Uluslararası Teknolojik Bilimler Dergisi
International Journal of Technological Sciences
UTBD
IJTS

Araştırma Makalesi/Research Article

Otonom mobil robotların güvenli veri iletimi için hibrit şifreleme yaklaşımı

Mustafa Emre Erbil[1], Ahmet Ali Süzen[2], Hilmi Cenk Bayrakçı[*1]

[1]Isparta Uygulamalı Bilimler Üniversitesi, Teknoloji Fakültesi, Mekatronik Mühendisliği Bölümü, 32000, Isparta, Türkiye
[2]Isparta Uygulamalı Bilimler Üniversitesi, Teknoloji Fakültesi, Bilgisayar Mühendisliği Bölümü, 32000, Isparta, Türkiye

Anahtar Kelimeler
Mobil otonom robotlar
Hibrit şifreleme
Veri güvenliği

Makale geçmişi:
Geliş Tarihi: 07.07.2023
Kabul Tarihi: 21.08.2023

Öz: Günümüzde teknolojik gelişmelerin sürekli ilerlemesiyle birlikte otonom ve yarı otonom araçlar; kişisel ve ticari taşımacılık, tarım, inşaat ve madencilik, araştırma ve keşif gibi birçok alanda süreçleri kolaylaştırmaktadır. Bu araçlar arasında fabrikalarda sıklıkla tercih edilen otonom mobil robotlar (Autonomous Mobile Robots, AMR) bulunmaktadır. AMR'lerin tercih edilmesinin sebepleri arasında birincisi, kullanıldığı alanlarda insan gücünün yetersiz olduğu durumlardır. İkinci olarak, bu robotlar, çalışanların sağlığını tehdit edebilecek riskli durumları ortadan kaldırabilir. Ayrıca, AMR'ler insan gücünden tasarruf sağlama potansiyeli de taşımaktadır. Bu çalışmada, kablosuz haberleşme teknolojileri kullanan AMR'lerin veri güvenliğinin hibrit bir şifreleme yöntemi kullanılarak sağlanması amaçlanmıştır. Asimetrik şifreleme yönteminin güvenlik performansı ve simetrik şifreleme yönteminin hızlı iletişim üstünlükleri, AMR'ler için her iki şifreleme yönteminin birlikte kullanılmasını zorunlu kılmıştır. Hibrit şifreleme yaklaşımında temel olarak asimetrik şifreleme yöntemi Elliptic Curve Cryptography (ECC) ile simetrik şifreleme yöntemi Advanced Encryption Standard (AES) birlikte kullanılmıştır. ECC şifreleme yönteminin temelinde oluşturulan dijital imzalama yöntemi olan ECDSA (Elliptic Curve Digital Signature Algorithm) referans alınarak kullanılmıştır. AES şifreleme yönteminde ise şifrelemenin yanında doğrulama da yapıldığı için GCM (Galois/Counter Mode) tercih edilmiştir. Bu çalışmada birbirleriyle veya kontrolcü sistemle haberleşen ve veri aktarım görevini üstlenen AMR'ler, Python programlama dili aracılığıyla ECDSA-AES/GCM hibrit şifreleme yöntemi kullanılarak şifrelenmiş ve performansı değerlendirilmiştir. Bu sayede kablosuz iletişim sırasında olası bir veri sızıntısı durumunda iletilen verinin, gizliliğinin ve bütünlüğünün korunması incelenecektir. Sonuç olarak önerilen bu yöntem ile kablosuz haberleşme teknolojileri kullanan otonom mobil robotlar için veri güvenliği açısından önemli bir adım sağlaması hedeflenmektedir.

Atıf için/To Cite: Erbil M.E. Süzen A.A. Bayrakçı H.C. Otonom mobil robotların (AMR) veri güvenliğinin hibrit şifrelem . Uluslararası Teknolojik Bilimler Dergisi, 15(2), 64-72, 2023.

A Hybrid encryption approach for secure data transmission of autonomous mobile robots

Keywords
Mobile autonomous robots
Hybrid encryption
Data security

Article history:
Received: 07.07.2023
Accepted: 21.08.2023

Abstract: Today, with the continuous advancement of technological developments, autonomous and semi-autonomous vehicles facilitate our operations in many areas such as personal and commercial transportation, agriculture, construction and mining, research and exploration. Among these vehicles are autonomous mobile robots (AMRs), which are frequently preferred in factories. One of the reasons why AMRs are preferred is that there is insufficient manpower in the areas where they are used. Secondly, these robots can eliminate risky situations that can threaten the health of workers. In addition, AMRs have the potential to save manpower. In this study, we aim to provide data security for AMRs using wireless communication technologies by using a hybrid encryption method. The security performance of asymmetric encryption and the fast communication advantages of symmetric encryption make it necessary to use both encryption methods together for AMRs. In the hybrid encryption approach, the asymmetric encryption method ECC

* İlgili yazar/Corresponding author: ahmetsuzen@isparta.edu.tr

(Elliptic Curve Cryptography) and the symmetric encryption method AES (Advanced Encryption Standard) are used together. ECDSA (Elliptic Curve Digital Signature Algorithm), the digital signature method based on the ECC encryption method, was used as a reference. In the AES encryption method, GCM (Galois/Counter Mode) is preferred since verification is performed in addition to encryption. In this study, AMRs, which communicate with each other or with the controller system and act as data senders, were encrypted using ECDSA-AES/GCM hybrid encryption method using Python programming language and their performance was evaluated. In this way, the protection of the confidentiality and integrity of the transmitted data in case of a possible data leakage during wireless communication will be examined. As a result, this proposed method aims to provide an important step in terms of data security for autonomous mobile robots using wireless communication.

