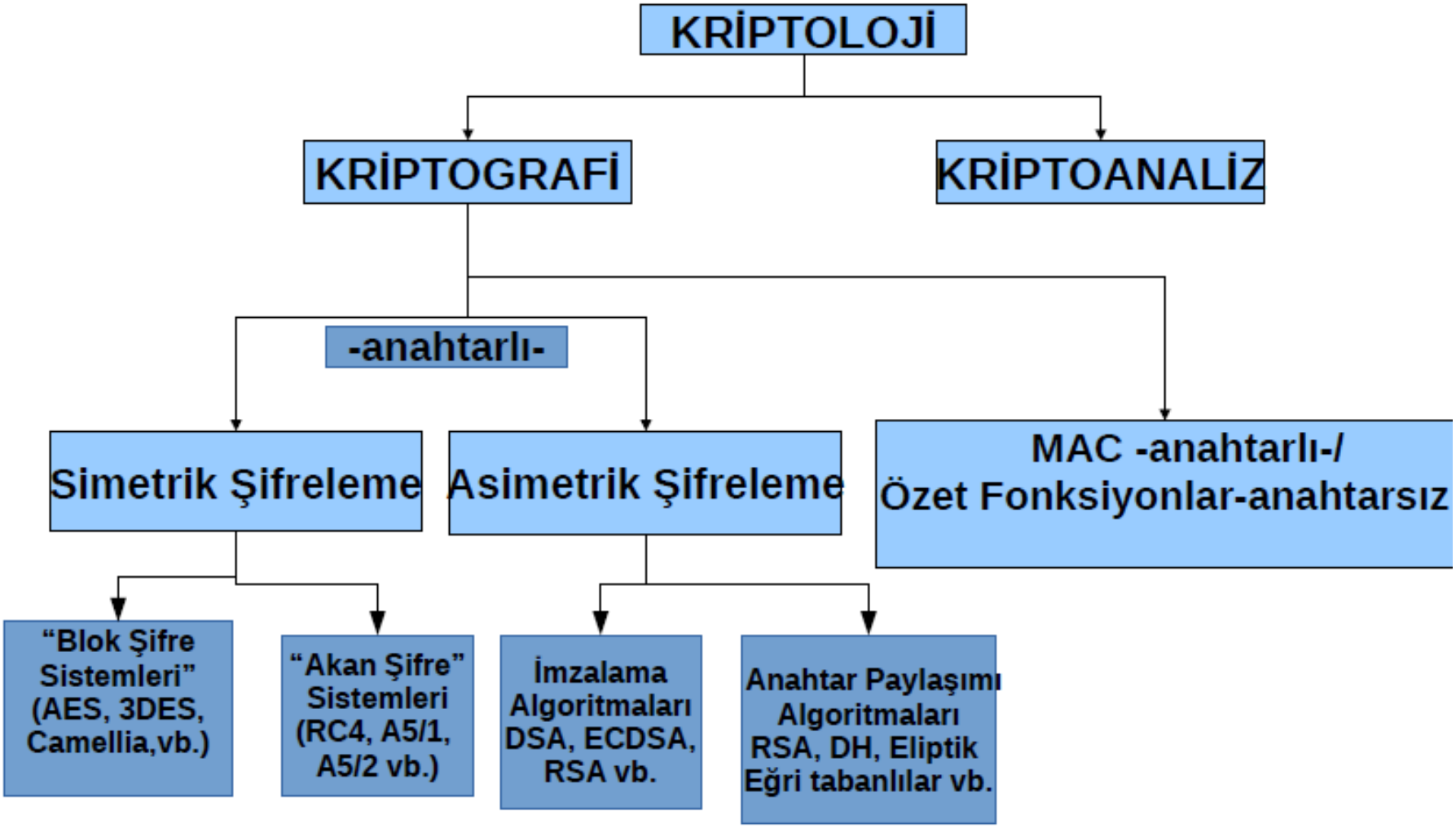
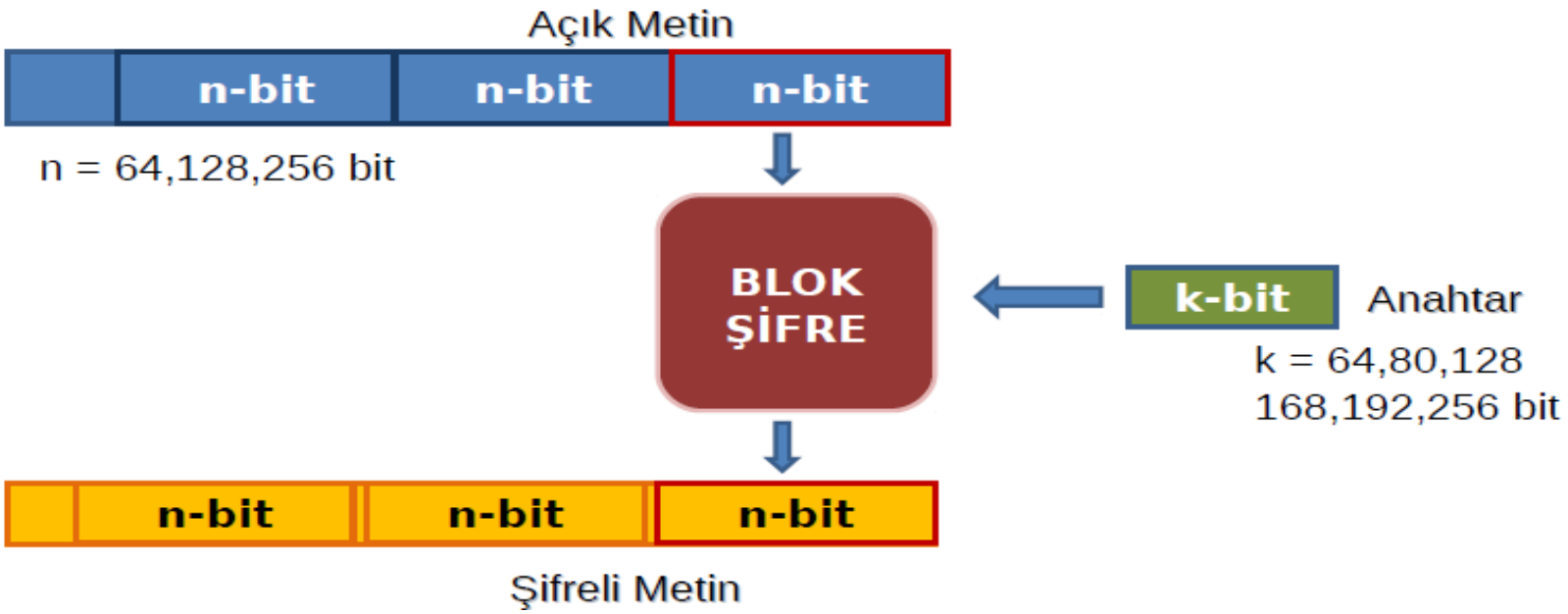
Bilgi Güvenliği
ve
Kriptografi


