ESTUDAM Bilişim Dergisi
Cilt 2, Sayı 2 , 22-28, 2021

Journal of ESTUDAM Information
Volume 2, Issue 2, 22-28, 2021

(Geliş Tarihi / Received Date: 13.03.2021, Kabul Tarihi/ Accepted Date: 15.04.2021)

Otonom Araçlara Yönelik Güvenlik Saldırıları

Ayça Nur KAHYA[1], Esra Nergis YOLAÇAN[2]

[1]Eskişehir Osmangazi Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, 26480, Eskişehir, ORCID No : https://orcid.org/0000-0002-6950-4421
[2]Eskişehir Osmangazi Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği.Bölümü, 26480, Eskişehir, ORCID No : https://orcid.org/0000-0002-0008-1037

Anahtar Kelimeler:
ESP32 Kamera,
DDoS Saldırıları,
Arduino,
Drone Güvenlik Saldırıları,
Otonom Araçlar

Özet: Otonom araçların (Autonomous Vehicle) ve drone'ların kullanım alanları gün geçtikçe artmaktadır. Otonom araçlar, otomatik kontrol sistemleri sayesinde insan faktörüne ihtiyaç duymadan direksiyon kontrolü, yavaşlama, hızlanma gibi manevraları gerçekleştirmektedir. Bu gibi sistemlerin temelinde birçok yazılım yer almaktadır. Sürüş seviyelerine göre kontrolde ise tamamen otonom olması veya sürücüye ait olması gibi durumlar mevcuttur. Otonom araçlar sayesinde insan hatasından kaynaklanan trafik kazaları olasılığı azalmaktadır. Diğer bir açıdan, birçok farklı teknolojiyi bir araya getiren otonom araçlar bazı güvenlik hatalarına ve saldırılarına karşı açık hale gelebilmektedir. Bu çalışmada, otonom araçlara ve drone'lara karşı gerçekleştirilen siber güvenlik saldırıları ve sonuçları ele alınmaktadır.

Security Attacks Against Autonomous Vehicles

Keywords:
ESP32 CAM,
DDoS Attacks,
Arduino,
Drone Security Attacks,
Autonomous Vehicles

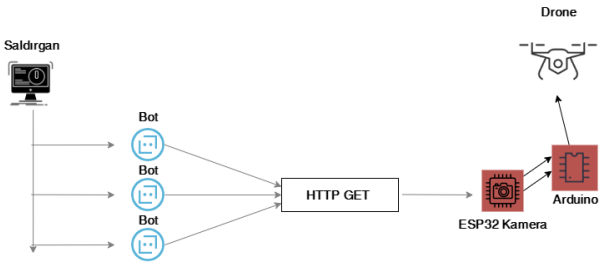
Abstract: The usage areas of autonomous vehicles and drones are increasing day by day. Autonomous vehicles enable maneuvers such as steering, deceleration and acceleration without the need for human factors through their automatic control systems. There is a lot of software on the basis of such systems. In control according to driving levels, there are situations such as being completely autonomous or belonging to the driver. Autonomous vehicles reduce the likelihood of traffic accidents caused by human error. On the other hand, autonomous vehicles that combine many different technologies can become vulnerable to some security errors and attacks. This study deals with cyber security attacks against autonomous vehicles and drones and their results.

1. GİRİŞ

Otonom araçlar (Autonomous Vehicles-AV), otomatik kontrol sistemleri sayesinde insan faktörüne ihtiyaç duymadan ya da duyulan ihtiyacı azaltan özellikleri sayesinde oldukça popüler bir konumdadır. Otonom araçlar, otomatik kontrol sistemleri sayesinde yol, trafik durumu, çevre şartları vb. algılayarak insansız sürüş sağlayabilen araçlardır. Otonom araçların ilk ortaya çıkışı 1920 ve 30'lu yıllarda gerçekleşmiştir. Ancak seyir halinde gidebilen ilk modeller 1980'li yıllarda ortaya çıkmıştır. 2014 yılında ise otonom araçların farklı sürüş seviyeleri gündeme gelmiştir. Günümüzde kullanılan otonom araçların gelişim süreci devam etmektedir.

Sunulan çalışmada otonom araçların siber güvenlik açıklarının tespiti ve iyileştirmesi için kriptografi çalışmaları incelenmiştir. İnsansız hava araçlarına ve otonom araçlara karşı yapılan siber saldırılar için alınan tedbirler kapsamında literatürdeki çalışmalar incelenerek bu alanda karşılaşılan problemlere kriptografik yaklaşımlar ile çözüm sağlamak hedeflenmiştir. Otonom araçlar ve drone'lar üzerine gerçekleştirilen saldırı türleri hemen hemen ortaktır. Önerilen çalışma ile araştırmalar sonucunda fiziksel olarak kameraları ve sensörleri etkileyebilecek saldırıların çok yaygın olduğu ortaya çıkmıştır. Şekil 1'de drone'a karşı yapılan saldırının şematik olarak gösterimi verilmiştir.

*İlgili yazar: Ayça Nur KAHYA, aycanurkahya@gmail.com



Şekil 1. Drone saldırı şeması

Otonom araçlar ve drone'lar hakkında yapılan literatür taramasına Bölüm 2'de detaylı şekilde yer verilmiştir. Bölüm 3'de önerilen yöntemin metodolojisi ve uygulamada kullanılan araçlara yer verilmiştir. Bölüm 4'te bulgular ve uygulama hakkında değerlendirmeler yapılmıştır. Bölüm 5'te ise elde edilen sonuçlar ve gelecekteki çalışmalar için öneriler yer almaktadır.

