	Time	Source	Destination	Protocol	Length	Info
7	4.809551	192.168.241.1	192.168.241.141	MQTT	131	Connect Command
9	4.809764	192.168.241.141	192.168.241.1	MQTT	60	Connect Ack
11	14.814309	192.168.241.1	192.168.241.141	MQTT	56	Ping Request
12	14.814569	192.168.241.141	192.168.241.1	MQTT	60	Ping Response

Şekil 5.15 Wireshark Bağlantı Bilgileri

Paket içeriği kontrol edildiğinde kullanılan broker sürüm numarası, istemci id bilgisi, kalite seviyesi ve kimlik bilgileri açık olarak görülmektedir. Paket içerik bilgisi detayları şekil 5.14'te gösterilmiştir.



Şekil 5.16 Paket Bilgileri

5.5.3 Servis Dışı Bırakma

MQTT broker, sabit bir bant genişliği akısına dayanacak ve bağlı istemcilere gerçek zamanlı bilgi sunacak şekilde tasarlanmıştır. Broker, IoT sistemini hedef alan herhangi bir DoS saldırısının birincil hedefi olmasının yanı sıra herhangi bir IoT uygulamasının da kalbidir. Bu senaryoda Mosquitto broker kullanılan bir sisteme SYN Flood saldırısı gerçekleştirilmiştir. Hping3[63] aracı kullanılarak hedef sisteme birçok farklı kaynaktan sürekli olarak SYN bayrağı ile işaretlenmiş TCP paketleri gönderilmiştir. Kali Linux üzerinde çalıştırılan komut seti aşağıdadır.

root@kali:# hping 192.168.241.142 –S –flood –p 1883 –rand-source

Saldırı sırasında Mosquitto broker performansı mqtt-benchmark [64] aracı kullanılarak test edilmiş ve sonuçlar paylaşılmıştır. Performans testi için kullanılan komut seti aşağıda gösterilmiştir.

root@server:/# mqtt-benchmark –broker=mqtt://192.168.241.142:1883

SYN Flood saldırısı öncesi broker performansı şekil 5.15'te gösterilmiştir.

```
root@server:/# mqtt-benchmark --broker=mqtt://192.168.241.142:1883

Trying to conenct to 10 clients ...
Conencted to 10 sucessfully

        Average result for each client of the 10 clients.

        Establishing connection (sec):    0.013
        Success in publishing (msg):      100/100
        Failure in publishing (msg):      0/100
        Duraion in publishing (sec):      0.085
```

Şekil 5.17 Broker Performansı

Saldırı öncesi istemci ile bağlantı kurma süresi 0.013 saniye iken saldırı esnasında bu süre 0.0321 saniye olmuştur. Aynı zamanda mesaj yayımlama süresi 0.085 saniye iken saldırı sırasında bu süre 0.6594 saniye olmuştur. SYN Flood saldırısı sırasında yapılan performans testi sonuçları şekil 5.16'da gösterilmiştir. Yayımlama süresinde %67 oranında bir gecikme meydana gelmiştir.

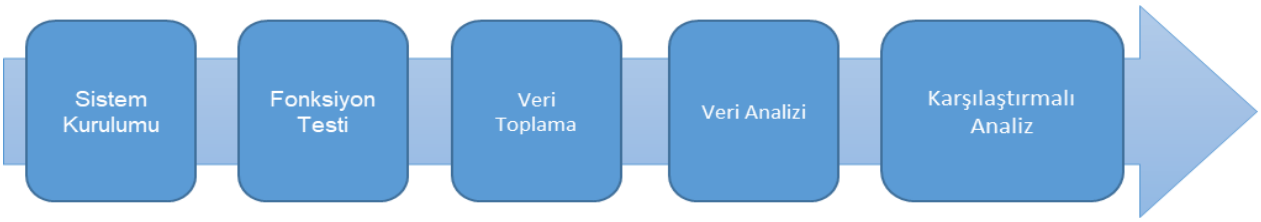
```
root@server:/# mqtt-benchmark --broker=mqtt://192.168.241.142:1883

Trying to conenct to 10 clients ...
Conencted to 10 sucessfully

        Average result for each client of the 10 clients.

        Establishing connection (sec):    0.0321
        Success in publishing (msg):      100/100
        Failure in publishing (msg):      0/100
        Duraion in publishing (sec):      0.6594
```

Şekil 5.18 SYN Flood Sırasında Broker Performansı

5.6 Örnek Senaryo

Bu bölümde yapılan çalışmanın amacı gerçek ortamda yer alan bir IP kameranın açıklıklarının belirlenmesi ve sömürülmesidir. Bu çalışma yapılırken Şekil 5.19'da yer

alan araştırma yöntemi[65] takip edilmiştir. Bu araştırma yöntemi beş adımdan oluşmaktadır; sistem kurulumu, fonksiyon testi, veri toplama, veri analizi ve karşılaştırmalı analiz.