KRİPTOLOJİ
KRİPTOGRAFİ
KRİPTOANALİZ
-anahtarlı-
Simetrik Şifreleme
Asimetrik Şifreleme
MAC -anahtarlı-/
Özet Fonksiyonlar-anahtarsız
"Blok Şifre Sistemleri" (AES, 3DES, Camellia,vb.)
"Akan Şifre" Sistemleri (RC4, A5/1, A5/2 vb.)
İmzalama Algoritmaları DSA, ECDSA, RSA vb.
Anahtar Paylaşımı Algoritmaları RSA, DH, Eliptik Eğri tabanlılar vb.

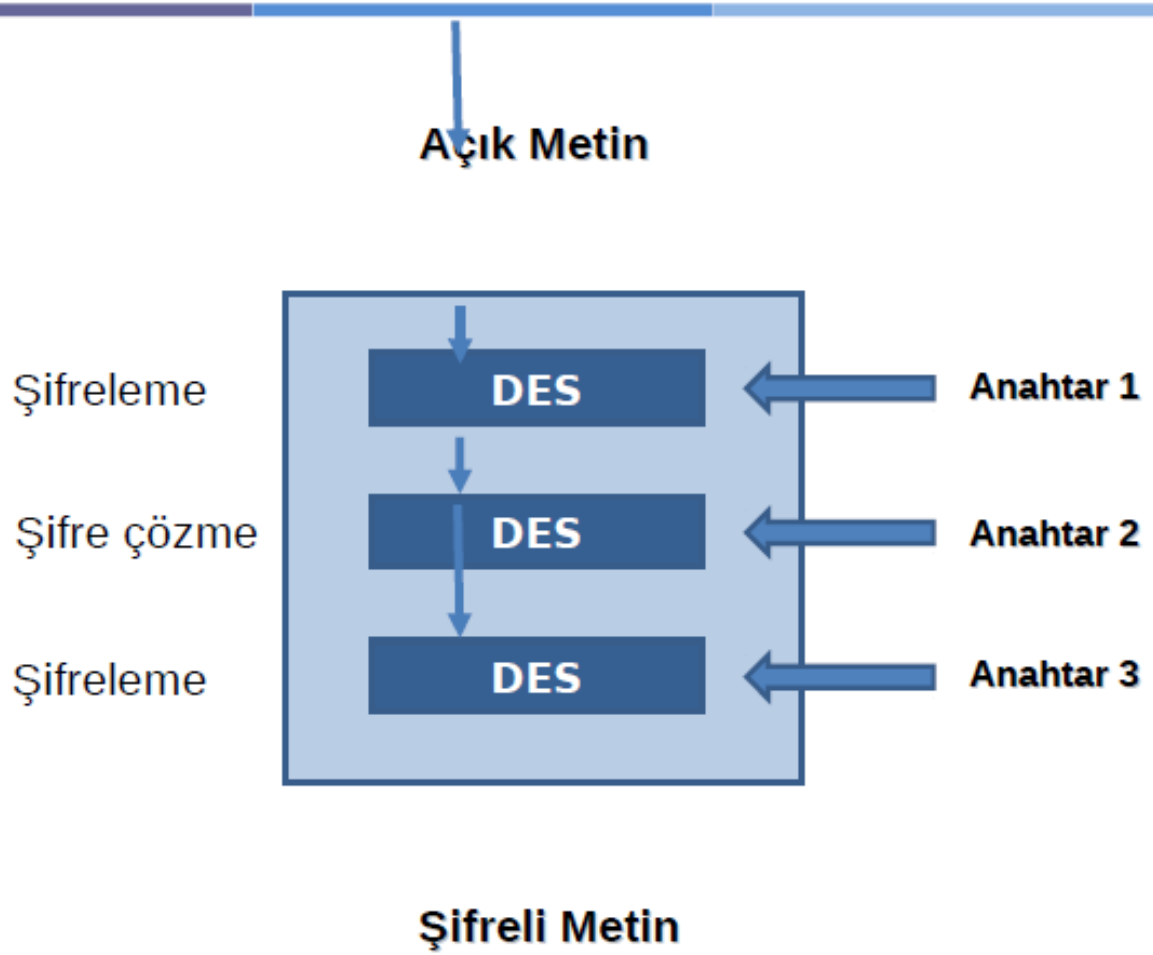
Blok Şifreler

➢ Blok şifreler, açık metni eşit uzunluktaki bloklara ayırıp, her bir bloğu bir fonksiyon yardımı ile tek tek şifreleyerek şifreli metni oluşturur.