2. LİTERATUR TARAMASI

Bu bölümde, otonom araçlar ve drone'lar için siber güvenlik kapsamındaki çalışmalar iki ayrı başlık altında incelenmiştir.

2.1. Otonom Araçlarda Güvenlik

Cui ve diğerleri tarafından sunulan çalışmada [1] otonom araçları ve sürücüleri tehlikeye atabilecek arızalar; araç bileşen arızaları (Vehicle Component Failures-VF) ve altyapı arızaları (Infrastructure Failures-IF) olmak üzere ikiye ayrılmıştır. VF, donanım bileşeni, entegrasyon platformu, aktüatör, kontrolör, yazılım, araç mekanik ve iletişim sistemi arızaları olarak sıralamak mümkündür. IF ise altyapı arızaları yoldaki yaya, araç vb. hataları, trafik sinyalleri ile ilgili arızalar, güvenli olmayan trafik işaretleri arızalarını ve hatalarını içermektedir. Bu arızaların bazılarının sisteme yapılan saldırılardan kaynaklandığı bilinmektedir. Bu nedenle gömülü sensörler ile kritik sürüş koşullarını tespit etme ve yakındaki araçlarla bu bilgileri paylaşma amaçlayan birçok uygulama literatürde yapılmıştır. Sistem için kritik olan bu bilgileri paylaşmak için IVC (Inter-Vehicle Communications) ve VANET (Vehicular Ad hoc NETworks) kullanılarak ağ kurulmuştur. Bu ağların sağladığı olanaklara rağmen gizli dinleme, parazit vb. dezavantajları da mevcuttur.

Mejri ve diğerleri tarafından sunulan bir çalışmada [2] VANET'lerle ilişkili bir PKI (Public Key Infrastructure) oluşturulmasını ve bir araç ağında hızlı kimlik doğrulama yöntemi olarak dijital sertifikaların kullanılması önerilmektedir. Ultrasonik sensörler ile otonom araçların araçların güvenliğini artırmak için sinyalleri doğrulayan tek sensör tabanlı kimlik doğrulama (Physical Shift Authentication- PSA) ve çoklu sensör tutarlılık kontrolü (Multiple Sensor Consistency Check- MSCC) olmak üzere iki savunma stratejisi önerilmiştir.

Xu ve diğerleri tarafından sunulan çalışmada [3] temel olarak TDMA çizelgeleme veya çoklu kod modülasyonları aracılığıyla sensörler arası paraziti çözmek için çalışmalar yapılmıştır. Bu çalışmalar fiziksel sinyal seviyesi saldırılarıdır. Modern otomobillerde araçtaki yolcuların güvenliğini sağlaması için ECU (Electronic Control Units)'lar yüksek hızlı denetleyici alan ağı (Control Area Network- CAN) katmanına bağlanmaktadır. Bu ECU'lara örnek olarak motor kontrol modülü (Engine Control Module- ECM), acil durum fren kontrol modülü (Emergency Brake Control Module- EBCM) ve şanzıman kontrol modülü (Transmission Control Module- TCM) verilebilir. Radyo ve uzaktan kumandalı kapı kilidi alıcısı (Remote Control Door Lock Receiver- RCDLR) gibi diğer ECU'lar, düşük hızlı CAN katmanına bağlanmaktadır. Bir ağ geçidi köprüsü seçilen verileri bu iki katman arasında yönlendirebilmektedir. Bu nedenle, ağ geçidi köprüsü aracılığıyla yüksek hızlı CAN katmanına aktarılmadan önce, kötü niyetli veri paketlerinin AV'nin düşük hızlı CAN katmanına herhangi bir tespit veya şüphe olmaksızın girmesi ve daha ciddi sonuçlara yol açma olasılığı bulunmaktadır. Verilerin bütünlüğünü korumak ve doğrulamak için mesaj doğrulama kodu (Message Authentication Code- MAC) algoritmalarının olması da gerekmektedir. AV denetleyicilerinin güvenilir olduğundan emin olmak için, kimlik doğrulama sürecini desteklemek için sertifikalar, güvenlik duvarı ve biyometrik tanımlama kullanılması önerilmiştir.

Thing ve diğerleri tarafından sunulan çalışmada [4] GPS anti-parazit teknolojileri kullanılarak, parazitleri sıfırlama üzerine bilgilere değinilmiştir. Siber güvenlik önerisi olarak, VANET üzerinden araçların birbirlerine yardım etmesi mümkündür. V2IoT (Vehicle-to-IoT) iletişimlerinin de buluta taşınması söz konusu olduğunda VANET avantajlı bir çözüm olacaktır. İmza tabanlı kötü amaçlı yazılım algılama, davranış tabanlı sezgisel tabanlı kötü amaçlı yazılım algılama yöntemleri de önerilmektedir. İmza tabanlı kötü amaçlı yazılım tespiti, iki sıralı adımdan oluşmaktadır. İlk olarak, yeni kötü amaçlı yazılım tanımlanır ve her kötü amaçlı yazılımın benzersiz bir temsili veya imzası oluşturulur. İkinci olarak, her bilgisayar kötü amaçlı yazılım imzalarını alır. Ardından, verileri kötü amaçlı yazılım imzalarına karşı tarayarak dosya veya veri akışının kötü amaçlı yazılım içerip içermediğini tespit eder. Kötü amaçlı yazılım imza veri tabanında yüzlerce megabayt veri ile sonuçlanmaktadır. Bu nedenle araç üretilirken her araca büyük bir veri tabanının yüklenmesi gerekir. Sonuç olarak, bir araca zamanla ek depolama kapasitesinin eklenmesi gerekmektedir. Kötü amaçlı yazılım imzalarının sayısı arttıkça, dosyaları kötü amaçlı yazılım imzalarına karşı taramak için gereken işlem gücü miktarı da artmaktadır. Davranış tabanlı kötü amaçlı yazılım tespiti ise bir programın yürütüldüğünde ne yaptığını gözlemleyerek kötü niyetli olup olmadığını belirlemektedir. Sezgisel algılama, kötü niyetli olup olmadıklarını belirlemek için bir programın özelliklerini öğrenmek için genellikle kural tabanlı, veri madenciliği ve makine öğrenimi tekniklerini kullanmaktadır. Bunların uygulanması önemli ölçüde daha karmaşıktır ve her bir araçta çalıştırılması yoğun kaynak gerektirmektedir.