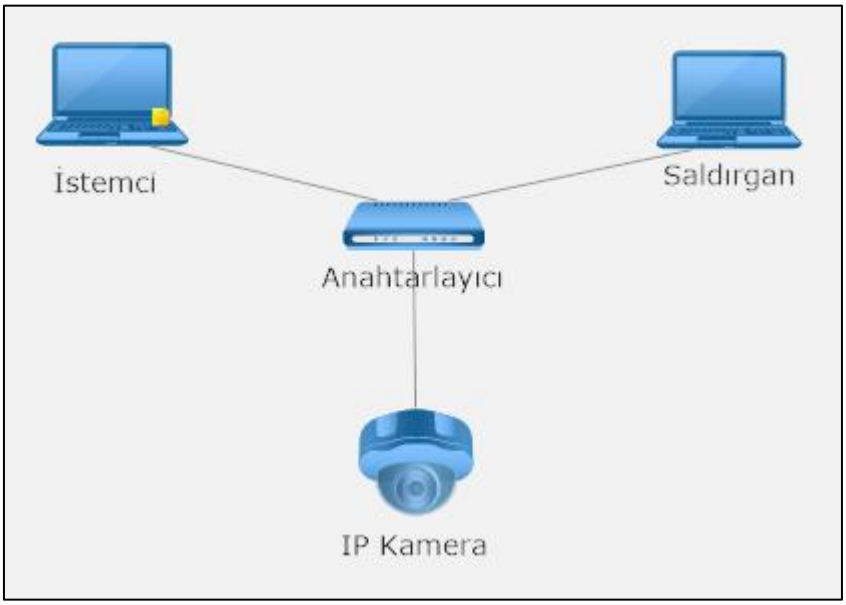


Şekil 5.19 Araştırma Yöntemi

5.6.1 Sistem Mimarisi

Çalışma için kullanılan cihazlar Şekil 5.20'de gösterilmiştir.

- Hedef IP Kamera (10.0.0.2): GeoVision GV-FD220D
- Ağ Anahtarlama Cihazı: Cisco 3550 Switch
- İstemci: HP 2QH52EA Laptop – Intel Core i7 CPU 8 GB Ram Windows 7 Professional
- Saldırgan: HP 2QH52EA Laptop – Intel Core i7 CPU 8 GB Ram Kali Linux x64 2019.2



Şekil 5.20 Sistem Topolojisi

5.6.2 Fonksiyon Testi

IP kamera sistemi ağını kurduktan sonra, kamera işlevlerinin yanı sıra tüm cihazlar arasında ağ bağlantılarını yapılandırmak ve test etmek için bir fonksiyon testi gerçekleştirilmiştir. İstemci IP adresini adres çubuğuna girerek Windows Explorer tarayıcısı üzerinden bağlanabilir veya ana bilgisayar IP adresini ve cihaz türünü seçerek kameranın DVR ara yüzüne bağlanmak için GeoVision DMMultiView istemci yazılımını kullanabilmektedir. Kamera IP adresini bulmak için GvIP Device Utility yazılımı kullanılabilir. Yapılan bağlantı bilgileri ve ekran görüntüsü Şekil 5.21'de gösterilmiştir.

Şekil 5.21 GvIP Device Utility Kullanıcı Ara Yüzü

Saldırgan tarafında Kali Linux işletim sistemi kullanılmaktadır. Aynı zamanda saldırgan sisteminin IP kameraya erişimi de bu aşamada test edilmiştir.

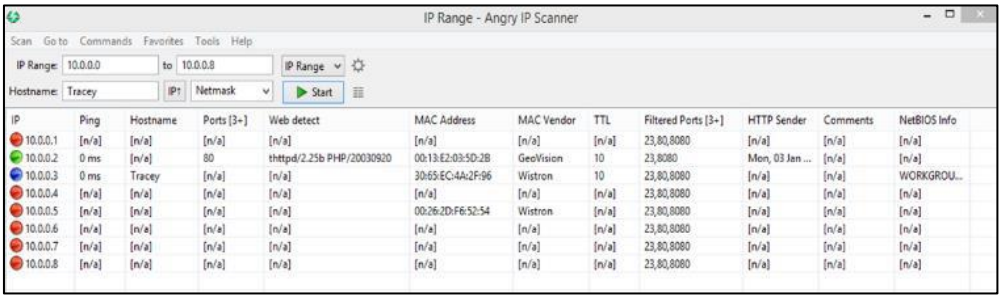
5.6.3 Veri Toplama

Fonksiyon testi sürecinde Angry IP Scanner ve WireShark araçlarının düzgün çalıştığı test edilmiştir. Her bir araç farklı özellik ve işlevde veri işleme özelliğine sahiptir. Nmap, Hydra ve Nikto araçları test edilmiştir. Bu araçlar komut satırında çalışmaktadır ve hedef sistemler hakkında bilgi toplamak için kullanılmışlardır. Toplanan veriler bu araçlar ile işlenmiş ve analiz edilmiştir.

5.6.4 Sistemin Ortam Tanımı

Angry IP Scanner ve Nmap araçları hedef sistem hakkında IP adresi, MAC (media access control address) adresi, üretici ve uygulama sürüm bilgisi gibi verileri toplamak için kullanılmıştır. Angry IP Scanner, hızlı ve hafif bir platformlar arası IP adresi ve port tarayıcıdır; herhangi bir aralıktaki IP adreslerini taramak için kullanılır. Canlı olup olmadığını kontrol etmek için her hedef IP adresine ping atıp, ardından isteğe bağlı

olarak ana bilgisayar adını çözümleyerek bağlantı noktalarından herhangi biri hakkında bilgi verir, MAC adreslerini ve üretici bilgisini elde edebilir[Şekil 5.22].