- DES, AES, IDEA, RC5, Blowfish

Bilgi Şifreleme Standardı(3DES)



3DES anahtar uzunluğu 56*2=112-bit veya 56*3=168-bit

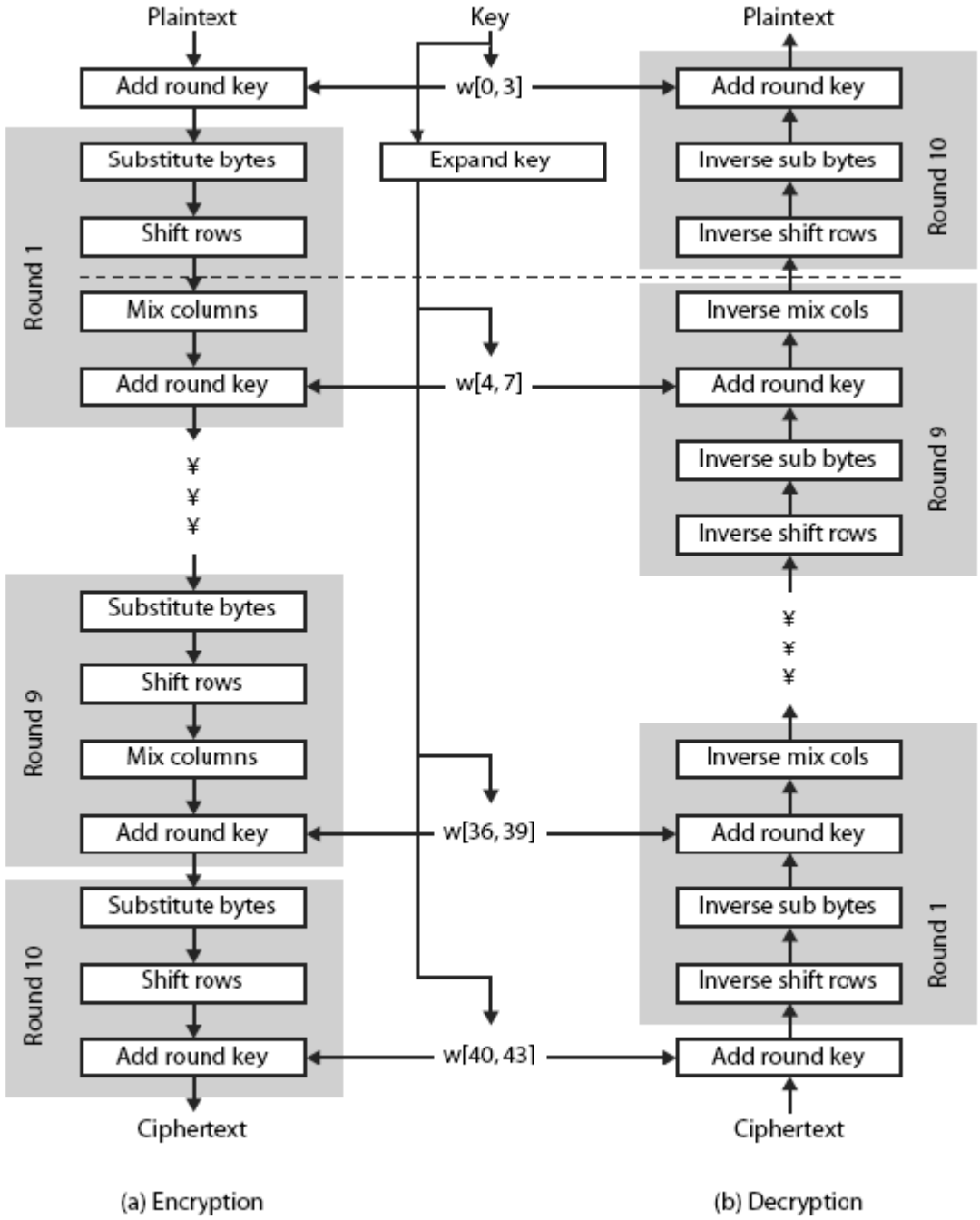
Algoritma	Anahtar Uzunluğu	Tur Sayısı	Matematiksel İşlemler
DES	56 Bit	16	XOR, Sabit S-boxes
Triple DES	112 veya 168 bit	48	XOR, Sabit S-boxes
IDEA	128 Bit	8	XOR, Toplama, Çarpma
Blowfish	Değişken, 448 bit	16	XOR, Değişken S-Boxes, Toplama
RC5	Değişken 2048 Bit	Değişken 255	Toplama, Çıkartma, XOR, Döndürme
CAST-128	40-128 bit	16	Toplama, Çıkartma, XOR, Döndürme, Sabit S-boxes

