KRİPTOGRAFİ VE AĞ GÜVENLİĞİ TEKNİK RAPORU
AES – DES – RSA Kullanarak İstemci–Sunucu Şifreleme Sistemi Analizi

Öğrenci Bilgileri
- Ad Soyad: Hüseyin Bünyamin Gülme
- Öğrenci No: 439591
- GitHub Deposu: https://github.com/huseyingulme/Kriptoloji

1. Giriş ve Problemin Tanımı

Bilgi güvenliği, modern bilgisayar ağlarının en temel gereksinimlerinden biridir. Günümüzde bankacılık sistemleri, e-ticaret platformları, bulut tabanlı servisler ve IoT altyapıları büyük ölçüde istemci–sunucu mimarisi üzerinden çalışmaktadır. Bu mimaride istemci ile sunucu arasında sürekli olarak veri alışverişi gerçekleşmekte ve bu veriler çoğu zaman açık ağlar (internet gibi) üzerinden iletilmektedir.

Açık ağ ortamlarında gerçekleştirilen veri iletimi;
- Pasif saldırılar (dinleme, trafik analizi)
- Aktif saldırılar (veri değiştirme, paket enjeksiyonu)
- Replay saldırıları
- Man-in-the-Middle (MITM) saldırıları

gibi çok sayıda tehdit barındırmaktadır. Bu tehditlere karşı en temel savunma mekanizması kriptografidir.

Bu projede ele alınan temel problem şudur:

Bir istemci ile sunucu arasında, ağ üzerinde dinlenmeye açık bir ortamda, veriler nasıl güvenli biçimde iletilebilir?

Bu soruya cevap ararken, yalnızca "hangi algoritma daha güçlüdür?" yaklaşımı benimsenmemiş; aynı zamanda hangi algoritma hangi amaç için uygundur, performans–güvenlik dengesi nasıl sağlanır ve ağ üzerindeki pratik etkiler nelerdir gibi sorular da araştırılmıştır.

1.1. Çalışmanın Amacı ve Kapsamı

Bu çalışmanın temel amaçları aşağıdaki şekilde özetlenebilir:
- Simetrik ve asimetrik kriptografik algoritmaların çalışma prensiplerini uygulamalı olarak incelemek
- AES, DES ve RSA algoritmalarını gerçek bir istemci–sunucu haberleşmesi üzerinde karşılaştırmak
- Hibrit şifreleme yaklaşımının neden modern sistemlerde zorunlu hale geldiğini göstermek
- Şifrelenmiş ağ trafiğinin Wireshark gibi araçlarla nasıl göründüğünü analiz etmek
- Manuel kriptografik implementasyonlar aracılığıyla algoritmaların iç yapısını derinlemesine kavramak

Bu kapsamda proje, hem kriptografi hem de ağ güvenliği disiplinlerini birlikte ele alan bütüncül bir çalışma olarak tasarlanmıştır.

2. Sistem Mimarisi ve Tasarım Yaklaşımı

2.1. İstemci–Sunucu Modelinin Seçilme Gerekçesi

İstemci–sunucu mimarisi, gerçek dünyada kullanılan güvenli iletişim sistemlerinin büyük çoğunluğunun temelini oluşturmaktadır. HTTPS, FTPS, SSH gibi protokoller bu mimari üzerine kuruludur. Bu nedenle projede seçilen mimari, teorik bilgilerin pratik karşılığını görmek açısından özellikle tercih edilmiştir.

TCP protokolü, aşağıdaki nedenlerle kullanılmıştır:
- Paket sıralaması ve yeniden iletim mekanizmaları
- Güvenilir bağlantı yapısı

- Uygulama katmanında güvenliğin test edilmesine olanak tanıması

Bu sayede kriptografik güvenlik, alt katman güvenliğinden bağımsız olarak ele alınabilmiştir.

2.2. Katmanlı Mimari Yaklaşım

Sistem, aşağıdaki mantıksal katmanlara ayrılmıştır:

1. Kullanıcı Arayüzü Katmanı (GUI)
2. Paketleme ve İletişim Katmanı
3. Kriptografik İşlem Katmanı
4. Kontrol ve Yönlendirme Katmanı (ProcessingManager)

Bu katmanlı yapı sayesinde:

- Kodun bakımı kolaylaşmış
- Güvenlik açıklarının izole edilmesi mümkün hale gelmiş
- Her katmanın sorumluluğu net biçimde ayrılmıştır

Bu yaklaşım, yazılım güvenliğinde önerilen separation of concerns ilkesine uygundur.