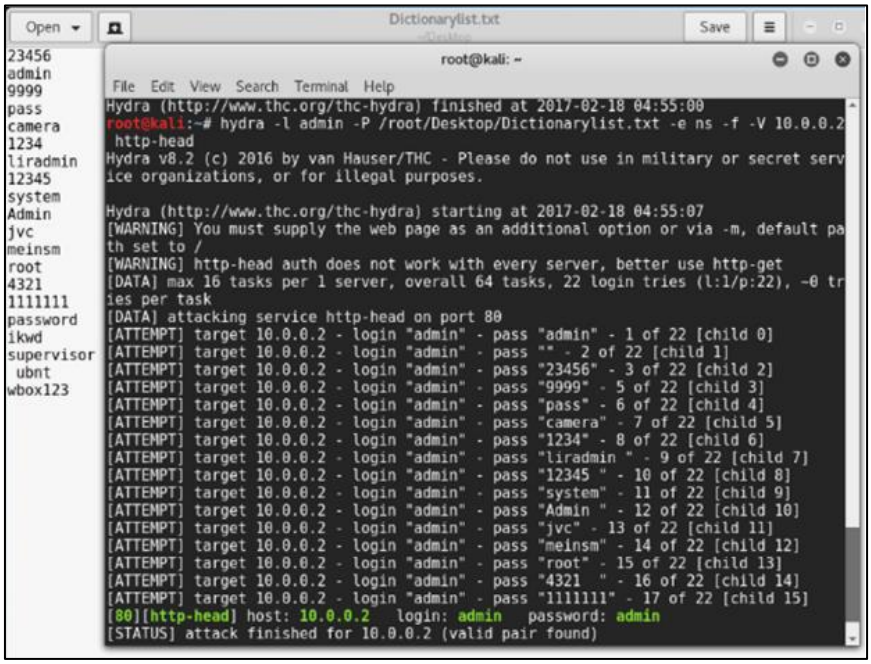
Şekil 5.22 Angry IP Scanner

Hedef sistem IP ve MAC adresi, üretici bilgisi ve açık portları tespit edilmiştir. IP adresini elde ettikten sonra daha derin bir analiz için Nmap kullanılmıştır. Ağdaki aktif bilgisayarları, sunulan hizmetleri, çalıştıkları işletim sistemini (OS), güvenlik duvarlarını ve ayrıca diğer kullanıcı özelliklerini belirlemek için ham IP paketlerini kullanır. Nmap tarama sonuçlarından, hedef IP kameranın TCP 80, 111 ve 10000 portlarının açık olduğu tespit edilmiştir. IP kamera 80 numaralı portu web ara yüzüne erişim için ve 10000 numaralı portu video aktarımı için kullanmaktadır. Saldırıyı daha ileriye taşımak için paket dinleme ve arp sahteciliği saldırıları gerçekleştirilmiştir.

Wireshark, kullanıcı adı ve şifreyi açık metin veya özet türünden yakalamak için hedef IP kamera sistemi web uygulamasında kimlik doğrulaması yapmak üzere izleme ve yakalama yapmak için kullanılmıştır. Yakalanan paketler daha sonra analiz edilmiş ve TCP paket akışları izlenerek, daha ileri saldırılar için kullanılmıştır. Dinleme sonucu iki adet MD5 özet değeri elde edilmiştir.

Hedef IP kameranın web ara yüzüne kaba kuvvet saldırısı yapmak için Hydra aracı kullanılmıştır. Ayrıca THC-Hydra olarak da adlandırılır ve birçok protokolden ve uygulamadan gelen şifrelerin şifresini çözmek için sözlük saldırısı yapabilen komut satırı tabanlı bir araçtır. Hydra ile sözlük saldırısı yapmadan önce kelime listesi oluşturulmuştur. Yaygın olarak kullanılan kullanıcı adı ve parolalar bu kelime listesinde kullanılmıştır. IP kameraya ait GeoHttpServer uygulamasının http başlık bilgisinde

çeşitli açıklıklar barındırmaktadır. Bu nedenle Hydra http-head parametresi ile kullanılmıştır.



5.23 Hydra Sözlük Saldırısı

Sonuç olarak, 22 muhtemel kullanıcı adı ve şifre üzerinde 17 deneme sonucu doğru bilgi elde edilmiştir. Elde edilen bilgilerin doğruluğu IP kamera web ara yüzünden manuel olarak denenmiş ve sonuç doğrulanmıştır.

IP kamera web servisinin açıklılarının tespit edilmesi için açık kaynak Nikto aracı kullanılmıştır. Nikto, 6700'den fazla potansiyel olarak tehlikeli dosya/program, web üzerinde 1,250'den fazla sunucunun eski sürümlerini ve 270'den fazla sunucudaki sürüme özgü sorunları kontrol etmek için kapsamlı testler gerçekleştirir. Ayrıca, birden fazla dizin dosyasının varlığı, HTTP sunucusu seçenekleri gibi sunucu yapılandırma öğelerini de denetler ve yüklü web sunucularını ve yazılımlarını belirlemeye çalışır. Yapılan tarama sonucu aşağıdaki zafiyetler tespit edilmiştir.