Tablo 6.16. Değişik Simetrik Kriptolama algoritmalarının özellikleri

6.12.2 AES (Advanced Encryption Standard)

3DES algoritması her nekadar 168 bitlik anahtar kullanıyor ve brute-force saldırılarına karşı yeterli güvenlik sağlıyor ise de üç adet DES'in ard arda çalışması nedeniyle yavaş bir algoritmadır. Bu nedenle NIST 1997'de 3DES'in yerini alacak daha hızlı ve güvenli bir simetrik şifreleme algoritması geliştirilmesini önerdi. Bu çağrı sonunda Belçikadan Dr. Joan Daemen ve Dr.Vincent Rijmen geliştirdiği Rijndael algoritması AES olarak kabul edildi. AES'in önemli özellikleri aşağıda verilmişitir.

1. 128 bit veri, 128/192/256 bitlik anahtar uzunluğuna sahiptir.
2. Feistel networkü yerine iteratif olarak çalışır
3. Veriyi dört bayt lık dört sütunluk bloklar halinde işler.
4. Herbir tur'da veri bloğunun tamamı üzerinde işlem yapar.
5. Basit, bilinen saldırılara karşı dirençli, birçok işlemcide hızlı ve kod basitliği sağlayacak şekilde tasarlanmıştır.



Şekil 6.19 : AES Şifreleme ve Deşifreleme adımları

1 DES(Feistel) mimarisinde veri bloğunun yarısı diğer yarısını modifiye etmekte kullanılır, sonra yer değiştirilir. AES(Rijndael) mimarisinde her iki yarı da paralel şekilde işlenir.

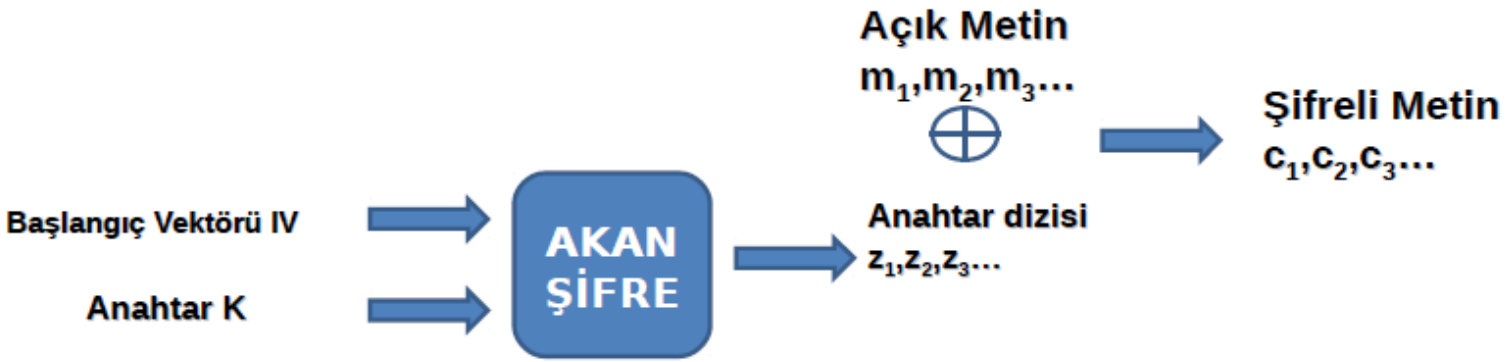
2 Sağlanan giriş anahtarı 40 adet dörtlük 32 bitli wordler şeklinde genişletilir w[i]. Dört farklı 128 bitlik kelime herbir turda tur anahtarı olarak kullanılır.
Herbir turdaki dört farklı everede, bir permutasyon ve üç yer değiştirme kullanılır.
- Substitute baytları: bloğun bayt bayt yer değiştirmesi için S-box'lar kullanılır(her bayt için bir S-box).
- Shift-Rows: Basit bir permutasyon(bayt'ları grup ve sütunlar arasında değiştirme)
- Mix columns: GF(2^8) üzerinde yapılan aritmetiği kullanarak yer değiştirme
- Add-Round key: Basit bit bit XOR işlemi(mevcut blok ve genişletilen anahtarın turdaki hali ile)

3 Yapı çok basit: Şifreleme ve deşifreleme için şifreleyici add round key evresi ile başlar. Herbiri 4 evre olan 9 tur ile devam eder.

4 Sadece add round key evresi anahtar kullanır. Bu nedenle şifreleyici add round key evresi ile başlar ve biter.

Akan Şifreler

- Akan Şifreler işlevini açık metnin her bir karakterini zamanla değişen bir fonksiyona sokarak yerine getirir .
- Girdi olarak alınan bir anahtar (K) ve başlangıç vektörü (IV) ile üreteç mümkün olduğu kadar uzun periyotlu ve rastgele gözüken anahtar dizilerini (z_1,z_2,z_3...) üretir ve elde ettiği anahtarı açık metinle şifreleme fonksiyonuna (h) sokarak şifreli metni elde eder.



- Vernam cipher (one tme pad),RC4, A5/1 (GSM-ses şifreleme)

Mesaj Kimlik Doğrulama Kodu (MAC)

Mesaj Kimlik Doğrulama Kodu (Message Authentication Code (MAC)) algoritması, bir gizli anahtar kullanır ve verilen mesaj için küçük bir veri oluşturur. Bu veri, mesajın sonuna eklenir.