Zhang ve diğerleri tarafından sunulan çalışmada [5] kötü amaçlı yazılımlara karşı bulut hizmetleri ile alınan koruma yöntemlerine değinilmiştir. Yerleşik güvenlik ağ geçidindeki kötü amaçlı yazılım savunma işlevleri şu şekildedir. İlk olarak güvenlik ağ geçidine gelen kötü amaçlı yazılım bulaşmış dosyaları tespit eder ve bunların güvenlik ağ geçidinde yürütülmesini ve diğer araç içi cihazlara bulaşmak için güvenlik ağ geçidinden geçmesini engellemektedir. İkinci olarak, her araç içi cihazda kötü amaçlı yazılım taraması yapma ihtiyacını ortadan kaldırmak için yerleşik güvenlik ağ geçidinden araç içi cihazlara geçen trafikteki kötü amaçlı yazılımları tespit etmektedir ve engellemektedir. Üçüncü olarak, araçta bir güvenlik saldırısına işaret edebilecek şüpheli etkinlikleri algılamaktadır. Güvenilir veriler tarafından kriptografik olarak imzalanabilir ve yalnızca imzaları yerleşik kötü amaçlı yazılım savunma yöneticisi tarafından pozitif olarak doğrulandıktan sonra bir araç üzerinde yürütülmesine izin verilmektedir.

2.2. Drone'larda Güvenlik

Akram ve diğerleri tarafından sunulan çalışmada [6] İHA'lara karşı yapılan saldırılara iletişim protokolleri kullanılarak alınabilecek önlemlerden bahsedilmiştir. İHA (İnsansız Hava Araçları) filolarının yönetilmesinin birçok yolu vardır. Örneğin; tüm kararlar filo yönetim yetkilisi tarafından alınabilir (kontrol merkezi olarak da anılır). Bu, drone'ların bilgiyi yer filo yönetim sistemine potansiyel olarak gerçek zamanlı olarak iletmesini ve alınan talimatlara hızlı bir şekilde tepki vermesini gerektirmektedir. Belirli durumlar için, drone filoları büyük olasılıkla otonom hareket eder ve yerdeki bir filo yönetim kontrol sisteminden açık izin gerektirmeden uçuş sırasında kararlar almalıdır. Sınırlı hesaplama ve depolama kaynakları ile gerçek zamanlı karar gereksinimleri zorluklar getirmektedir. Bu sınırlamanın üstesinden gelmek için, drone filoları, Sürü Zekası Paradigması (Swarm Intelligence Paradigm) için tasarlanmış yapay zeka (Artificial Intelligence-AI) algoritmalarının uygulanabileceği sürüler gibi davranabilmektedir. Sürü zekası kullanan drone'ları drone sürüsü (Swarm of Drones- SoD) olarak adlandırılırken geri kalanlar da drone filoları (Fleets of Drones- FoD) olarak adlandırılmıştır. Statik bir filoda, tüm drone'lar görevin tamamı boyunca bir arada kalır. Dinamik filolarda, bireysel drone'lar, filoya gerektiği veya görevlerinin gerektirdiği şekilde girip çıkabilirler. Toplanan verilerin gizliliğini ve mahremiyetinin korunmasını sağlar. Bir drone ele geçirilirse, verilere bir düşman tarafından erişilemez.

Akram ve diğerleri tarafından sunulan çalışmada [7] toplanan verileri almak ve gizliliklerini sağlamak için bir drone ve Uçak Kablosuz Ağları (Aircraft Wireless Networks- AWNs) sensörlerinin iletişimini sağlamak için güvenilir bir kanal protokolü önermiştir. Yanlış veri enjeksiyonunu veya gizli verilerin ifşasını önlemek için İnsansız Hava Araçları (Unmanned Aerial Vehicles-UAV) veya Otonom Sürüş Ağı (Autonomous Driving Network- ADN) yalnızca yetkili cihazlarla alışveriş yaptıklarına dair güvence sağlamaktadır. Uçağın güvenliğini artırmaya yardımcı olabilir ve yerdeki uçak (Aircraft On Ground- AOG) süresini en aza indirmeye yardımcı olmaktadır.

Pigatto ve diğerleri tarafından sunulan çalışmada [8] insansız araçları için HAMSTER (HeAlthy, Mobility and Security) çözümünde kriptografik şemalar görülmektedir. HAMSTER, sağlıklı, mobilite ve güvenlik tabanlı veri iletişim mimarisini kullanmaktadır. HAMSTER, iki özel modül olan Sphere ve Nimble ile donatılmıştır. Sphere, bileşenlerin sağlığı ve modül kimlik doğrulaması ile ilgili güvenlik konularını kapsar. Nimble, her bir uygulama alanının içsel iletişimiyle güçlü bir şekilde harekete geçen senaryolarda genel hareketliliği artırmayı hedeflemektedir.