- *The anti-clickjacking X-Frame-Options header is not present*
- *OSVDB-3268:GET /images/?pattern=/etc/*&sort=name: Directory indexing found*
- *GET The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS*

- *OSVDB-2119: GET/shopexd.asp?catakigud=’42:VP-ASP Shopping Cart 5.0 contains multiple SQL injection vulnerabilities. CVE-2003-0560, BID-8159*
- *GET The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type*
- *OSVDB-3092: GET /htpasswd: This might be interesting*
- *OSVDB-3268: GET /tmp/: Directory indexing found*
- *OSVDB-3092: GET /tmp/: This might be interesting*
- *OSVDB-3268: GET /images/: Directory indexing found*

Çalışma sonucu elde edilen açıklık bilgileri Tablo 5.4’te gösterilmiştir.

Yazılım	**İşlevi**	**Saldırı Türü**
Fonksiyon Testi		
Windows Explorer	IP kamera web ara yüzü erişimi	Vektör
GeoVision DMMultiView	IP kamera DVR bağlantısı	Vektör
GvIP Device Utility	IP kamera yönetimi	Vektör
Gerçek Saldırı		
Angry IP Scanner	IP av MAC adreslerinin ve çeşitli sistem bilgilerinin elde edilmesi	Bilgi Toplama
WireShark	Trafik dinleme ve kayıt altına alma	Bilgi Toplama
Nmap	Hedef sistem hakkında çeşitli bilgilerin toplanması	Bilgi Toplama
Hydra (THC-Hydra)	Sözlük saldırısı	Analiz
Nikto	Web açıklık taraması	Bilgi Toplama

Tablo 5.4 Açıklık Bilgileri

5.6.5 Servis Dışı Bırakma ve Görüntü Analizi

Bu bölümde Kali Linux işletim sistemi üzerinde bulunan Hping3 aracı kullanılarak IP kameraya yoğun bir trafik gönderimi yapılmış ve kamerada oluşan olası görüntü değişimleri gözlemlenmiştir. Hping3 aracı kullanılarak yapılan saldırıda kullanılan komut seti aşağıda gösterilmiştir.

root@kali: hping 10.0.0.2 –S –flood –rand-source

Burada hedef sisteme SYN Flood olarak adlandırılan DoS saldırısı gerçekleştirilmiştir. Hedef sisteme sürekli olarak SYN bayrağı ile oluşturulmuş TCP paketleri farklı kaynak IP adreslerinden gönderilmiştir. Saldırı öncesi ve saldırı esnasında hedef IP kameradan ekran görüntüleri alınarak çevrimiçi adli inceleme araçları [66] ile incelenmiştir. Her iki görüntüye ait bilgiler Şekil 5.24'te gösterilmiştir.

File Type:	image/jpeg
Dimensions:	900x515
Color Channels:	3
File Size:	101,362 bytes
MD5:	6822308dd460c2d7bce405b70969568d
SHA1:	47c997803bc796211e8d201f28081f6ab602cfda

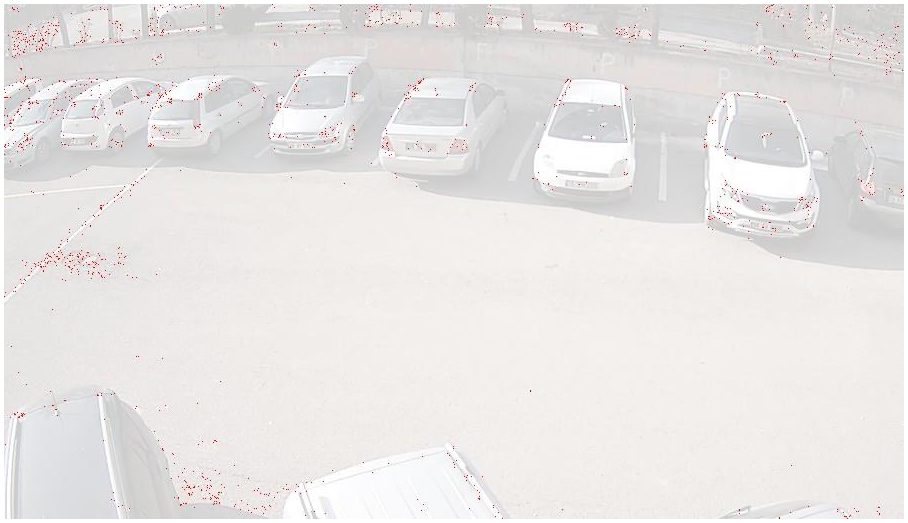
File Type:	image/jpeg
Dimensions:	900x515
Color Channels:	3
File Size:	111,089 bytes
MD5:	05611d372b9bed99facd2121a4743a99
SHA1:	524c970e95a50e82c9054fc736587e7b4d025e00

Şekil 5.24 Görüntü Bilgileri

Görüntüler ELA (Error Level Analysis) tekniği ile incelenmiş ve saldırı esnasında görüntüde değişimin meydana geldiği tespit edilmiştir. Saldırı öncesi alınan görüntünün kalitesi %92 kalite ile kaydedilirken, saldırı esnasında elde edilen görüntü %79 kalite ile kaydedilmiştir. Görüntü kalitesinde yaklaşık %14 oranında bir düşüş tespit edilmiştir. ELA analizinde gözle görülür bir değişim olduğu görülmektedir. Hata seviyesi analiz sonuçları Şekil 5.25'de paylaşılmıştır.