1. Giriş

Teknolojik gelişmelerin sürekli ilerlemesiyle birlikte otonom ve yarı otonom araçlar birçok alanda önemli bir rol oynamaktadır. Bu araçlar insan müdahalesine gerek kalmadan görevleri yerine getirebilen ve çevresel bilgileri toplayabilen akıllı sistemlerdir [1]. AMR, endüstriyel tesislerde yaygın olan karmaşık ya da basit işlemleri gerçekleştirmek için bir konumdan diğerine hareket edebilen robotlardır [2]. Hem iç hem de dış mekân güvenlik işlerinde, tehlikeli işlerde veya tekrarlayan süreçlerde yaygın olarak kullanılabilmektedir [3]. Çoğunlukla fabrikalarda kullanılan ve askeri alanda, hastane operasyonlarında, tarımda da kullanıldığı görülen AMR'ler, otomatik malzeme taşıma, depolama ve paketleme gibi görevleri gerçekleştirerek operasyonel verimliliği artırırken insan gücünden tasarruf sağlamaktadır [4]. Bununla birlikte, AMR'lerin kablosuz haberleşme yetenekleri, veri güvenliği konusunda bazı zorlukları beraberinde getirmektedir.

Kablosuz iletişim, yetkisiz erişimlerden kaynaklanan veri sızıntılarına karşı hassas endüstriyel verilerin korunmasını gerektirmektedir. Bu nedenle, AMR'lerin kablosuz haberleşmelerinin güvenliği büyük önem taşımaktadır. Bu çalışmanın amacı, kablosuz haberleşme kullanan AMR'lerin veri güvenliğini sağlamak için bir hibrit şifreleme yöntemi kullanmasını sağlamaktır. Hibrit şifreleme, asimetrik ve simetrik şifreleme yöntemlerinin birleşimi olarak hem güvenlik hem de iletişim hızı açısından avantajlar sağlamaktadır. Asimetrik şifreleme yöntemi, güçlü güvenlik sağlama yeteneğine sahip olsa da işlemci yoğunluğu ve hesaplama süresi açısından yavaş olabilmektedir. Simetrik şifreleme yöntemi ise hızlı iletişimi sağlarken anahtar paylaşımı ve yönetimi gibi zorluklarla karşılaşabilmektedir. Hibrit şifreleme yaklaşımı, ECC ve AES yöntemlerinin birlikte kullanılmasıyla veri güvenliğini optimize etmektedir. AES, farklı bitlerde şifreleme imkanı sunan ve donanım hızlandırma yetenekleri ile veri gizliliği için hızlı şifreleme yapan bir algoritmadadır [17]. ECC, eliptik bir eğri üzerinde tanımlanan aritmetik işlemleri eğri türü ve anahtar uzunluğuna bağlı olarak şifreleyen güçlü bir asimetrik algoritmadır[14].

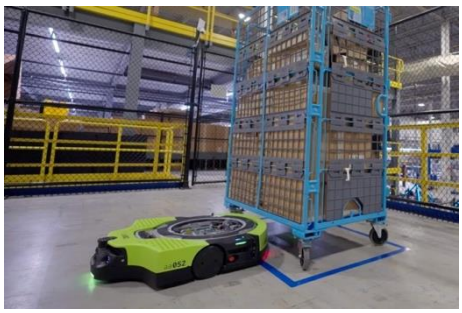
Bu çalışmada, birbirleri arası iletişim sağlayabilen ve kontrolcü sistemle haberleşebilen AMR'lerin verileri, Python programlama dilindeki PyCryptodome, ecdsa, hashlib kütüphaneleri aracılığıyla ECCDSA-AES/GCM hibrit şifreleme yöntemi kullanılarak şifrelenecektir. Bu yaklaşım, kablosuz iletişim sırasında olası veri sızıntısı durumunda, iletilen verinin gizliliğinin ve bütünlüğünün korunmasını amaçlamaktadır. Elde edilecek sonuçlar, kablosuz haberleşme teknolojileri kullanan otonom mobil robotlar için veri güvenliği açısından önemli bir adım sağlamış olacaktır.

2. Otonom Mobil Robotlar (AMR's)

AMR'ler önceden verilmiş komut olmaksızın dolaşmak ve görevleri tamamlamak için sensörler ve makineler kullanmaktadır. AMR'ler, ortamları yorumlamak ve gezinmek için gelişmiş bir sensör seti, yapay zekâ ve makine öğrenimi kullanır [5]. Çevrelerini anlayabilir ve bağımsız hareket edebilirler, bu da onları çeşitli endüstriler ve uygulamalar için önemli hale getirir. AMR'ler, üretim ve dağıtım tesisleri dahil olmak üzere çeşitli endüstrilerde ve uygulamalarda kullanılır [6]. Konveyör sistemleri gibi farklı şekil ve büyüklükteki malzemeleri bir noktadan başka bir noktaya taşıyabilirler. Ayrıca sağlık, konaklama, perakende sektörleri ve diğer birçok alanda teslimat, temizlik, envanter yönetimi gibi görevleri yerine getirmek için de kullanılırlar. AMR'lerin çok yönlülüğü ve uyarlanabilirliği, operasyonel verimlilik ve üretkenlik artışı sağlamada değerli bir araç haline gelmelerini sağlamıştır [7]. AMR'ler, artan verimlilik, üretkenlik ve güvenlik gibi çeşitli avantajlar sunar. Tekrarlayan, tehlikeli veya yüksek hassasiyet gerektiren görevleri yerine getirerek çalışanların daha karmaşık görevlere odaklanmalarını sağlarlar. Ayrıca, çalışanları riske atmadan tehlikeli ortamlarda çalışabilmesi, işyeri yaralanmaları ve kazaları riskini azaltmaktadır. AMR'ler için hareket kontrolörleri, otonom bir makine oluşturmak için gereken dayanıklılığı ve verimliliği sağlar [8]. AMR'lerde kullanılan ileri teknoloji, uygun maliyetli ve verimli bir çözüm sunarak çeşitli endüstriler ve

uygulamalar için değerli bir katkı sağlayarak çeşitli görevleri yerine getirir. AMR'ler, otomasyonun sağladığı hassasiyet ve doğrulukla iş hatalarını azaltır ve ürün kalitesini artırır. Ek olarak, AMR'ler endüstriyel sektörlerde yaygın olarak kullanılmaktadır. Sağlık sektöründen, perakende sektörüne, envanter yönetiminden, tehlikeli malzemelerin taşınmasına kadar çeşitli alanlarda fayda sağlarlar. İmalat sektöründe, AMR'ler genellikle malzeme taşıma, üretim, paketleme ve envanter takibi gibi görevlerde kullanılır. Perakende sektöründe, AMR'ler genellikle depolama, toplama ve paketleme görevlerinde kullanılır. Sağlık sektöründe (Şekil 1a), AMR'ler genellikle ilaç dağıtımı, hastane envanter yönetimi ve bazen hasta bakımı gibi görevlerde kullanılır. Tüm bu uygulamalar, AMR'lerin endüstriyel sektörlerdeki çeşitliliğini ve esnekliğini gösterir (Şekil 1b).