Akram ve diğerleri tarafından sunulan çalışmada [9], filoların her bir insansız hava aracına güvenli bir unsur ekleyerek yüksek saldırı potansiyeline sahip bir düşmanı ele almayı önermiştir. Saldırganın Gezici Amaca Yönelik Ağ (Mobile Ad Hoc Network- MANets), Veri İletim Ağları (Data Transmission Networks- DTN) ve Kablosuz Sensör Ağlarında (Wireless Sensor Networks- WSN) bulunanlara benzer saldırılar gerçekleştirebileceği düşünülmüştür. Ayrıca saldırgan, yönlendirme protokollerine yönelik bazı saldırılara teşebbüs etmek için sahte bir İHA oluşturabilmektedir. Örneğin: Kara Delik Saldırısı (Blackhole Attack), Seçici Yönlendirme Saldırısı (Selective Forwarding Attack), Düden Saldırısı (Sinkhole Attack), Rushing Saldırısı (Rushing Attack), Sybil Saldırısı (Sybil Attack), Solucan Deliği Saldırısı (Wormhole Attack) vb.

Akram ve diğerleri tarafından sunulan çalışmada [9] birbiriyle iletişim kuran drone'lar arasında güvenli bir kanal oluşturmak ve her bir drone'un güvenilir durumda olduğuna dair güvence sağlamak için güvenilir bir kanal protokolü önerilmiştir. Bu protokol ile, CasperFDR ve AVISPA kullanarak resmi bir analiz sağlanmıştır. İletişim cihazlarının gizlilik ve bütünlük sağlamak için bazı kriptografik sırlara sahip olması gerektiği düşünülmüştür. Bu kriptografik sırları oluşturmak için (Secure Element-SE)' ler güvenli bir kanal protokolü çalıştırmaktadır. Casper derleyicisi, bir saldırganın tanımı ve yetenekleri ile birlikte güvenlik gereksinimleri ile birlikte protokolün yüksek seviyeli bir tanımını girdi olarak almaktadır. Derleyici daha sonra açıklamayı iletişim ardışık süreçlerin (Communicating Sequential Processes- CSP) süreç cebirine çevirir. Protokolün CSP açıklaması, Arızalar-Sapma İyileştirme (Failures-Divergence Refinement-FDR) model denetleyicisi kullanılarak makine tarafından doğrulanabilmektedir.

Akram ve diğerleri tarafından sunulan bir diğer çalışmada [10] önerilen protokolü İnternet Güvenlik Protokolleri ve Uygulamalarının Otomatik Doğrulaması (Automated Validation of Internet Security Protocols and Applications-AVISPA) için protokol tanımlama dili olarak Yüksek Düzeyli Protokol Belirtim Dili'nde (High-Level Protocol Specification Language- HLPSL) yazılmıştır. Ancak DoS saldırıları tespit etme ve etkisiz hale getirme ile ilgili temel sorunlar üzerine çalışmalar devam etmektedir.

Yan ve diğerleri tarafından sunulan çalışmada [11] sürüş yönlendirmek için kullanılan sensörler, milimetre dalga

radarları, ultrasonik sensörler, kameralar üzerine olabilecek saldırıları raporlamaktadır. Bu sorunları önlemek için saldırılara karşı sensör dayanıklılığını artıracak yazılım ve donanım önlemleri açıklanmıştır. Ultrasonik sensörlere karşı ultrason, Milimetre Dalgalar (Milimeter Waves- MMW) radarlarına karşı telsiz ve kameralara karşı lazer kullanıldığında, ultrason, radyo ve lazer, hedeflenen sensörlerle fiziksel temas etmemektedir. Böylece bu saldırıları etkisiz hale getirmiştir. Tablo 1'de yazarlar tarafından tanımlanan saldırı çeşitlerinin karşılaştırması verilmiştir.

Tablo 1. Saldırıların Karşılaştırılması [11].

Saldırı Adı	Hangi Cihaza Karşı	Sonuç
Jamming Saldırısı	Sensörlere karşı	SNR (Signal Noise Ratio) oranı düşer. Algılamayı imkansız hale getirir.
Spoofing Saldırısı	Sensörlere karşı	Sensörden okunan verilerde bozulma görülür.
Blinding Saldırısı	Kameralara karşı	Bozuk görüntüler oluşur. ADAS (Advanced Driver Assistance Systems) ünitesinin kararını etkiler ve kazalara neden olabilir.

3. METODOLOJİ

Bu çalışmada, Drone üzerinde yer alan Arduino UNO ve ESP32 CAM WiFi Bluetooth Geliştirme Kartı kullanılarak test ortamı oluşturulmuştur. Ardından, Drone ve otonom araçlara yapılan DDoS saldırılarını incelemek için saldırı araçları kullanılarak kurulan bu ortamda test edilmiştir. Saldırı ortamları Wifi üzerinden olacak şekilde planlanarak sonuçları gözlemlenmiştir. Testler sonucu Çerçeve Süresi (Frame Time) ve Ara Bellek Boyu (Buffer Length) değerlerinin grafiksel karşılaştırılması sunulmuştur.