Şekil 5.25 Hata Seviyesi Analiz Sonuçları

Her iki görüntü çevrimiçi görüntü farkı tespit eden uygulama [67] ile analiz edilmiştir. Yapılan analiz sonucunda iki görüntü arasındaki farklar tespit edilmiştir. İki görüntü arasında tespit edilen farklılıklar kırmızı renk ile işaretlenerek Şekil 5.26'da gösterilmiştir.



Şekil 5.26 Görüntü Farkı

6. SONUÇ VE TARTIŞMA

Nesnelerin interneti sistemlerinin artarak önem kazandığı son dönemde bu kavramın kullanım alanındaki cihaz sayısında ciddi bir artış gözlenmiştir. Günlük yaşamda yer alan birçok eşya arasında yerini alan bu cihazlarda çeşitliliğin ve sayının fazla olması saldırganlar için hedefe koyulabilecek yeni bir alan anlamına gelmektedir. Bu durum bu kapsama giren sistemlerde daha fazla açıklık oluşmasına ve saldırıların artmasına sebebiyet vermiştir.

Yeterli kaynaklara sahip olmadığı için bu sistemlerde geleneksel savunma yöntemleri uygulanamamaktadır. Bundan dolayı nesnelerin interneti sistemlerinde farklı savunma ve tespit yöntemleri geliştirilmeye çalışılmaktadır. Tuzak sistemler bu yöntemlerinden bir tanesi olup; saldırgan davranışları hakkında bilgi elde etmeyi amaçlamaktadır.

Bu tez çalışmasında, IoT sistemlerde çok yoğun olarak kullanılan yönetim protokolü olan Telnet ve veri aktarım protokolü olan MQTT protokollerine yönelik benzetim ortamları oluşturularak bilinen bazı saldırı yöntemleri uygulanmıştır. Telnet protokolünün benzetimi için düşük etkileşimli Cowrie tuzak sistem kullanılmıştır. Bilgi toplama aşamasında hedef sistem ilgili port ve servis bilgileri elde edilmiştir. Kaba kuvvet saldırısı ile Telnet servisine ait kullanıcı bilgileri elde edilmiştir. Bu saldırıları engellemek için kullanılan şifre uzunluğu ve karmaşıklığının daha iyi seviyede olması

gerektiği değerlendirilmiştir. Şifrelerin en az 12 karakter uzunluğunda ve en az bir büyük harf, küçük harf, özel karakter ve rakam bulunmasının saldırıların etkisini azaltacağı öngörülmektedir. Trafik dinleme saldırısı ile Telnet protokolüne ait iletişim bilgisi içeriği elde edilmiştir. Telnet protokolünün doğasında şifreleme yapısı bulunmadığı için kritik verilerin kontrol edildiği sistemlerde kullanılmamasının daha güvenli olacağı değerlendirilmiştir. Bunun yerine daha güvenli olan SSH protokolünün ya da IPSec protokolü ile şifrelenmiş bir iletişim hattının kullanılmasını önermekteyiz. Servis dışı bırakma saldırısı yapılmış ve hedef sistemde meydana gelen kaynak tüketimleri incelenmiştir. Saldırı esnasında CPU kullanımında artış olduğu gözlemlenmiştir. Ağ trafiği kullanımında artış meydana gelmiştir. Bellek tüketiminde herhangi bir değişiklik gözlenmemiştir. Saldırılar sonucunda tuzak sistem kayıt dosyası içerisinde bağlantı zaman bilgisi, saldırgan IP adresi ve komut satırında çalıştırdığı komutlar kayıt altına alınmıştır.

MQTT protokolü için benzetim ortamı tasarlanmış ve bilinen bazı saldırılar denemiştir. Bilgi toplama aşamasında hedef sistem ilgili port ve servis bilgileri elde edilmiştir. Trafik dinleme saldırısı sonucunda hedef sistemde kullanılan broker sürüm numarası, istemci id bilgisi, kalite seviyesi ve kimlik bilgileri açık olarak elde edilmiştir. İletişimin gizliliğinin sağlanması için MQTT de yerleşik olarak bulunmayan TLS protokolünün kullanılması önerilmektedir. TLS, temel protokol olarak kullanıldığında, tüm MQTT paketleri şifrelenebilir ve bütünlükleri kontrol edilebilir. SYN Flood saldırısı ile servis dışı bırakma saldırısı benzetimi yapılmıştır. Saldırı sırasında istemcinin broker ile bağlantı kurma süresinde artış olduğu gözlemlenmiştir. Aynı zaman da mesaj yayımlama süresinde gecikme meydana gelmiştir.