Varsayım: A ve B kullanıcıları ortak ve gizli anahtar K_{AB} üzerinde anlaşsınlar.

$MAC_M = F(K_{AB}, M)$

Mesaj Kimlik Doğrulama Kodu (MAC)

- Blok Şifre Sistemleri kullanılarak MAC algoritmaları oluşturulmaktadır.
- MAC algoritmasının tersi yoktur.
- Bir çeşit girdi mesajına ve ayrıca gizli anahtara özgü özeti çıktı olarak verir.
- Alternatifi, kriptografik özet fonksiyonlarıdır.

Tek yollu Özet Fonksiyonu

- Özet Fonksiyonu Hash function değişken uzunlukta bir mesajı girdi olarak alıp, çıktı olarak sabit uzunlukta $H(M)$ (parmak izi, özet) yi üretir.

- MAC algoritmaları gibi gizli bir anahtar girdilerden biri olmaz

- Mesajın parmak izi, mesaj ile birlikte kimlik (mesajın kaynağını) doğrulama için gönderilir

Kriptografik Özet Fonksiyonu

"İdeal bir kriptografik özet fonksiyonu şu dört özelliği sağlamalıdır:

+ Herhangi bir mesaj için özet hesaplamak kolay olmalıdır.
+ Bir özete karşılık gelecek mesajı oluşturmak (hesaplama karşı imkansız) zor olmalıdır.
+ Özeti değişmeyecek şekilde mesajı değiştirmek (hesaplama karşı imkansız) zor olmalıdır.
+ Aynı özete sahip iki farklı mesaj bulmak zor olmalıdır."

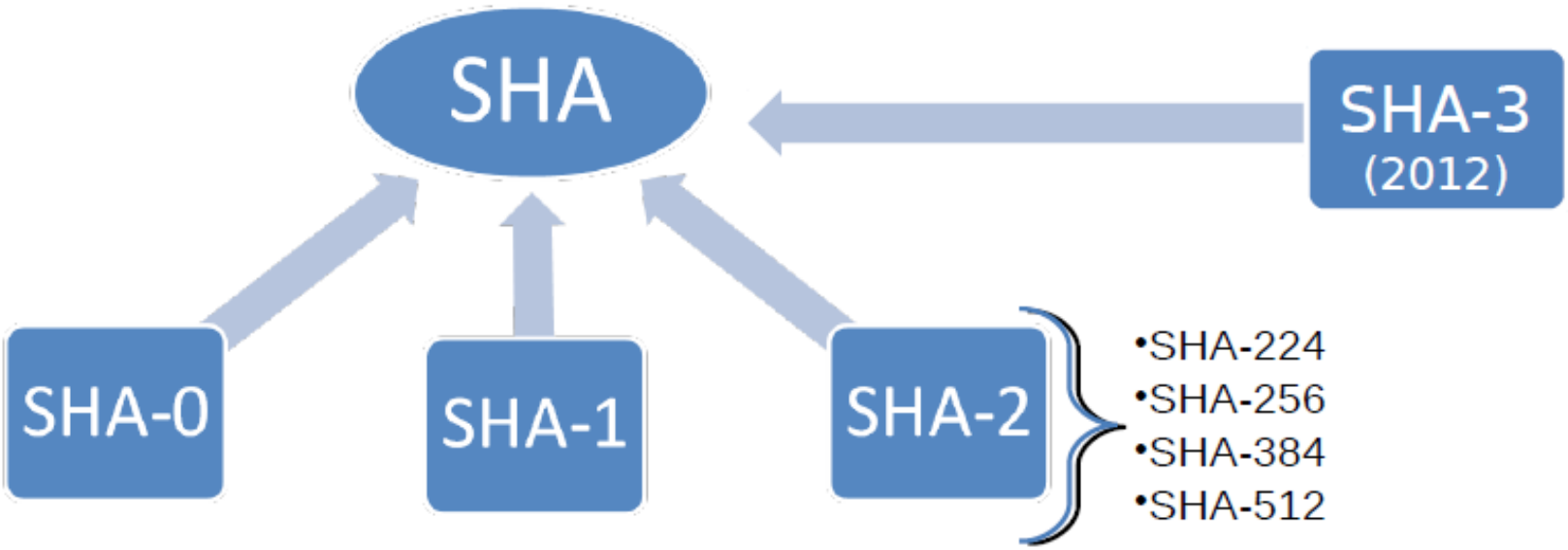
Kriptografik özet fonksiyonun sağlaması gereken özellikler fazlası için:

" Kriptografik özet fonksiyonları, bilgi güvenliği konuları olan sayısal imza, mesaj doğrulama kodu ve diğer doğrulama yöntemlerinde yaygın olarak kullanılmaktadır. "

- MD5, SHA

SHA (Secure Hash Algorithm)

• 1993'te Amerikan Ulusal Güvenlik Kurumu (NSA) tasarladı ve Ulusal Teknoloji ve Standartlar Enstitüsü (NIST) yayımlandı (SHA-0)



• 1995'te SHA-0 SHA-1 olarak yeniden tasarlandı

HMAC

- HMAC, bir çeşit MAC fonksiyonudur ve kriptografik bir özet fonksiyonu kullanılarak tasarlanır.
- Yazılımda hızlı çalışır
- Gizli anahtar kullanımı gereklidir.
- Tasarım ilkeleri RFC 2104 de listelenmiştir.
- Ipsec te kullanılır
- Aynı anda hem bütünlük hem de kimlik doğrulama için kullanılır

Asimetrik Kriptografi (Açık Anahtar Kriptografisi)

- 1976 yılında Diffie ve Hellman
- Gizliliğin yanı sıra, kimlik doğrulama ve inkar edememe
- Simetrik Kriptografi'deki anahtar dağıtımı sorununa çözüm
- Gizli ve açık, iki çeşit anahtar mevcut
- Açık anahtarlar herkes tarafından bilinir.
- Gizli anahtarlar kişiye özeldir.