3.1. ESP32 Kamera

ESP32-CAM, Wi-Fi ve çift modlu Bluetooth ile çipli mikro denetleyiciler üzerinde bir dizi düşük güç sistemidir. ESP32 serisi, hem çift çekirdekli hem de tek çekirdekli varyasyonlarda bir Tensilica Xtensa LX6 mikroişlemci kullanır ve dahili anten anahtarları, RF balun, güç amplifikatörü, düşük gürültülü alıcı amplifikatör, filtreler ve güç yönetimi modülleri içerir [21]. ESP32-CAM, kamera ile çekilmiş görüntüleri saklamak için veya dosyaları depolamak için microSD kart yuvası bulundurmaktadır. Bu kartı programlamak için Arduino IDE kullanılmıştır. Arduino'da bu modül ile geliştirme yapmak için ESP32 eklentisi yüklenmelidir. Kullanılan Arduino'da USB bağlantısı olduğu için USB-TTL UART dönüşümüne ihtiyaç olmamıştır, fakat USB bağlantısı olmaması durumunda FTDI (Future Technology Devices International) programlama kartı da kullanılmalıdır. FTDI, programlayıcı ve dönüştürücü bir kart olarak kullanılmaktadır. Kamera ve ESP32 kartı sayesinden drone ile görüntülerin anlık olarak kaydedilmesi sağlanmıştır. ESP32 kartı Wifi ile bağlantılı çalışmaktadır.

3.2. Arduino Programlama Kartı

Arduino bir Giriş/Çıkış (Input/Output) kartı ve İşleme/Kablolama (Processing/Wiring) dilinin bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformudur. Arduino kartlarının donanımında bir adet Atmel AVR mikrodenetleyici (ATmega328, ATmega2560, ATmega32u4 gibi) ve programlama ve diğer devrelere bağlantı için gerekli yan elemanlar bulunur. Her Arduino kartında en azından bir 5 voltluk regüle entegresi ve bir 16MHz kristal osilator (bazılarında seramik rezonatör) vardır. Arduino kartlarında programlama için harici bir programlayıcıya ihtiyaç duyulmaz, çünkü karttaki mikrodenetleyiciye önceden bir bootloader programı yazılıdır [12]. Arduino Uno, Arduino programlama kartının bir çeşididir. Bu çalışmada IoT (Internet of Things) projelerinde sıklıkla kullanıldığı için Arduino UNO tercih edilmiştir.

3.3. Saldırı Araçları

DDoS saldırıları için LOIC (Low Orbit Ion Cannon), HOIC (High Orbit Ion Cannon), Netwox ve HULK araçları kullanılmıştır. Bu saldırıları araçları ile gerçekleştirilebilen saldırı tipleri Tablo 2'de verilmiştir.

3.3.1. LOIC (Low Orbit Ion Cannon)

C # ile yazılmış açık kaynaklı bir ağ stres testi ve hizmet reddi saldırı uygulamasıdır. LOIC, belirli bir ana bilgisayarın hizmetini kesintiye uğratmak amacıyla sunucuyu TCP, UDP veya HTTP paketleri ile doldurarak bir hedef sitede bir DoS (Denial od Service Attack) saldırısı (birden fazla kişi tarafından kullanıldığında, bir DDoS saldırısı) gerçekleştirmektedir. LOIC saldırıları, sistem günlüklerinde kolayca tespit edilir ve saldırı, kullanılan IP adreslerine kadar izlenmektedir [13].

3.3.2. HOIC (High Orbit Ion Cannon)

High Orbit Ion Cannon (HOIC), aynı anda 256 URL'ye kadar saldırmak için tasarlanmış açık kaynaklı bir ağ stres testi ve hizmet reddi saldırı uygulamasıdır [14]. HOIC saldırı altındaki bir bilgisayara HTTP POST ve GET istekleri göndermek için kullanılmaktadır. HOIC, birden fazla kişi tarafından koordine edildiğinde öncelikle bir hizmet reddi (DoS) saldırısı ve bir DDoS (Distributed Denial of Service Attack) saldırısı gerçekleştirir. Hedef URL'ye yapılan hizmet reddi (DoS) saldırısı, siteyi aşırı yükleme ve onu indirme girişiminde aşırı trafik göndererek gerçekleştirilir.

3.3.3. Netwox

SYN saldırısı, DDoS saldırısının bir biçimidir. Bu saldırı biçiminde bir saldırgan sistemin yasal trafiğini isteklere cevap veremeyecek duruma getirmek için yeterli sunucu kaynaklarını tüketme girişiminde bulunarak, hedef alınan sisteme ardışık SYN istekleri (SYN requests) gönderir [15]. SYN saldırısı için bu çalışmada Netwox aracı kullanılmıştır.

3.3.4. Hulk

Hulk, bir DDoS saldırı aracıdır. HTTP flood'una benzer ve birçok kaynak saldıran makineden sürekli olarak tek veya birden fazla URL talep ederek web sunucularının kaynaklarını zorlamak için tasarlanmıştır. Sunucuların eşzamanlı bağlantı sınırlarına ulaşıldığında, sunucu artık diğer kullanıcıların isteklerine yanıt verememektedir [16]. DDoS saldırısı için Hulk aracı kullanılmıştır.

Tablo 2. Saldırı Araçlarının Karşılaştırılması

Saldırı Aracı	Saldırı Tipi				
	GET Flood	TCP Flood	UDP Flood	HTTPS Flood	SYN Flood
LOIC	✓	✓	✓	✓	
HOIC	✓			✓	✓
Netwox		✓			✓
Hulk		✓		✓	✓

3.4. İzleme Aracı

Bu çalışmada, izleme aracı olarak Wireshark kullanılmıştır. Wireshark, kullanıcının ağ arabirimi denetleyicilerini rastgele moda (ağ arabirimi denetleyicisi tarafından destekleniyorsa) koymasına olanak tanır. Böylece, bu ağ arabirimi denetleyicisi MAC adresine gönderilen tek noktaya yayın trafiği de dahil, bu arabirimde görünen tüm trafiği görebilir [17]. Yapılan saldırılar Wireshark ile izlenerek saldırı anı tespit edilmiştir.