Şekil 1. (a) Hastanelerde kullanılan, hastalara ilaç, yiyecek gibi önemli ihtiyaçları taşımak için kullanılan AMR



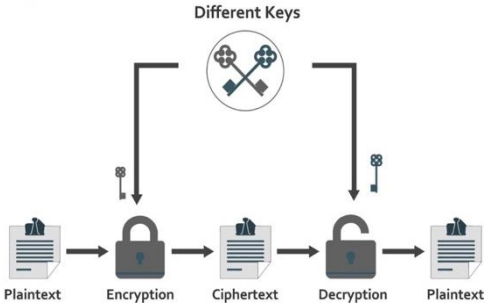
Şekil 1. (b) Koli, paket taşımacılığı yapmak için kullanılan AMR

3. Şifreleme Yöntemleri

Şifreleme, bilgiyi koruma ve veri güvenliğini sağlama süreci olup, bilginin erişilemez veya anlaşılmaz hale getirilmesi anlamına gelir. Şifreleme işlemi, bir anahtar veya bir dizi anahtar kullanarak bir mesajı veya bilgiyi okunabilir bir format (plaintext) 'den okunamaz bir formata (ciphertext) dönüştürür [11]. Şifreleme teknikleri, bilgi güvenliği ve veri gizliliği açısından son derece önemlidir. Şifreleme yöntemleri, genel olarak, asimetrik ve simetrik şifreleme olarak iki ayrı başlık altında toplanır.

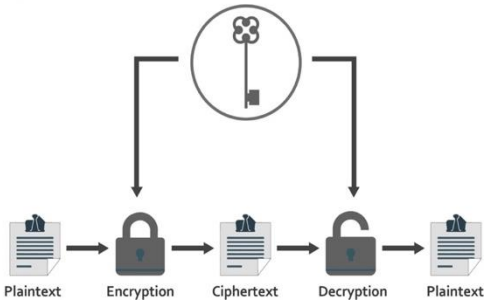
2.1. Asimetrik ve simetrik şifreleme

Asimetrik ve simetrik şifreleme dijital dünyada bilgi ve verileri korumak için kullanılan iki ana yöntemdir. Asimetrik şifreleme, genellikle "public key" (açık anahtar) şifrelemesi olarak bilinir ve iki anahtar (bir özel, bir açık) kullanılır (Şekil 2). Bilgiyi şifrelemek için açık anahtar kullanılır ve şifreli bilgiyi çözmek için özel anahtar kullanılır. Özel anahtar sadece bilgiyi çözecek olan taraf tarafından bilinir. Bu yöntem bilgilerin güvenli bir şekilde aktarılmasını sağlamakla beraber bu işlemler için daha fazla işlem gücü gerektirir. Bu yöntem, birçok modern güvenlik protokolü ve uygulaması tarafından kullanılır ve bilgiyi güvenli bir şekilde aktarma yeteneği nedeniyle popülerdir [12].



Şekil 2. Asimetrik şifreleme şeması

Aynı anahtarın hem şifreleme hem de şifre çözme işlemleri için kullanıldığı yöntem ise Simetrik Şifreleme yöntemidir (Şekil 3). Bu yöntem, asimetrik şifrelemeye göre daha hızlıdır, ancak anahtarın güvenli bir şekilde aktarılması gerekliliği nedeniyle riskler taşır. Simetrik şifrelemenin hızlı doğası, büyük miktarda verinin hızlı bir şekilde şifrelenmesi gereken durumlar için idealdir [13].



Şekil 3. Simetrik şifreleme şeması

2.2. Eliptik Eğri Şifreleme (ECC)

ECC dijital imzalar ve anahtar değişim protokolleri gibi uygulamaları hedefleyen asimetrik şifreleme metodolojilerinin bir türüdür. ECC, eliptik bir eğri üzerinde tanımlanan aritmetik işlemleri temel alır. ECC'nin kullandığı eliptik eğri Denklem 1 ile ifade edilebilir [14].

$$y^2 = x^3 + ax + b \pmod{p} \quad (1)$$

Burada a ve b, belirli parametreleri, p ise bir asal sayıyı belirtir. ECC'nin temel prensibi, bir özel anahtar (private key) ve onunla ilişkili bir genel anahtar (public key) oluşturmaktır. ECC'nin etkili özelliklerinden biri daha düşük anahtar uzunluklarına rağmen en az asimetrik şifreleme yöntemleri kadar güvenlik sunabilmesidir. Örneğin, 256-bit bir ECC anahtarı, RSA'nın 3072-bit anahtarına karşı eşdeğer güvenlik sağlayabilir. Bu özellik ECC'nin daha düşük hesaplama ve hafıza gereksinimleri ile işlem gücü ve enerji kısıtlamaları olan sistemlerde büyük avantaj sağlar [15]. ECC'nin genel güvenlik seviyesi, seçilen eğriye ve özel anahtarın boyutuna bağlıdır. Genellikle daha büyük anahtar boyutları daha güçlü güvenlik sağlar, ancak bu da daha fazla hesaplama kapasitesi gerektirir. ECC'nin sunduğu yüksek güvenlik ve performans verimliliği nedeniyle bilgi güvenliği kullanan süreçlerde yaygın tercih edilmektedir.