3. Veri Paketleme Stratejisi ve Güvenlik Etkileri

3.1. Metadata Kullanımının Önemi

Sadece verinin şifrelenmesi, güvenli iletişim için yeterli değildir. Alıcı tarafın:

- Hangi algoritmanın kullanıldığını
- Hangi anahtar ile çözümleme yapılacağını
- Paketin geçerli olup olmadığını

bilmesi gerekir. Bu nedenle DataPacket yapısı, metadata destekli olarak tasarlanmıştır. Timestamp alanı özellikle dikkat çekicidir. Bu alan sayesinde:

- Aynı paketin tekrar gönderilmesi tespit edilebilmekte
- Replay saldırıları için temel bir savunma sağlanmaktadır

Gerçek dünyada TLS protokollerinde kullanılan nonce ve sequence number mekanizmaları ile benzer bir mantık izlenmiştir.

3.2. JSON Paketleme Modunun Akademik Katkısı

JSON tabanlı paketleme modu, doğrudan güvenliği artırmasa da analiz edilebilirliği ciddi ölçüde artırmıştır. Bu sayede:

- Şifreli veri ile kontrol bilgileri net biçimde ayrılmış
- Ağ analiz araçlarında paket içeriği daha anlaşılır hale gelmiştir
- Öğrenci açısından kriptografik süreçlerin gözlemlenmesi kolaylaşmıştır

Bu özellik, projeyi salt bir kodlama çalışması olmaktan çıkarıp, deneysel bir güvenlik analizine dönüştürmüştür.

4. Kriptografik Algoritmaların Derinlemesine İncelenmesi

Kriptografik algoritmalar yalnızca veriyi gizlemekle kalmaz, aynı zamanda matematiksel yapı, anahtar yönetimi, blok yapısı ve operasyonel adımlar içeren karmaşık süreçlere dayanır. Bu bölümde AES, DES ve RSA'nın bu yönleri ayrıntılı olarak ele alınmıştır.

4.1. AES (Advanced Encryption Standard)

AES, simetrik anahtarlı blok şifreleme algoritmalarının günümüzdeki fiili standardıdır. 2001 yılında NIST tarafından DES'in yerini almak üzere kabul edilmiştir ve yüksek güvenlik ile performans sunması nedeniyle modern iletişim protokollerinde yaygın olarak kullanılır. (Fiveable)

Teknik Özellikleri

- Blok boyutu: 128 bit
- Anahtar uzunlukları: 128, 192 veya 256 bit
- Round sayısı: Anahtar uzunluğuna göre 10/12/14
- Yapı: Substitution–Permutation Network (SPN)
- AES, DES'in aksine Feistel yapısı kullanmaz; tüm blok üzerinde paralel dönüşümler uygular.

İşlem Adımları

AES'in şifreleme süreci aşağıdaki adımlardan oluşur:

1. AddRoundKey: Blok ile anahtarın bit düzeyinde XOR'u
2. SubBytes: Her byte, önceden tanımlanmış bir S-Box tablosu kullanılarak doğrusal olmayan şekilde dönüştürülür
3. ShiftRows: Durum matrisindeki satırlar farklı miktarda sola kaydırılır
4. MixColumns: Her sütun, bir sabit matris ile çarpılarak difüzyon sağlanır
5. Bu dört adım çoğu turda uygulanır, son turda MixColumns atlanır.

Anahtar Genişletme (Key Schedule)

AES, başlangıç anahtarını her tur için kullanılacak ayrı alt anahtarlara dönüştürür. Bu genişletme süreci, özgün anahtarın farklı kombinasyonlarını üretir ve saldırılara karşı direnç sağlar.