Gerçek bir ortamda GeoVision GV-FD220D IP kamera üzerine çeşitli güvenlik testleri yapılmıştır. Yapılan testler sonucu hedef sistemde birçok açıklık tespit edilmiş ve hassas bilgilere erişilmiştir. Sistemin şifresi kolaylıkla kırılmış ve tam kontrolü ele geçirilmiştir. Ayrıca hedef kameraya servis dışı bırakma saldırısı yapılarak görüntü değişimleri incelenmiştir. Görüntülerde saldırı sonucu bozulmalar olduğu tespit edilmiştir. Saldırı öncesi ve sonrası görüntülerin farkı analiz edilmiştir. Bu durumun adli

bilimler açısından olay inceleme sürecinde birçok olumsuz etki oluşturabileceği değerlendirilmiştir. Tedbir amacı ile bu tip sistemlerin ayarlarının varsayılanda bırakılmaması önerilmektedir. Araştırma, IP kameraların saldırılara karşı savunmasız olduğunu ve karşı önlemlerin alınmasında daha fazla aciliyetinin olduğunu savunmaktadır.

Yapılan birçok çalışmanın aksine bu çalışmada siber güvenlik hücuma dayalı ve savunmaya dayalı olarak iki yönü ile araştırılmıştır. Yapılan çalışma sonucunda MQTT protokolünün tuzak sistem çalışmalarının çok kısıtlı olduğu ve bu alanda yeni çalışmalar yapılabileceği değerlendirilmiştir. MQTT protokolü için saldırı tespiti ve istihbaratı için tuzak sistem tasarlanabileceği öngörülmüştür. Ofansif açıdan ise IoT sistemlerde sızma testi süreçlerinin tam olarak ortaya konulamadığı ve bu yönde bir çerçevenin oluşturulabileceği değerlendirilmiştir. Son yıllarda tuzak sistem teknolojisinin gelişim gösterdiği ve kullanımında ciddi bir artış olduğu gözlenmiş ve ticari bir boyut kazandığı gözlemlenmiştir.

KAYNAKLAR

[1] Wikipedia (2017d). Internet of things. http://www.wiki-zero.net/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvSW50ZXJuZXRfb2ZfdGhpbmdzI2NpdGVfbm90ZS0yMg. (Mayıs **2018**)

[2] Weyrich, M., Ebert, C., Reference Architectures for the Internet of Things, IEEE Software, vol. 33, no. 1, pp. 112-116, **2016.**

[3] K. Rose, S. Eldridge, and L. Chapin, “The internet of things: An overview,” The Internet Society (ISOC), pp. 1–50, **2015.**

[4] J. Voas, “Demystifying the internet of things,” Computer, vol. 49, no. 6,pp. 80–83, **2016.**

[5] R. Minerva, A. Biru, and D. Rotondi, “Towards a definition of the internet of things (iot)”, IEEE Internet Initiative, no. 1, **2015.**

[6] L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. Computer networks, **2010.**

[7] Gartner, "Gartner Says the Internet of Things Installed Base Will Grow to 26 Billion Units By 2020," Gartner, 12 December 2013. https://www.gartner.com/newsroom/id/2636073. (Mayıs **2018**).

[8] Cisco, ""Connections Counter The Internet of Everything in Motion, " Cisco's The Network," Cisco, 29 July 2013. https://newsroom.cisco.com/feature-content?type=webcontent&articleId=1208342. (Mayıs **2018**).

[9] EMC & IDC, "The Internet of Things: Data from Embedded Systems Will Account for 10% of the Digital Universe by 2020 | The Digital Universe of Opportunities: Rich Data and Increasing Value of the Internet of Things," EMC & IDC, April 2014. https://www.emc.com/leadership/digital-universe/2014iview/internet-of-things.htm. (Mayıs **2018**)

[10] G. E. Moore, “Cramming More Components onto Integrated Circuits,” http://www.cs.utexas.edu/~fussell/courses/cs352h/papers/moore.pdf. (Mayıs **2018**).

[11] http://www.intel.com/content/dam/www/public/us/en/documents/corporate-information/history-intel-chips-timeline-poster.pdf. (Mayıs **2018**)

[12] A. Erdal Nesnelerin interneti ve Herşeyin İnterneti - Cisco Connect Turkey 2014, https://www.slideshare.net/CiscoTurkey/nesnelerin-interneti-ve-hereyin-nterneti-cisco-connect-turkey-2014 (Mayıs **2018**)

[13] J. Pescatore and G. Shpantzer. Securing the internet of things survey. SANS Institute, **2014**.

[14] Sicari, Sabrina, et al. ”Security, privacy and trust in Internet of Things:The road ahead.” Computer Networks 76 (**2015**)

[15] Alaba, Fadele Ayotunde, et al. ”Internet of things Security: A Survey.” Journal of Network and Computer Applications (**2017**).

[16] Pan, Yao, et al. ”Taxonomies for Reasoning About Cyber-physical Attacks in IoT-based Manufacturing Systems.” International Journal of Interactive Multimedia & Artificial Intelligence 4.3 (**2017**).