2.3. ECDSA (Elliptic Curve Digital Signature Algorithm)

ECDSA**,** bir mesajın doğruluğunu ve bütünlüğünü kanıtlamak için kullanılan bir sayısal imza algoritmasıdır. ECDSA mimarisi ECC teknikleri olan karakteristik p ve bir taban noktası G ile ayrık bir uzay Fq üzerinde tanımlanmış bir eliptik eğri E Etki alanı parametreleri üzerine kurulur [16]. Özellikle sunucu-clint arasında veri gizliliğinin sağlanması için kullanılan TLS/SSL'de ECDSA teknikleri kullanılır.

2.4. Advanced Encryption Standard(AES)

AES, simetrik bir blok şifreleme algoritmasıdır. AES, 128-bitlik bloklar halinde verileri şifreler ve bu nedenle bir blok şifre olarak adlandırılır. AES 128, 192 veya 256 bit uzunluğunda anahtarlarla çalışabilir. Bu seçenekler AES'in farklı güvenlik gereksinimlerine sahip uygulamalar için esnek bir seçenek olmasını sağlamaktadır [17]. Aynı zamanda AES, donanım hızlandırma avantajına sahiptir. Bu avantajı sayesinde çok büyük miktarda verinin hızlı bir şekilde şifrelenmesini ve şifresinin çözülmesini sağlar [17-18]. AES, güçlü güvenlik özellikleri ve geniş kabul görmesi nedeniyle veri gizliliği ihtiyacının olduğu birçok sektörde kullanılmaktadır.

2.5. Advanced Encryption Standard/Counter Mode (AES/GCM)

AES/GCM, son zamanlarda yaygın bir şekilde kullanılan simetrik şifreleme yöntemidir. AES/GCM, özellikle iletişim kanallarında veri bütünlüğünü ve güvenliğini sağlamak amacıyla tasarlanmıştır. Hem verilerin şifrelenmesi hem de kimlik doğrulama için kullanılır. Böylece her iki işlemi de tek bir adımda gerçekleştirir. GCM, gelişmiş bir hız ve güvenlik seviyesi sunar. Ayrıca paralel işlem kapasitesine sahip olması nedeniyle yüksek performanslı sistemlerde idealdir. GCM yaygın olarak ağ trafiğinin güvenliği, disk şifrelemesi veya dosya şifrelemesi olmak üzere bir dizi farklı uygulama için idealdir. Ayrıca TLS ve IPsec gibi birçok güvenlik protokolü standarttı AES/GCM'yi bir şifreleme seçeneği olarak kabul etmektedir [19].

4. Materyal ve Metot

4.1. Hibrit Şifreleme Yaklaşımı

Hibrit şifreleme, simetrik ve asimetrik şifreleme tekniklerinin bir arada kullanıldığı bir güvenlik stratejisidir. Bu yaklaşım hem simetrik hem de asimetrik şifreleme tekniklerinin avantajlarını birleştirerek, güçlü bir güvenlik çözümü sunar. ECC ve AES, hibrit bir şifreleme yaklaşımında birlikte kullanılabilir. Simetrik şifreleme yöntemi ECC kullanımı olarak ECDSA, asimetrik şifreleme yöntemi AES kullanımı olarak AES/GCM kullanılır. Hibrit şifreleme yaklaşımının ana prensibi oldukça basittir. Öncelikle, veri aktarımını başlatan taraf, rastgele bir simetrik anahtar oluşturur. Bu anahtar, veriyi şifrelemek ve çözmek için kullanılır. Daha sonra, bu simetrik anahtar, alıcının açık anahtarı ile şifrelenir. Alıcı, kendi özel anahtarı ile şifrelenmiş anahtarı çözer ve şifrelenmiş veriyi çözmek için simetrik anahtarı kullanır [20]. Bu yaklaşımın en önemli avantajı, simetrik ve asimetrik şifrelemenin birleştirilmesi ile güvenlik ve hız arasında etkili bir denge kurabilmesidir. ECDSA, güvenli bir anahtar dağıtımı sağlarken, AES/GCM, hızlı veri şifrelemesi, çözmek ve kimlik doğrulanmasını sağlar. Bu durum, özellikle büyük miktarda verinin aktarılması gereken durumlar için idealdir [21]. ECC'nin bir diğer önemli avantajı, kısa anahtar boyutları kullanırken güvenliğin sağlanabilmesidir. Bu, özellikle bant genişliği veya depolama alanı kısıtlı olan durumlar için yararlıdır. Öte yandan, AES/GCM, hem şifreleme hem de kimlik doğrulama özelliğine sahip olması nedeniyle, iletişimin bütünlüğünü ve kimlik doğrulamasını sağlar [22]. Bu yaklaşım, özellikle gizlilik ve bütünlük gerektiren birçok modern uygulamada değerli olabilir. Bu uygulamalar arasında bulut bilişim, e-sağlık hizmetleri, mobil uygulamalar, Internet of Things (IoT) cihazları ve daha fazlası bulunabilir [23].