Açık Anahtar Şifreleme Uygulamaları

Şifreleme / Deşifreleme (Encryption/Decryption) – mesaj alıcının açık anahtarı ile şifrelenir

Dijital İmza (Digital signature) – gönderici, mesajını (genelde mesaj özetini) kapalı anahtarı ile şifreler

Anahtar Değişimi (Key Exchange) – Simetrik şifreleme algoritmasının gizli anahtarını, ik taraf paylaşır. Basitçe, taraflardan biri bu anahtarı belirler ve alıcının açık anahtarı ile şifreler ve alıcıya iletir.

Anahtar Dağıtımı

Şimetrik şifreleme yöntemlerinde ortak bir anahtar her iki grup tarafından paylaşılır. Problem, bu anahtarın güvenli olarak dağıtılmasıdır. Güvenli bir sistem sık sık anahtar dağıtım yönteminin kırılmasıyla etkisiz hale gelebilir

Verilen A ve B grupları için değişik anahtar dağıtımı alternatifleri olabilir

A anahtarı seçer ve fiziksel olarak B'ye iletir.

Üçüncü şahıs anahtarı seçer , A ve B'ye dağıtır

Eğer A ve B önceden haberleşiyorsa,önceki anahtarı kullanarak yeni anahtarı şifreler

Eğer A ve B , C ile birlikte güvenli bir iletişim kanalına sahipse, C anahtarı A ve B arasında iletir

Tipik olarak anahtarların bir hiyerarşisi vardır.

Oturum anahtarı, Herbir oturum için kullanılır. Ağdaki N adet hostun kurabileceği oturum sayısı N(N-1)/2 adettir. Yani N(N-1)/2 adet oturum anahtarı kullanılabilecektir.

Oturum anahtarı:

Geçici anahtardır

Verinin kullanıcılar arasında şifrelenmesi için kullanılır.

Tek bir oturumda kullanılır ve sonra atılır.

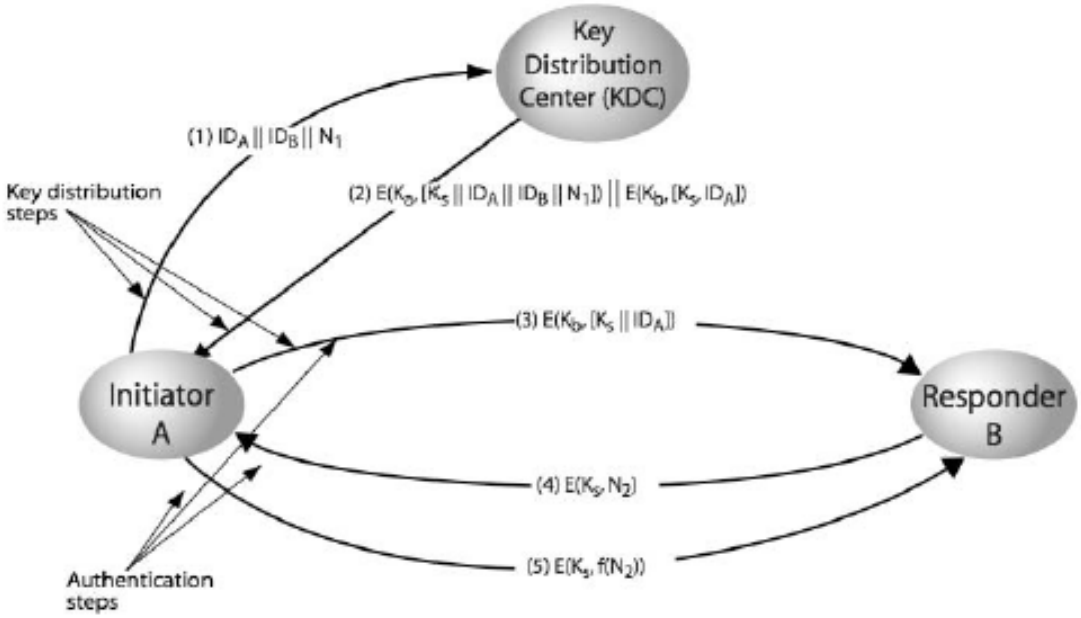
Ana Anahtar

Anahtar dağıtım merkezi ile kullanıcılar arasında N adet tir.

Ana anahtar:

Oturum anahtarlarını şifrelemek için kullanılır

Kullanıcı ile anahtar dağıtım merkezi arasında paylaşılır



Şekil 6.23: Anahtar dağıtımı senaryosu

Merkezi olmayan anahtar dağıtımı

Merkezi olmayan anahtar dağıtımında, herbir uç sistem, oturum anahtarı dağıtımı için güvenli bir şekilde haberleşmesi gerekir. Böylece N adet uç sistemin konfigürasyonu için N(N-1)/2 adet ana anahtar gerekebilir.