3.5. Değerlendirme Yöntemleri

Yapılan saldırıları değerlendirmek ve analiz etmek için frame time, FPS (Frame Per Second), buffer length kullanılmaktadır.

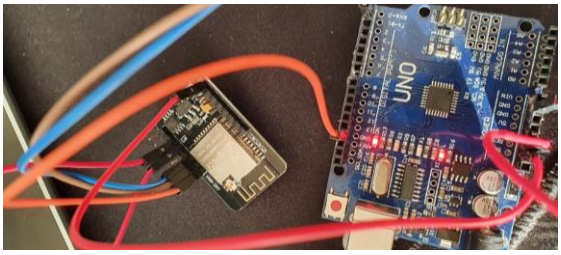
Frame time, bir kareyi yürütmek veya oluşturmak için geçen süredir. Ayrıca bazen her kare arasındaki milisaniye olarak da tanımlanmaktadır. Bir kareyi oluşturmak için ne kadar az zaman alırsa, kullanıcı girdileri o kadar hızlı işlenmektedir [18].

FPS (saniye başına kare) olarak ifade edilen kare hızı, ekranda kare adı verilen ardışık görüntülerin göründüğü frekanstır. Kare hızı, kare frekansı olarak da adlandırılabilir ve hertz cinsinden ifade edilmektedir [19].

Buffer length olarak ifade edilen arabellek boyutu, eklenti sayısı ve bilgisayarın işlem gücüne bağlıdır. Kayıt sırasında arabellek boyutu çok yüksek ayarlanırsa, oldukça uzun bir gecikme olacaktır [20].

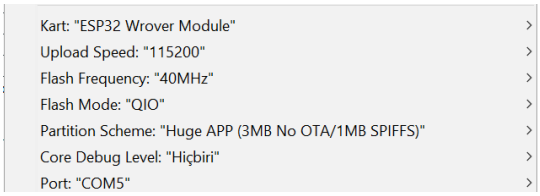
4 BULGULAR

Arduino UNO ve ESP32 ile geliştirilmiş çalışmada DDoS saldırıları ve bu saldırıların izlenmesi, çözüm önerisi bu bölümde yer almaktadır. Bu geliştirme kartı ve önceki sürümlerinde DDoS saldırılarına karşı koruma seviyesi düşük olduğu araştırmalar sonucunda tespit edilmiştir. Şekil 2'de ESP32 ve Arduino UNO bağlantıları verilmiştir. Arduino ile yazılan kodu yüklemek için GPIO 0 ve GND pinlerinin bağlı olması gerekmektedir. Yükleme tamamlandığında pinlerin bağlı olmasına gerek yoktur.



Şekil 2. ESP32 Kamera ve Arduino UNO bağlantısı

Arduino ile yapılan yazılımda "WiFi.h" ve "esp_camera.h" kütüphaneleri kullanılmıştır. ESP Wrover Module için gerekli ayarlar aşağıdaki resimde verilmiştir.

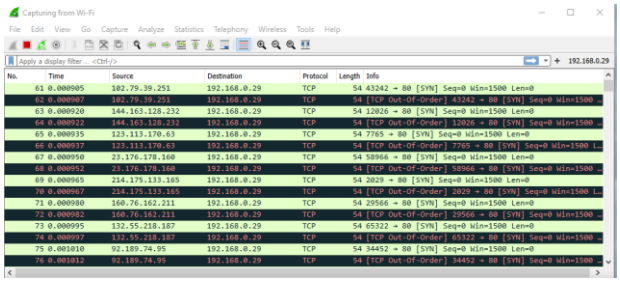


Şekil 3. ESP32 Kamera ayarları

ESP32 modülüne uygulanan saldırılar için LOIC, HOIC, Netwox ve Hulk kullanılmıştır. LOIC ile gerçekleştirilen saldırılar sonucunda kamera da yanılsamalar ve yavaşlama meydana gelmiştir. LOIC ve HOIC kullanım açısından oldukça etkili bir saldırı aracıdır.

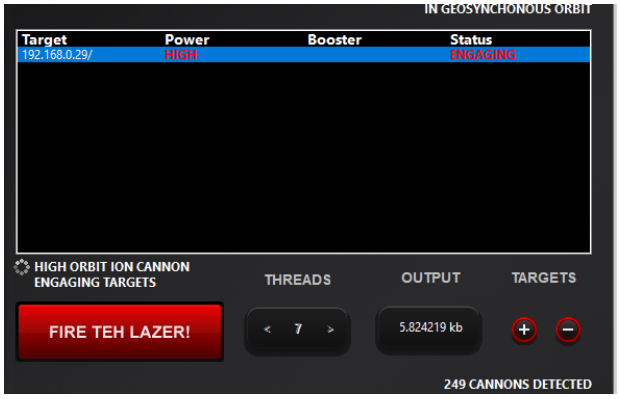
Şekil 4. LOIC DDoS saldırısı

Uygulanmış olan 4 farklı saldırı Wireshark üzerinden izlenmiştir. Şekil 4'te LOIC saldırı aracında yapılan ayarlar verilmiştir. Selected target: "IP adresi", time out:30, port:80, method: UDP, threads:99 olarak girilen değerler sonucunda requested: 71307 olarak bulunmuştur.