4.2. Kablosuz Haberleşme ve Veri Güvenliği

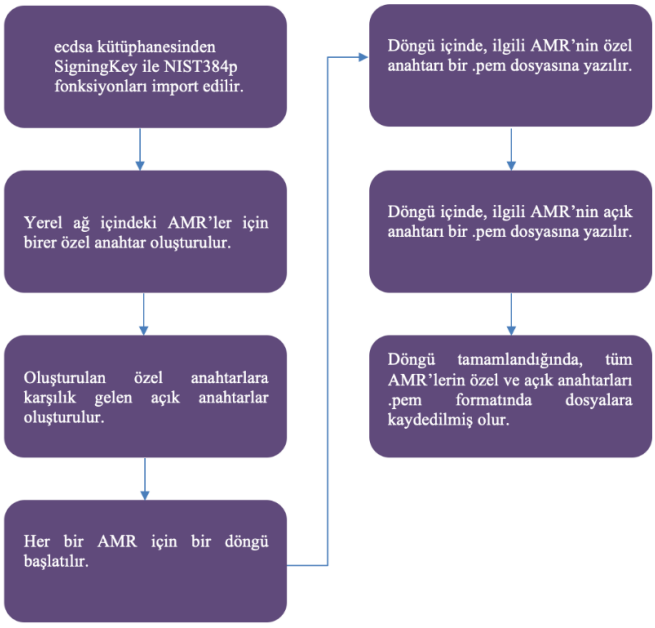
Kablosuz haberleşme, günümüzde yaygın olarak kullanılan mobil cihazlar, sensör ağları ve Internet of Things (IoT) cihazları gibi kablosuz ağlar üzerinden gerçekleştirilen iletişimi ifade etmektedir. Kablosuz haberleşme, bir dizi güvenlik zorluğuyla karşı karşıya kalmaktadır. Verilerin kablosuz ortamda iletilmesi,

fiziksel olarak korunmasız bir ortamda gerçekleştiği için saldırılara karşı daha savunmasız hale geldiği için, verilerin güvenliği ve gizliliği kablosuz haberleşme sistemlerinde büyük önem taşır [24]. Veri güvenliğini sağlamak için şifreleme yöntemleri kablosuz haberleşme sistemlerinde yaygın olarak kullanılır. Şifreleme, verileri anlaşılabilir halden şifreli bir forma dönüştürerek izinsiz erişime karşı korur. Kablosuz haberleşme sistemlerinde kullanılan şifreleme yöntemleri, güvenlik gereksinimlerini karşılamak ve verilerin bütünlüğünü, gizliliğini ve kimlik doğruluğunu sağlamak amacıyla tasarlanmıştır. Hibrit şifreleme, kablosuz haberleşme sistemlerinde yaygın olarak tercih edilen bir güvenlik stratejisidir. Bu yöntemde, simetrik ve asimetrik şifreleme algoritmaları bir arada kullanılır ve her bir algoritmanın avantajları birleştirilerek daha güçlü bir güvenlik çözümü elde edilir [25]. Asimetrik şifreleme anahtar değişimi ve kimlik doğrulama gibi işlemler için kullanılırken, simetrik şifreleme veri şifreleme ve şifre çözme için kullanılır [26]. Bu şekilde, hibrit şifreleme yaklaşımı, güvenli bir anahtar değişimi sağlarken veri şifrelemesi için hızlı ve etkili bir çözüm sunar. Kablosuz haberleşme sistemlerinde hibrit şifreleme yönteminin kullanılması, veri güvenliği konusunda önemli avantajlar sağlar. Asimetrik şifreleme algoritmaları, güvenli bir anahtar değişimi sağlayarak veri aktarımı sırasında izinsiz erişim riskini azaltır. Simetrik şifreleme algoritmaları ise hızlı veri şifrelemesi ve şifre çözme sunarak yüksek performanslı veri aktarımı sağlar. Bu nedenle, hibrit şifreleme yaklaşımı, kablosuz haberleşme sistemlerinde güvenlik ve performans arasında bir denge sağlar [27].

4.2. Python ile Hibrit Şifreleme Uygulaması

Bu çalışmada önerilen hibrit şifreleme yaklaşımı Python dili kullanarak geliştirilmiştir. Python çok amaçlı bir programlama dilidir ve ayrıca güçlü kütüphaneleri sayesinde şifreleme uygulamaları için popüler bir seçimdir. Çalışmada PyCryptodome kriptoloji kütüphanesinden *Crypto.Cipher* ve *Crypto.Random* modülleri kullanılmıştır. *Crypto.Cipher* modülü simetrik şifreleme için, *Crypto.Random* modülü ise rastgele veri üretme işlevleri için kullanılmaktadır. Crypto.Cipher modülünde, AES parametresi kullanılmıştır. *Crypto.Random* modülünde ise *get_random_bytes* parametresi kullanılmıştır [28]. Eliptik eğri dijital imza algoritması (ECDSA) kullanarak dijital imzalar oluşturulmasını ve doğrulamasını sağlamak için, ecdsa kütüphanesi kullanılmıştır. Ecdsa kütüphanesinden, özel anahtarları oluşturmak ve doğrulamak için SigningKey sınıfı ve 384 bitlik anahtar uzunluğuna sahip olması için NIST384p eğrisi kullanılmıştır [29]. National Institute of Standards and Technology (NIST) tarafından belirlenen 384 bitlik anahtar uzunluğu, oldukça güvenilir sayılmaktadır [30]. Ayrıca çeşitli kripto grafik hash işlevlerini sunan hashlib kütüphanesi kullanılmıştır. Hashlib kütüphanesinden, bir anahtar türetmek için SHA-256 hash işlevi kullanılmıştır. Anahtar oluşturma, şifreleme ve çözme adında 2 adet ayrı Python kod bloğu oluşturulmuştur. Hibrit şifrelemenin başarılı bir şekilde yapılması için tüm AMR'lerin açık anahtarları (public key) birbirleriyle paylaştırılır. Gizli anahtarlar (private key), her AMR'de sadece kendisinin gizli anahtarı olacak şekilde paylaştırılır. ECDSA ile oluşturulan açık anahtarlar AES/GCM aracılığıyla şifrelenip doğrulanır ve hedef AMR'ye şifrelenmiş veri iletilir. Şifrelenmiş veriyi teslim alan AMR, ECDSA ile oluşturulan gizli anahtarıyla kendisine gönderilen şifrelenmiş veriyi çözer. Böylelikle hibrit şifreleme ile veri transferi başarılı bir şekilde gerçekleşmiş olur. Bu sayede iletilen veriler aynı ağdaki AMR'ler haricinde hiçbir sistem tarafından okunamaz. Olası verilere yönelik bir siber saldırıda, saldırgan şifrelenmiş veriye ulaşsa bile, kendisinde gizli anahtar olmadığı için, şifreyi çözemez ve veriye ulaşamaz.