Güvenlik ve Performans

AES'in gücü:

- Yüksek anahtar karmaşıklığı
- Güçlü S-Box tablosu sayesinde doğrusal olmayanlık
- Donanım hızlandırma (AES-NI gibi) desteği
- Paralel ve yazılım dostu yapısı sayesinde hızlı işlem

  Bu özellikler AES'i hem güvenli hem de pratik kılar.

4.2. DES (Data Encryption Standard)

DES, kriptografi tarihinde simetrik blok şifrelemenin ilk yaygın standardıdır. Ancak 56 bit'lik anahtar uzunluğu nedeniyle günümüz standartlarına göre güvenli değildir.

Teknik Özellikleri

- Blok boyutu: 64 bit
- Anahtar uzunluğu: 56 bit (8 bit parite dahil)
- Round sayısı: 16
- Yapı: Feistel Network

Feistel Ağı

Feistel şeması, blok veriyi iki eşit parçaya böler ve bir taraf fonksiyondan geçirilerek diğer tarafla XOR edilir. İşlem ardından yarılar yer değiştirir ve bu 16 tur boyunca devam eder.

Tur Fonksiyonu

Her turda:

1. Sağ yarı 48 bit'e genişletilir
2. Alt anahtar ile XOR'lanır
3. S-Box'larda doğrusal olmayan dönüşümden geçirilir
4. P-Box ile permütasyon yapılır

   Bu süreç DES'e belirli ölçüde güvenlik sağlar ancak kısa anahtar uzunluğu brute–force saldırılarına karşı yetersizdir.

Güvenlik Açıkları

DES'in zayıflığı, özellikle anahtar uzunluğunun artık modern hesaplama gücüyle kolayca kırılabilir olmasıdır. Bu yüzden DES, güncel güvenlik standartlarında artık güvenli kabul edilmemektedir.

4.3. RSA (Asimetrik Kriptografi)

RSA, açık anahtar kriptografisinin en bilinen algoritmalarındandır. RSA'nın güvenliği, büyük asal sayıların çarpımı ve modüler aritmetik üzerine kuruludur.

Anahtar Yapısı

RSA'da iki anahtar vardır:

- Public Key (e, n): Herkesle paylaşılabilir
- Private Key (d, n): Sadece alıcı tarafında saklanır

Bu sayede:

- Public key ile şifreleme yapılır
- Private key ile çözme gerçekleşir

Matematiksel Güvenlik

RSA'nın güvenliği, büyük sayıları asal çarpanlarına ayırmanın zorluğuna dayanır. Bu problem, mevcut klasik bilgisayarlarla pratikte çözülemez olarak kabul edilir.

Performans Dezavantajı
RSA'nın hesaplama maliyeti yüksektir; bu nedenle modern protokollerde yalnızca anahtar değişimi veya dijital imza gibi işlemler için kullanılır. Veri şifreleme için doğrudan AES veya benzeri simetrik algoritmalar tercih edilir.

4.4. Simetrik ve Asimetrik Arasındaki Temel Farklar

Özellik	AES	RSA
Tür	Simetrik	Asimetrik
Anahtar	Tek anahtar	Public + Private
Performans	Yüksek	Düşük
Kullanım	Veri şifreleme	Anahtar değişimi / imza
Anahtar Güvenliği	Paylaşım gerektirir	Public anahtar paylaşılabilir

Performans–Güvenlik Dengesi
AES gibi simetrik algoritmalar büyük veri bloklarını çok hızlı şifrelerken, RSA gibi asimetrik algoritmalar daha ağır ama anahtar paylaşımı açısından daha güvenli yöntemler sunar. Bu nedenle hibrit modeller modern sistemlerde standart hale gelmiştir.

5.Hibrit Şifreleme Modelinin Stratejik Önemi

Hibrit şifreleme yaklaşımı, bu projenin en kritik kazanımlarından biridir. Bu yapı sayesinde:

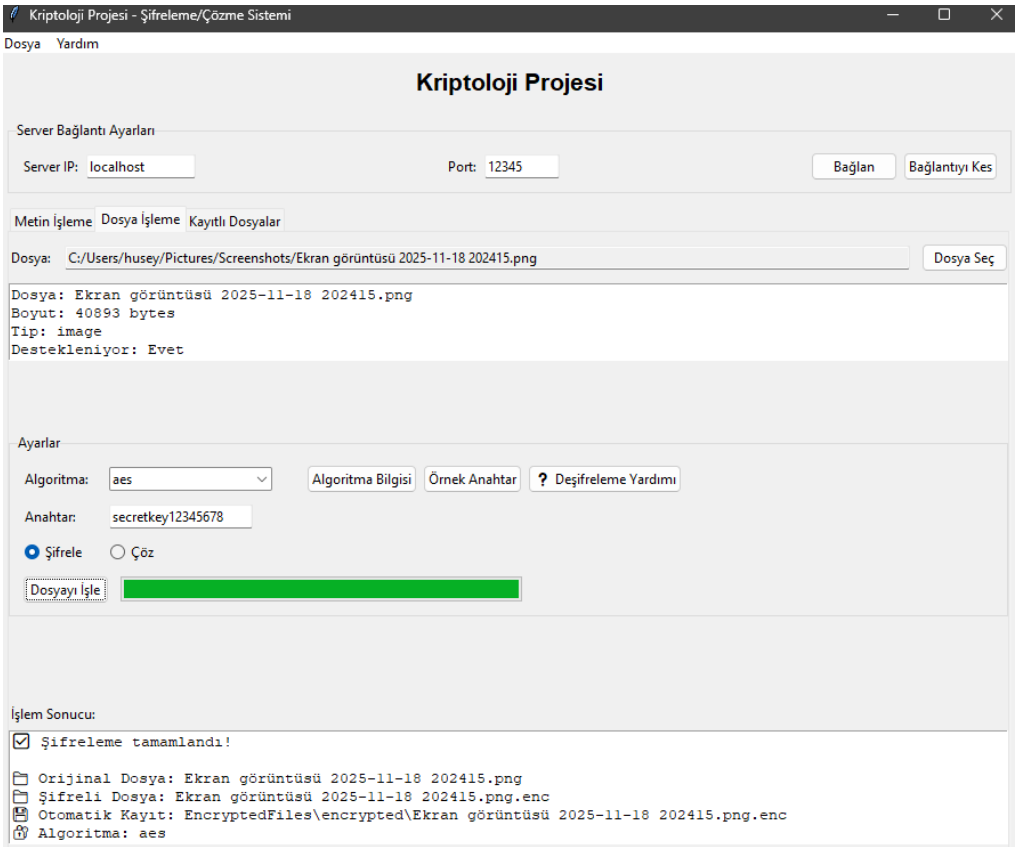
- Asimetrik kriptografinin güvenli anahtar paylaşımı avantajı
- Simetrik kriptografinin yüksek performansı

bir araya getirilmiştir.

Bu yaklaşım, modern TLS protokollerinde kullanılan:

- RSA + AES
- ECDHE + AES

kombinasyonlarının mantığını birebir yansıtmaktadır.