Oturum anahtarı aşağıdaki adımlar ile sağlanır

- A,B 'den N_1 içeren bir mesaj ile oturum anahtarı ister
- B, ortak olan ana anahtar ile şifrelenmiş şekilde A'ya cevap verir. Mesajda B'nin seçtiği oturum anahtarı ve $f(N_1)$, (N_1+1) , N_2 bulunur
- Yeni oturum anahtarı ile A, $f(N_2)$ 'yi B'ye gönderir.

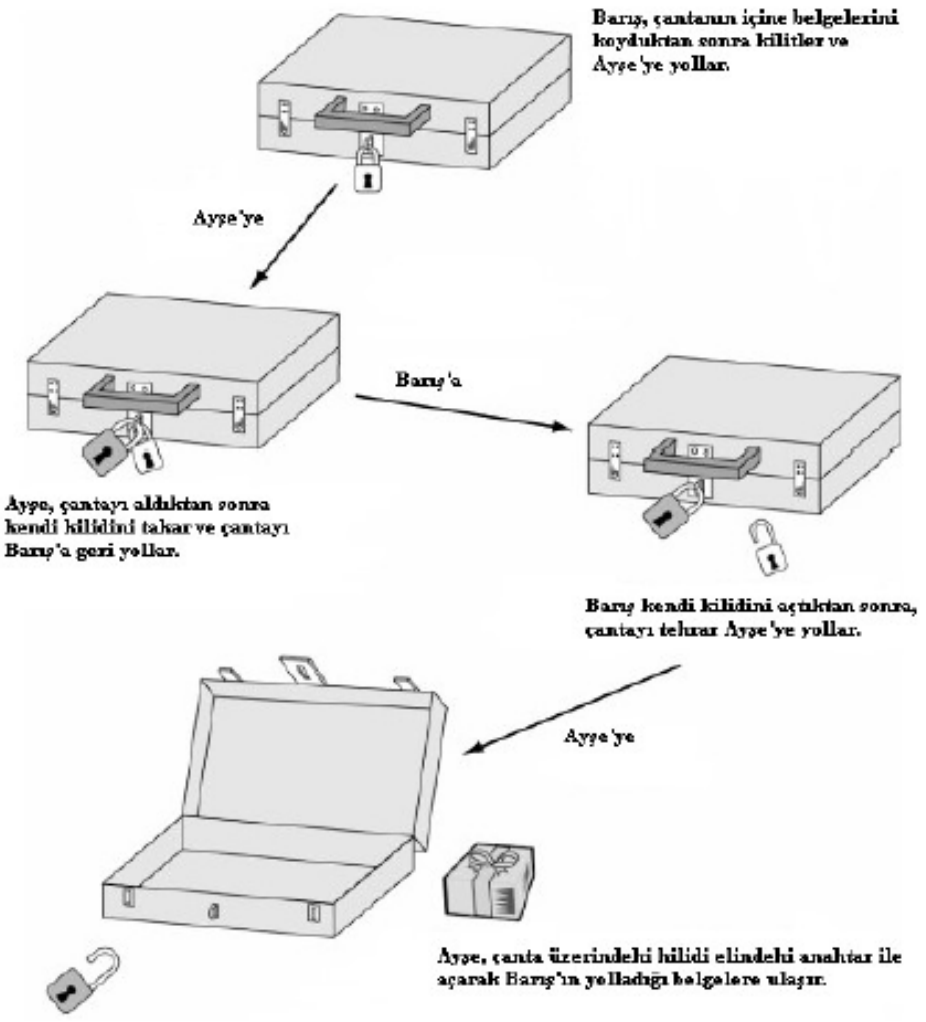
Böylece herbir düğüm ençok (N-1) ana anahtar saklamak zorunda kalır veya gerektiğinde üretilip kullanılır

Diffie-Hellman Anahtar Değişimi

•Gizli anahtarlı sistemlerde şifreleme ve şifre çözme için ortak bir anahtar gerekmektedir.

•Diffie-Hellman anahtar değişimi bu anahtarı oluşturmak için kullanılmaktadır

•Sonlu cisimler üzerinde veya eliptik eğri aritmetiğinde bu anahtar değişiminin uygulaması yapılabilmektedir.



Ayrık Logaritma Problemi

- Z_p *= {1, 2, …,p-1}

Z_p*'de ***g*** ve ***y*** verilmiş ve $g^x = y \pmod p$ ise x kaçtır?