Hibrit şifreleme işleminin tam olarak gerçekleşmesi için, her AMR'de; .pem uzantılı kendisinin gizli anahtarı ve diğer AMR'lerin .pem uzantılı açık anahtarı bulunması gerekmektedir. Bunun için her AMR'de anahtar oluşturma algoritmasının tek seferlik aktif edilmesi gerekmektedir. Ayrıyeten şifreleme ve veri çözme algoritmalarının her AMR'de bulunması gerekmektedir. Python dili ile yazılan anahtar oluşturma algoritması akış diyagramı olarak Şekil 4'te verilmiştir.



Şekil 4. Anahtar çifti oluşturma algoritması akış diyagramı

İletişimde güvenli şifreleme için kullanılacak algoritması geliştirilme adımları aşağıdaki gibi verilmiştir.

Adım 1:
- Şifreleme işlemleri için gereken AES ve get_random_bytes fonksiyonları Crypto.Cipher ve Crypto.Random modüllerinden içe aktarılır.
- ECDSA işlemleri için gerekli olan SigningKey ve NIST384p fonksiyonları ecdsa kütüphanesinden içe aktarılır
- Hash işlemleri için hashlib modülü içe aktarılır.

Adım 2:
- Gönderici AMR ile aynı ağda bulunan alıcı AMR/AMR'ler için birer özel anahtar .pem dosyalarından okunur ve bir Python sözlüğünde saklanır.
- Oluşturulan her bir özel anahtardan karşılık gelen açık anahtarlar elde edilir ve ayrı bir Python sözlüğünde saklanır.

Adım 3:
- İlk olarak, gönderici ve alıcı/alıcılar için bir döngü başlatılır.
- Gönderici ve alıcı/alıcıların anahtarları kullanılarak ortak bir anahtar hesaplanır.
- Bu ortak anahtar, bir SHA-256 hash işleminden geçirilerek AES şifrelemesi için kullanılan anahtar haline getirilir.
- Veri, bu anahtar kullanılarak AES şifrelemesi ile şifrelenir.
- Şifrelenmiş veri ve şifreleme bilgileri bir "tuple" olarak saklanır

Adım 4:
- Şifrelenmiş veri, her bir alıcı için ayrı bir .txt dosyasına yazılarak gönderilmeye hazır hale getirilir.

Şifre çözme algoritması aşağıda verilen adımlar kapsamında geliştirilmiştir.

Adım 1:
- Şifreleme işlemleri için gereken AES ve get_random_bytes fonksiyonları Crypto.Cipher ve Crypto.Random modüllerinden içe aktarılır.
- ECDSA işlemleri için gerekli olan SigningKey ve NIST384p fonksiyonları ecdsa kütüphanesinden içe aktarılır
- Hash işlemleri için hashlib modülü içe aktarılır.

Adım 2:
- Gönderici AMR ile aynı ağda bulunan alıcı AMR/AMR'ler için birer özel anahtar .pem dosyalarından okunur ve bir Python sözlüğünde saklanır.
- Oluşturulan her bir özel anahtardan karşılık gelen açık anahtarlar elde edilir ve ayrı bir Python sözlüğünde saklanır.

Adım 3:
- Gönderici AMR tarafından gönderilen şifrelenmiş, nonce, tag ve ciphertext olmak üzere 3 parçadan oluşan mesaj alınır.

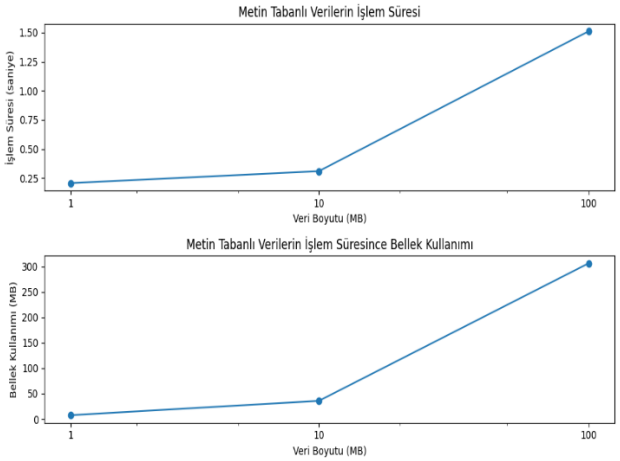
Adım 4:
- İlk olarak, gönderici ve alıcı/alıcılar için bir döngü başlatılır.
- Gönderici ve alıcı/alıcıların anahtarları kullanılarak ortak bir anahtar hesaplanır.
- Bu ortak anahtar, bir SHA-256 hash işleminden geçirilerek AES şifrelemesi için kullanılan anahtar haline getirilir.
- Veri, bu anahtar kullanılarak AES şifrelemesi ile çözülür.

Adım 5:
- Çözülen veri, ekrana yazdırılarak işleme hazır hale getirilir.