[17] Wahid, Abdul, et al. ”A Survey on Attacks, Challenges and Security Mechanisms in Wireless Sensor Network.” International Journal For Innovative Research in Science & Technology 1, **2015.**

[18] Farooq, M. U., et al. ”A critical analysis on the security concerns of internet of things (IoT).” International Journal of Computer Applications 111.7, **2015.**

[19] H. Suo, J. Wan, C. Zou, and J. Liu, “Security in the internet of things: A review,” Proc. - 2012 Int. Conf. Comput. Sci. Electron. Eng. ICCSEE 2012, vol. 3, pp. 648–651, **2012.**

[20] Hossain, Md Mahmud, Maziar Fotouhi, and Ragib Hasan. ”Towards an analysis of security issues, challenges, and open problems in the internet of things.” Services (SERVICES), 2015 IEEE World Congress on. IEEE, **2015.**

[21] Sari, Arif. ”Security issues in RFID Middleware Systems: Proposed EPC implementation for network layer attacks.” Transactions on Networks and Communications 2.5, **2014.**

[22] Hossain, Md Mahmud, Maziar Fotouhi, and Ragib Hasan. ”Towards an analysis of security issues, challenges, and open problems in the internet of things.” Services (SERVICES), 2015 IEEE World Congress on. IEEE, **2015.**

[23] Borgohain, Tuhin, Uday Kumar, and Sugata Sanyal. ”Survey of security and privacy issues of Internet of Things.” arXiv preprint arXiv:1501.02211, **2015.**

[24] Cviti´c, Ivan, Miroslav Vuji´c, and Sini˘sa Husnjak. ”Classification of security risks in the IoT environment.” 26th International DAAAM Symposium on Intelligent Manufacturing and Automation. **2016.**

[25] Ye, Ning, et al. ”An efficient authentication and access control scheme for perception layer of internet of things.” **2014.**

[26] Chen, Long. Security Management for The Internet of Things. Diss. University of Windsor (Canada), **2017.**

[27] Rolf H. Weber “Internet of Things – New security and privacy challenges” computer law & security review 26(2010) 23 – 30

[28] https://www.postscapes.com/internet-of-things-protocols/ (Ekim **2018**)

[29] IEEE802.15.4-2011, “IEEE standard for local and metropolitan area network–part 15.4: Low-rate wireless personal area networks (LR-WPAN),” in IEEE Standards, April, **2012**, pp.1-225.

[30] M. Park, “IEEE 802.11ah: sub-1-ghz license-exempt operation for the internet of things,” IEEE Communication A. Kim, F. Hekland, S. Petersen, and P. Doyle, “When hart goes wireless: Understanding and implementing the wirelesshart standard,” in IEEE International Conference on Emerging Technologies and Factory Automation, 2008, pp. 899–907. s Magazine, vol. 53, no. 9, **2015**, pp. 145-151.

[31] A DEEP DIVE INTO THE Z-WAVE BINDING,” Ekim 2017, https://www.openhabfoundation.org/documents/2017-10_Chris_Jackson_A_Deep_Dive_into_Z-Wave.pdf, (Ekim **2018**).

[32] Wikipedia. Bluetooth Low Energy, http://www.wiki-zero.net/index.php?q=ble (Haziran **2018**)

[33] Zegbee http://www.wiki-zero.net/index.php?q=zegbee (Kasım **2018**)

[34] Cetinkaya and O. Akan, “A dash7-based power metering system,” in 12th Annual IEEE Consumer Communications and Networking Conference (CCNC), **2015**, pp. 406–411.

[35] HomePlog Alliance, “HomePlug™ AV2 Technology,” 2007 http://www.homeplug.org/media/filer_public/2c/32/2c327fc8-25bb-409e-abf7-c398534c24dc/homeplug_av2_whitepaper_130909.pdf , (Kasım **2018**).

[36] A. Brandt and J. Buron, “Transmission of IPv6 Packets over ITU-T G.9959 Networks,” IETF RFC 7428, February 2015, https://www.ietf.org/rfc/rfc7428.txt, (Kasım 2018).

[37] M. Hasan, E. Hossain, and D. Niyato, “Random access for machine- to-machine communication in lte-advanced networks: issues and approaches,” in IEEE Communications Magazine, vol. 51, no. 6, **2013**, pp. 86-93.

[38] N. Sornin, M. Luis, T. Eirich, T. Kramp, and O.Hersent, “Lorawan specification,” LoRa Alliance, January 2015, https://www.loraalliance.org/portals/0/specs/LoRaWAN%20Specification%201R0.pdf, (Kasım **2018**).

[39] S. Bush, “Dect/ule connects homes for iot,” September 2015, http://www.electronicsweekly.com/news/design/communications/dect-ule-connects-homes-iot-2015-09/, (Kasım **2018**).

[40] EnOcean, “EnOcean – The World of Energy Harvesting Wireless Technology,” 2015, https://www.enocean.com/en/technology/white-papers/, (Kasım **2018**).

[41] OASIS, “Oasis advanced message queuing protocol (amqp) version 1.0,” 2012, from http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-complete-v1.0-os.pdf, (Kasım **2018**).

[42] Z. Shelby, K. Hartke, and C. Bormann, “The Constrained Application Protocol (CoAP),” IETF RFC 7252, June 2014, http://www.ietf.org/rfc/rfc7252.txt, (Kasım **2018**).

[43] P. Saint-Andre, “Extensible Messaging and Presence Protocol (XMPP): Core,” IETF RFC 6120, March 2011 https://tools.ietf.org/html/rfc6120, (Kasım **2018**).

[44] Peter, E. and Schiller, T. (2008). A practical guide to honeypots. http://www.cs.wustl.edu/~jain/cse571-09/ftp/honey.pdf. (Kasım **2018**)

[45] Mohammed H. Almeshekah and Eugene H. Spafford (2016). Cyber Security Deception. https://www.springer.com/cda/content/document/cda_downloaddocument/9783319326979-c2.pdf?SGWID=0-0-45-1579369-p179938846. (Kasım **2018**)

[46] Furche, J. and Elingehausen, R. (**1999**). Cybercop sting,getting started guide version 1.0.