Şekil 5. Netwox SYN Flood saldırısı ve Wireshark üzerinden izleme

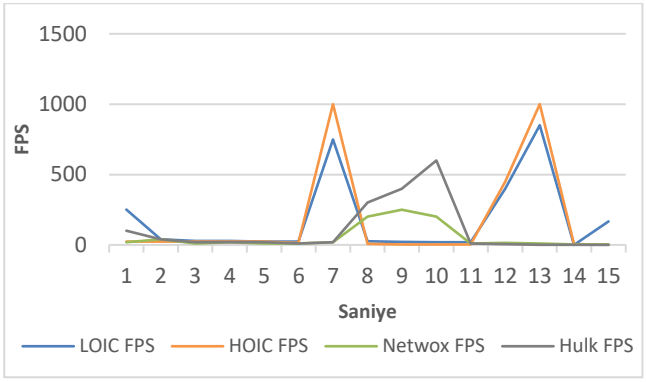
ESP32 kartına kapasitesinin üzerinde SYN paketi göndererek yeni paket alamaması sağlanarak gerçekleştirilen saldırı Şekil 5'te verilmiştir. Hulk ile gerçekleştirilen DDoS saldırısında ESP32 kamerası erişilemez hale gelmiştir. HOIC tarafından gerçekleştirilen saldırıda tek bir kullanıcının sistemi etkilemediği görülmüştür. Birden fazla kullanıcı saldırı gerçekleştirdiğinde kamera tamamen kapanmaktadır.



Şekil 6. HOIC DDoS saldırısı

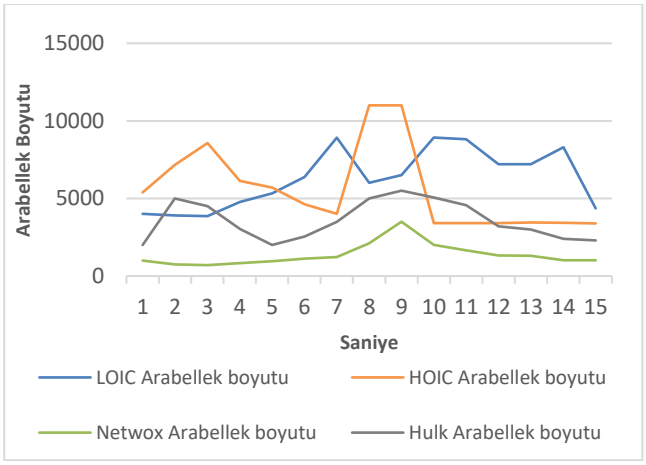
Şekil 6'da HOIC ile yapılan saldırı için girilen değerler görülmektedir. Threads:7 değeri ile saldırı yapıldığında sonucun (output) 5.8242 kb olduğu verilmiştir.

ESP32 kamerasında görüntüler MJPG formatında tercih edilmiştir. MJPG (Movition JPEG Video File), her bir karesi JPEG ile sıkıştırılmış bir video formatıdır. IP kameralarda ve web kameralarda özellikle tercih edilmektedir. Saldırı anında kamera FPS değerlerinin karşılaştırılması Şekil 7'de verilmiştir. Saldırı anında FPS değeri, normal değerin oldukça üzerindedir. Artıştan sonra ilk değer, başlangıçtaki değerler ile benzer olmaktadır. Şekil 7'de görüldüğü gibi saldırı anında FPS değerinin en yüksek olduğu saldırı HOIC saldırısıdır.



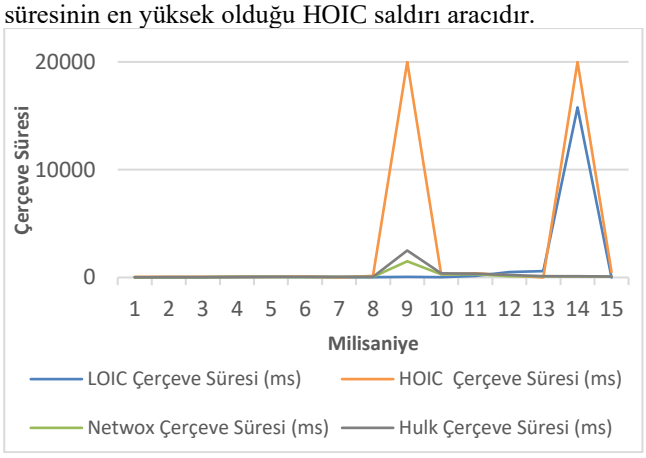
Şekil 7. Saldırı araçlarının FPS değerlerinin karşılaştırılması

Saldırı araçlarının arabellek boyutlarının karşılaştırılması Şekil 8'de verilmektedir. Saldırı sırasında HOIC, diğerlerinden daha yüksek arabellek boyutuna ulaşmaktadır. Bu nedenle HOIC saldırı aracı, diğer saldırı araçlarından daha etkilidir. İkinci en yüksek arabellek boyutuna sahip saldırı aracı ise LOIC olduğu görülmektedir.



Şekil 8. Saldırı araçlarının arabellek boyutlarının karşılaştırılması

Şekil 9'da saldırı araçlarının çerçeve sürelerinin karşılaştırılması verilmiştir. Saldırı anında çerçeve süresinin en yüksek olduğu HOIC saldırı aracıdır.