5. Bulgular

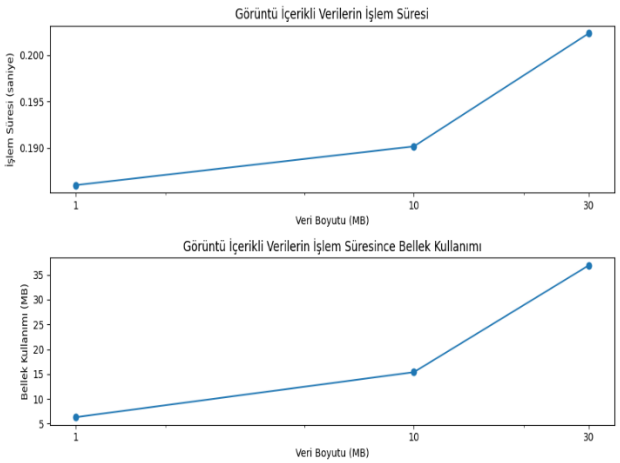
Otonom mobil robotlar genellikle; sensör verileri, kontrol komutları, görüntü/video verileri ve harita verileri kullanarak haberleşmektedir. Sensör verileri, kontrol komutları ve harita verileri metin tabanlı olarak ele alınmıştır. Görüntü/video verileri ise ayrı olarak ele alınmışlardır. Dolayısıyla Python dili ile yazılan, şifreleme ve şifre çözme algoritmaları; metin tabanlı, görüntü içerikli ve video içerikli olarak 3 adet farklı veri türü kullanılarak test edilmiştir. Algoritmalar, işlem süresi ve işlem süresince kullanılan bellek üzerinden performans testine tâbi tutulmuştur. İşlem süresi testi için, Python dili üzerinde yerleşik bulunan time kütüphanesi kullanılmıştır. İşlem süresince kullanılan bellek testi için ise Python dili üzerinde memory_profiler kütüphanesinin memory_usage modülü kullanılmıştır. Metin tabanlı veriler kullanılarak yapılan testlerde 1,10,100 MB boyutlarındaki dosyalar, görüntü içerikli veriler kullanılarak yapılan testlerde 1,10,30 MB boyutlarındaki dosyalar, video içerikli veriler kullanılarak yapılan testlerde 100,500,1000 MB boyutlarındaki dosyalar kullanılmıştır. Şifreleme algoritması sadece 1 alıcı AMR olacak şekilde düzenlenmiştir. Şifre çözme ile şifreleme algoritmaları kullanılarak yapılan test sonuçlarının aritmetik ortalaması alınmıştır. Elde edilen ortalama sonuçlar ve veri türleri kullanılarak, Python dili üzerinde matplotlib kütüphanesi aracılığıyla çizgi grafiği oluşturulmuştur.

Metin tabanlı verilerin işlem süresi ve işlem süresince bellek kullanımı ile ilgili oluşturulan çizgi grafiği Şekil 5'de belirtilmiştir. Metin tabanlı veri türleri için 1 MB boyutundaki dosyanın işlem süresi yaklaşık 0.25 saniye olarak ölçülmüştür. Bu değer ile 100 MB boyutundaki dosyanın işlem süresi arasında yaklaşık 1.25 saniye işlem süresi farkı olduğu ölçülmüştür. Ek olarak 1 MB boyutundaki dosyanın işlem süresince kullandığı bellek miktarı yaklaşık 7 MB olup, 100 MB boyutundaki dosyanın kullandığı bellek miktarı yaklaşık 300 MB'dir.



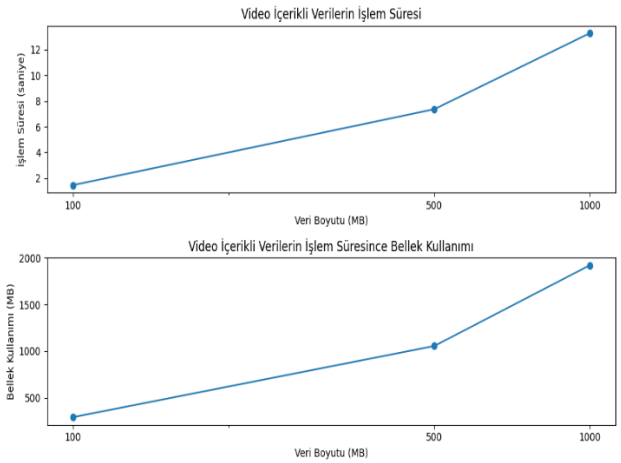
Şekil 5. Metin tabanlı verilerin çizgi grafiği

Görüntü içerikli verilerin işlem süresi ve işlem süresince bellek kullanımı ile ilgili oluşturulan çizgi grafiği Şekil 6'de belirtilmiştir. Görüntü içerikli veri türleri için 1 MB boyutundaki dosyanın işlem süresi yaklaşık 0.2 saniye olarak ölçülmüştür. Bu değer ile 30 MB boyutundaki dosyanın işlem süresi arasında yaklaşık 0.4 saniye işlem süresi farkı olduğu ölçülmüştür. Ek olarak 1 MB boyutundaki dosyanın işlem süresince kullandığı bellek miktarı yaklaşık 3 MB olup, 30 MB boyutundaki dosyanın kullandığı bellek miktarı yaklaşık 100 MB'dir.



Şekil 6. Görüntü İçerikli Verilerin Çizgi Grafiği

Video içerikli verilerin işlem süresi ve işlem süresince bellek kullanımı ile ilgili oluşturulan çizgi grafiği Şekil 7'de belirtilmiştir. Video içerikli veri türleri için 100 MB boyutundaki dosyanın işlem süresi yaklaşık 1 saniye olarak ölçülmüştür. Bu değer ile 1000 MB boyutundaki dosyanın işlem süresi arasında yaklaşık 10 saniye işlem süresi farkı olduğu ölçülmüştür. Ek olarak 100 MB boyutundaki dosyanın işlem süresince kullandığı bellek miktarı yaklaşık 300 MB olup, 1000 MB boyutundaki dosyanın kullandığı bellek miktarı yaklaşık 1900 MB'dir.



Şekil 7. Video İçerikli Verilerin Çizgi Grafiği

Ölçülen sonuçlarla, benzer hibrit şifreleme yöntemleri [31-33] ile ilgili çalışmalar ile kıyaslandığında kabul edilebilir bir performans sergilediği ve AMR'ler arasındaki kablosuz iletişim sırasında veri güvenliğini sağlayabildiği gözlenmiştir.