[47] Schneier, B. (1999). Honeypots and the honeynet project. https://www.schneier.com/crypto-gram/archives/2001/0615.html#1.

[48] desaster (2014). Kippo - ssh honeypot. https://github.com/desaster/kippo. (Kasım **2018**)

[49] Welcome to dionaea's documentation! https://dionaea.readthedocs.io/en/latest/ (Kasım **2018**)

[50] Phype (2016). Python telnet honeypot for catching botnet binaries. https://github.com/Phype/telnet-iot-honeypot. (Aralık **2018**)

[51] Omer Erdem (2015). Honeything. https://github.com/omererdem/honeything. (Aralık **2018**)

[52] Wikipedia (2016b). Cymmetria. https://en.wikipedia.org/wiki/Cymmetria. (Aralık **2018**)

[53] Yin Minn Pa Pa, Shogo Suzuki, K. Y. T. M. T. K. C. R. (2015). Iotpot: Analysing the rise of iot compromises. http://christian-rossow.de/publications/iotpot-woot2015.pdf. (Aralık **2018**)

[54] "Mqtt version 3.1.1 plus errata 01. Edited by Andrew Banks and Rahul Gupta 10 December 2015," OASIS Standard Incorporating Approved Errata 01. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html. (Aralık **2018**)

[55] S. Tarkoma, Publish/Subscribe Systems: Design and Principles. John Wiley & Sons, **2012**, ch. 7. Distributed Publish/Subscribe.

[56] "Mqtt essentials part 5 - mqtt topics & best practices," 2015, http://hivemq.com/mqtt-essentials-part-5-mqtt-topics-best-practices.

[57] https://www.endustri40.com/haberlesme-protokollerinde-endustri-4-0-devrimi-mqtt/ (Aralık **2018**)

[58] Cowrie - https://github.com/cowrie/cowrie (Aralık **2018**)

[59]http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvVGVsbmV0 (Mart **2019**)

[60] https://www.ibm.com/downloads/cas/OAN7VKK4 (Mart **2019**)

[61] https://www.shodan.io/ (Mart **2019**)

[62] Hacking the IoT with MQTT -https://morphuslabs.com/hacking-the-iot-with-mqtt-8edaf0d07b9b (Mart **2019**)

[63] http://www.hping.org/ (Mayıs **2019**)

[64] https://www.npmjs.com/package/mqtt-benchmark (Mayıs **2019**)

[65] Bryman, A. (2012). Social research methods. Oxford: Oxford University Press.

[66] http://fotoforensics.com/analysis.php (Eylül 2019)

[67] https://online-image-comparison.com/ (Eylül 2019)

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/~~DOKTORA~~ TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
ADLİ BİLİMLER ANABİLİM DALI BAŞKANLIĞI'NA

Tarih: 28/09/2019

Tez Başlığı / Konusu: NESNELERİN İNTERNETİ SİSTEMLERİNE YAPILAN SALDIRILARIN ANALİZİ ÜZERİNE TUZAK SİSTEMLER İLE BİR DURUM ÇALIŞMASI

Yukarıda başlığı/konusu gösterilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler d) Sonuç kısımlarından oluşan toplam 79 sayfalık kısmına ilişkin, 28/09/2019 tarihinde ~~şahsım~~/tez danışmanım tarafından *Turnitin* adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 9 'dur.

Uygulanan filtrelemeler:
1- Kaynakça hariç
2- Alıntılar ~~hariç~~/dâhil
3- 5 kelimeden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Tez Çalışması Orjinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

Tarih ve İmza
30.09.2019

Adı Soyadı: Ferhat DAL
Öğrenci No: N16127614
Anabilim Dalı: Adli Bilimler
Programı: Adli Bilimler Tezli Yüksek Lisans
Statüsü: ☒ Y.Lisans ☐ Doktora ☐ Bütünleşik Dr.

DANIŞMAN ONAYI

UYGUNDUR.

Doç.Dr. Harun ARTUNER
(Unvan, Ad Soyad, İmza)

ÖZGEÇMİŞ

Adı Soyadı	: Ferhat DAL
Doğum yeri	: Yozgat
Doğum tarihi	: 26.09.1989
Medeni hali	: Evli
Yazışma adresi	: Alsancak Mah. 2192 Sk. 23/13 Etimesgut/ANKARA
Telefon	: 0545 213 4080
Elektronik posta adresi	: ferhat.dal@outlook.com
Yabancı dili	: İngilizce

EĞİTİM DURUMU

Lise	: Sorun Endüstri Meslek Lisesi Bilgisayar Yazılım
Önlisans	: Afyon Kocatepe Üniversitesi Bilgisayar Programcılığı
Lisans	: Anadolu Üniversitesi İşletme Fakültesi
Yüksek Lisans	:
Doktora	:

İş Tecrübesi

2009 - 2018 Hava Kuvvetleri Komutanlığı Siber Güvenlik Uzmanı

2019 – Devam TNB Bilişim Teknolojileri San.ve Tic. A.Ş. Sistem ve Ağ Güvenlik Uzmanı