Şekil 9. Saldırı araçlarının çerçeve sürelerinin karşılaştırılması

4. TARTIŞMA VE SONUÇ

Sunulan çalışmada saldırı yöntemleri ve saldırı araçları üzerine bir inceleme yapılarak performans ve etkileri karşılaştırılmıştır. Arduino ile geliştirilen uygulamada ve testler sonucunda otonom araçların fiziksel ve yazılımsal olarak saldırılara hala açık olduğu ortaya konulmuştur. Otonom araçlarda ve drone'larda kamera modülleri yer almaktadır. Bu kamera modüllerine gerçekleştirilen saldırılar sonucunda kamera erişilemez hale gelebilmektedir. Bu durum araçlarda kullanılan diğer sensörler için de büyük açık oluşturmaktadır ve otonom araçlarda kazalara neden olabilmektedir. Drone'larda ise sistemin tamamen çökmesine yani drone'un yere düşmesi olarak görülmektedir. Fırçasız motor, ESC (Electronic Speed Controller), lipo pil, Arduino ve ESP-CAM ile birlikte yapılan testlerde saldırı anında fırçasız motor tamamen durmaktadır. Bu nedenle drone üzerinde bu sistemin kullanılması uygun bulunmamıştır. Çalışmada saldırı yöntemleri ve karşılaştırmaları grafiksel olarak açıklanmıştır. Bu çalışma ile ileride bu konu üzerine çalışma yapacak araştırmacılar için geniş ve pratik bilgiler verilmiştir.

KAYNAKÇA

[1] Cui, Jin, Liew, Lin Shen, Sabaliauskaite, Giedre, Zhou, Fengjun. "A Review on Safety Failures, Security Attacks, and Available Countermeasures for Autonomous Vehicles". Ad Hoc Networks 90 (2019): 5-6.

[2] Mejri, Mohamed Nidhal, Jalel Ben-Othman, Mohamed Hamdi. "Survey on VANET security challenges and possible cryptographic solutions". Vehicular Communications 1.2 (2014): 1-11.

[3] Xu, Wenyuan, Member, Senior, IEEE, Yan, Chen, Jia, Weibin, Ji, Xiaoyu, Liu, Jianhao. "Analyzing and Enhancing the Security of Ultrasonic Sensors for Autonomous Vehicles". IEEE Internet of Things Journal 5.6 (2018): 1-14.

[4] Thing, Vrizlynn LL, Jiaxi Wu. "Autonomous Vehicle Security: A Taxonomy of Attacks and Defences". 2016 IEEE International Conference on Internet of Things (iThings), IEEE Green Computing, Communications (GreenCom), IEEE Cyber, Physical and Social Computing (CPSCom), IEEE Smart Data (SmartData). (IEEE, 2016): 165-170.

[5] Zhang, Tao, Helder Antunes, Siddhartha Aggarwal. "Defending connected vehicles against malware: Challenges and a solution framework". IEEE Internet of Things journal1.1 (2014): 15-20.

[6] Akram, Raja Naeem, Markantonakis, Konstantinos, Mayes, Keith, Habachi, Oussama, Sauveron, Damien, Steyven, Andreas, Chaumette, Serge. "Security, Privacy and Safety Evaluation of Dynamic and Static Fleets of Drones". 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC). IEEE, (2017): 1-3.

[7] Akram, Raja Naeem, Markantonakis, Konstantinos, Mayes, Keith, Bonnefoi, Pierre-François, Sauveron, Damien, Chaumette, Serge. "A secure and trusted protocol for enhancing safety of on-ground airplanes using uavs." 2017 Integrated Communications, Navigation and Surveillance Conference (ICNS). IEEE, (2017): 3-9.

[8] Pigatto, Daniel Fernando ve arkadaşları. "The HAMSTER Data Communication Architecture for Unmanned Aerial, Ground and Aquatic Systems." Journal of Intelligent & Robotic Systems 84.1-4 (2016): 1-16.

[9] Akram, Raja Naeem ve arkadaşları. "Secure autonomous uavs fleets by using new specific embedded secure elements". 2016 IEEE Trustcom/BigDataSE/ISPA. IEEE, (2016): 608.

[10] Akram, Raja Naeem ve arkadaşları. "A secure and trusted channel protocol for uavs fleets". IFIP International Conference on Information Security Theory and Practice. Springer, Cham, (2017): 3-19.

[11] Yan, Chen, Wenyuan Xu, Jianhao Liu. "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle".nDEF CON 24.8 (2016): 1-12.

[12] https://www.robotiksistem.com/arduino_nedir_arduino_ozellikleri.html. Web.25.01.2021.

[13] https://en.wikipedia.org/wiki/Low_Orbit_Ion_Cannon. Web.25.01.2021.

[14] https://en.wikipedia.org/wiki/High_Orbit_Ion_Cannon. Web.25.01.2021.

[15] https://tr.wikipedia.org/wiki/SYN_saldırısı. Web.25.01.2021.

[16] https://kb.mazebolt.com/knowledgebase/hulk-flood/. Web.25.01.2021.

[17] https://tr.wikipedia.org/wiki/Wireshark. Web.25.01.2021.

[18] https://www.ropaku.com/what-is-frame-time-why-is-frame-time-important/. Web.25.01.2021.

[19] https://en.wikipedia.org/wiki/Frame_rate. Web.25.01.2021.

[20] https://www.sweetwater.com/sweetcare/articles/which-buffer-size-setting-should-i-use-in-my-daw/. Web.25.01.2021.

[21] https://en.wikipedia.org/wiki/ESP32. Web.5.02.2021.