6. Sonuç

Bu çalışmada, otonom mobil robotların kablosuz haberleşme esnasında veri güvenliğini sağlamak amacıyla hibrit bir şifreleme yöntemi olan ECC ve AES yöntemlerinin bir arada kullanılarak veri güvenliğinin sağlanması önerilmiştir. Bu araştırmada, ECC ve AES'in birlikte kullanıldığı ECDSA-AES/GCM yöntemi ile AMR'ler arasında kablosuz iletişim sırasında oluşabilecek veri sızıntılarının önüne geçilmesi hedeflenmiştir. Çalışmanın sonucunda, bu hibrit şifreleme yöntemi ile AMR'ler arasında gerçekleştirilen veri iletiminin güvenli olduğu görülmüştür. Asimetrik şifreleme yöntemi ECC'nin güvenlik performansı ve simetrik şifreleme yöntemi AES'in hızlı iletişim ayrıcalığı, AMR'lerin kablosuz haberleşmeleri sırasında kullanılan verilerin güvenliğini sağlamakla kalmamış, ayrıca şifrelenmiş verilerin ulaşımını da hızlı bir şekilde sağlamayı başarmıştır. Bu çalışmanın ilgili şifreleme yapacak olanlara bir alternatif sağlama yolunda örnek olması düşünülebilir.

Kaynaklar

[1] Işık A.H., Çetin Ö. Multifunctional and Low Cost Autonomous Mobile Robot. Gazi Journal of Engineering Sciences, 6(2), 105-110, 2020.

[2] Siegwart R, Nourbakhsh I.R., Introduction to Autonomous mobile robots. The MIT Press, USA, 2004.

[3] Jaulin L. Mobile robotics. ISTE Press Elsevier, USA, 2007.

[4] To AWK, Paul G, Liu D. A comprehensive approach to real-time fault diagnosis during automatic grit-blasting operation by autonomous industrial robots. Robotics and Computer-Integrated Manufacturing, 49(1), 13-23, 2018.

[5] Intel. Otonom Mobil Robot (AMR) Genel Bakış. http://www.intel.com.tr (Erişim Tarihi: 22.05.2023).

[6] Milvus Robotics. Otonom Mobil Robotlar- AGV- AMR. http://www.milvusrobotics.com/tr/blog (Erişim Tarihi: 22.05.2023).

[7] Optimak Optimum STU. Otonom Mobil Robot Nedir? http://www.optimak.com.tr/otonom-mobil-robot-nedir/ (Erişim Tarihi: 22.05.2023).

[8] Advanced Motion Controls. Otonom Mobil Robotlar için Motor Kontrolörleri. http://www.a-m-c.com (Erişim Tarihi: 22.05.2023).

[9] TechCrunch. Erişim Tarihi: 2.06.2023. http://techcrunch.com/2022/04/12/aethon-robots-hospitals-hijacks/

[10] The Verge. Erişim Tarihi: 2.06.2023. https://www.theverge.com/2022/6/21/23177756/amazon-warehouse-robots-proteus-autonomous-cart-delivery

[11] Menezes AJ, van Oorschot PC, Vanstone SA. Handbook of applied cryptography. CRC press, 1996.

[12] Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120-126, 1978.

[13] Diffie W, Hellman ME. New directions in cryptography. IEEE transactions on Information Theory, 22(6), 644-654, 1976.

[14] Wikipedia. Elliptic-curve cryptography. http://en.wikipedia.org/wiki/Elliptic-curve_cryptography (Erişim Tarihi: 24.05.2023).

[15] Barker E, Barker W, Burr W, Polk W, Smid M. Recommendation for Key Management – Part 1: General (Revision 4). National Institute of Standards and Technology, 2016.

[16] Johnson D, Menezes A, Vanstone S. The Elliptic Curve Digital Signature Algorithm (ECDSA). International Journal of Information Security, 1(1), 36-63, 2001.

[17] Ferguson N, Schneier B, Kohno T. Cryptography Engineering: Design Principles and Practical Applications. Wiley Publishing, 2010.

[18] Daemen J, Rijmen V. The Design of Rijndael: AES - The Advanced Encryption Standard. Springer, 2002.

[19] McGrew D, Viega J. The Galois/Counter Mode of Operation (GCM). National Institute of Standards and Technology, 2004. http://csrc.nist.gov/publications/detail/sp/800-38d/final

[20] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE transactions on information theory, 31(4), 469-472, 1985.

[21] Barkan E, Biham E, Keller N. Instant ciphertext-only cryptanalysis of GSM encrypted communication. Journal of cryptology, 21(3), 392-429, 2005.

[22] Bernstein DJ. Introduction to post-quantum cryptography. Springer, Berlin, Heidelberg, 2009.

[23] Hu H, Ahn GJ, Jorgensen J. Detecting and resolving privacy conflicts for collaborative data sharing in online social networks. Proceedings of the 27th annual computer security applications conference, 103-112, 2011.

[24] Li Q, Li Y, Li X. Wireless Communication Security: A Survey. Security and Communication Networks, 2018, 1091493.

[25] Rivest RL, Shamir A, Adleman L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM, 21(2), 120–126, 1978.

[26] Daemen J, Rijmen V. The Design of Rijndael: AES - The Advanced Encryption Standard. Springer, 2002.

[27] Kamaludin NS, Razak MZA, Abdullah AH. Review of Hybrid Cryptography Algorithms for Secure Multimedia Communication. Journal of Telecommunication, Electronic and Computer Engineering, 6(1), 15–19, 2014.

[28] Legrand S, Lackorzynski A. PyCryptodome Documentation. http://www.pycryptodome.org (Erişim Tarihi: 2023).

[29] Langley A, Hamburg M. Elliptic Curve Digital Signature Algorithm (ECDSA). http://www.ecdsa.org (Erişim Tarihi: 2023).

[30] National Institute of Standards and Technology. Recommended Elliptic Curves for Federal Government Use, 2023.

[31] Yıldırım, K., & Demiray, H. E. (2008). Simetrik ve asimetrik şifreleme yöntemlerine metotlar: çirpilmiş ve birleşik akm-vkm. Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 23(3).

[32] Zhang, Q. (2021, January). An overview and analysis of hybrid encryption: the combination of symmetric encryption and asymmetric encryption. In 2021 2nd international conference on computing and data science (CDS) (pp. 616-622). IEEE.

[33] Alkady, Y., Habib, M. I., & Rizk, R. Y. (2013, December). A new security protocol using hybrid cryptography algorithms. In 2013 9th International Computer Engineering Conference (ICENCO) (pp. 109-115). IEEE.