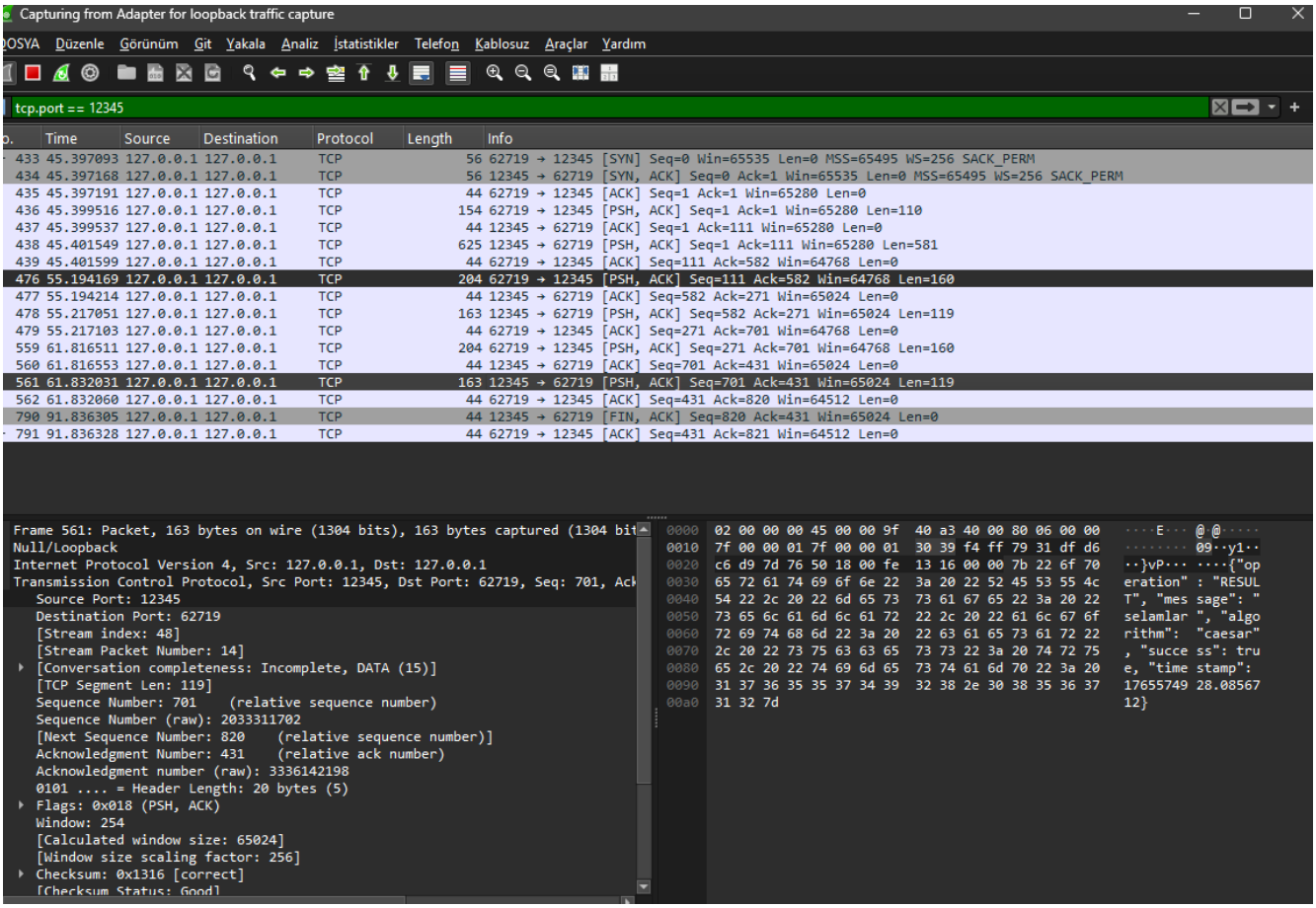


6.Wireshark Analizi ve Ağ Güvenliği Açısından Değerlendirme

Wireshark üzerinden yapılan incelemeler, kriptografinin ağ seviyesindeki etkilerini somutlaştırmıştır:

- Düz metin verinin tamamen gizlendiği
- Saldırganın yalnızca anlamsız byte dizileri görebildiği
- Paket boyutlarının kullanılan algoritmaya göre ciddi biçimde değiştiği

Bu analiz, “şifreleme yapılınca her şey güvenlidir” algısının yanlış olduğunu da göstermiştir. Yanlış algoritma seçimi, ağ performansını doğrudan olumsuz etkileyebilmektedir.



7.Genel Değerlendirme ve Öğrenilen Dersler

Bu proje sürecinde elde edilen en önemli kazanımlar şunlardır:

- Kriptografi yalnızca matematik değil, aynı zamanda sistem tasarımıdır
- Güvenlik, tek bir algoritma ile değil, bütüncül bir mimari ile sağlanır
- Manuel implementasyonlar, algoritmaların gerçek doğasını anlamak için vazgeçilmezdir
- Ağ trafiği analiz edilmeden yapılan güvenlik iddiaları eksik kalır

8.Sonuç

Bu çalışma, güvenli istemci–sunucu haberleşmesinin yalnızca güçlü algoritmalarla değil; doğru mimari, hibrit şifreleme, paketleme stratejileri ve ağ seviyesi analizlerle mümkün olduğunu göstermiştir. Proje, teorik kriptografi bilgisini pratik ağ güvenliği uygulamalarıyla birleştiren kapsamlı ve öğretici bir çalışma olarak tamamlanmıştır.