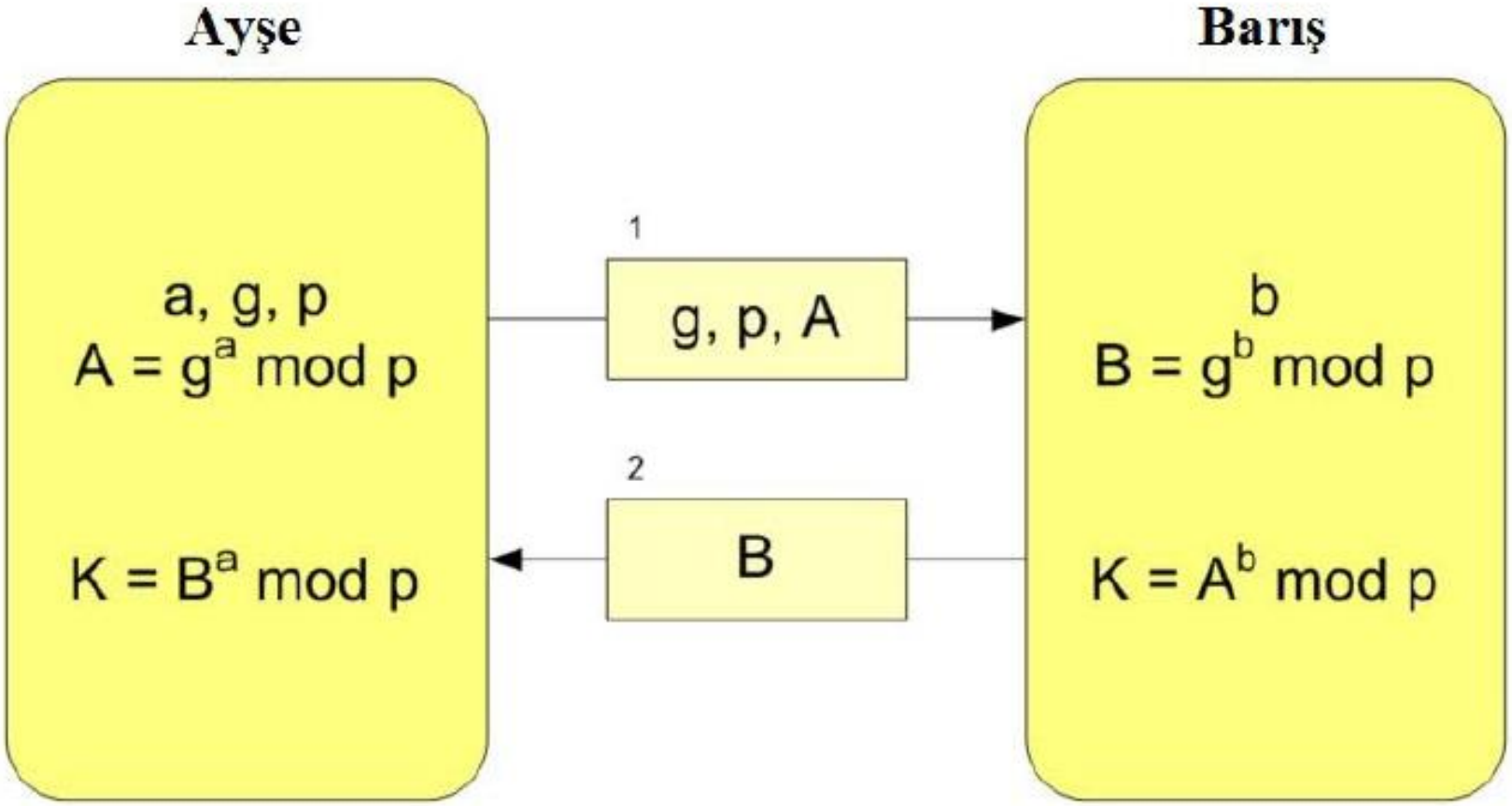
$x = \log_g y$ bulunması zor bir problemdir.

Örnek: Z_{17}'de g=3 ve y=11 ,

$3^x = 11 \pmod{17} \Rightarrow x = ?$

$3 \bmod 17 = 3$
$3^2 \bmod 17 = 9$
$3^3 \bmod 17 = 10$
$3^4 \bmod 17 = 13$
$3^5 \bmod 17 = 5$
$3^6 \bmod 17 = 15$
$3^7 \bmod 17 = 11$
$3^8 \bmod 17 = 16$
$3^9 \bmod 17 = 14$
$3^{10} \bmod 17 = 8$
$3^{11} \bmod 17 = 7$
$3^{12} \bmod 17 = 4$
$3^{13} \bmod 17 = 12$
$3^{14} \bmod 17 = 2$
$3^{15} \bmod 17 = 6$
$3^{16} \bmod 17 = 1$
$3^{17} \bmod 17 = 3$

Diffie-Hellman Anahtar Değişimi



$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$

RSA Güvenilirliği

- Deneme-yanılma saldırısını (Brute force attack): tüm olası anahtarları deneyerek bulmaya– dayanıklık için büyük e and d seçilebilir
- Bununla beraber ne kadar büyük anahtar kullanılır ise açık şifreleme algoritması o kadar yavaşlayacak
- Büyük n tam sayısı için büyük iki asal sayı çarpanını (Standartların tavsiyede ettiği şekilde, saldırılara kuvvetli asalların seçildiğini varsayarsak) bulmak adına çarpanlarına ayırma metodu zor bir problemdir.
- 1994 yılında 428 bit RSA n çarpanlarına ayrıldı. Ödül: $100
- 2010 yılı başlarında 768 bit uzunluğunda n tam sayısı (232 ondalık basamaklı), **"Number Field Sieve"** algoritması ile çarpanlarına ayrılmıştır (http://eprint.iacr.org/2010/006.pdf).
- RSA anahtarının uzunluğu genelde 1024–2048 bit. Bazı uzmanlar, 1024-bit anahtarların yakın zamanda kırılabileceğini düşünüyorlar.
- Standartlar, n tam sayısının en az 2048 bit olmasını tavsiye ediyor.

Eliptik Eğri Kriptografi

- Eliptik Eğri Kriptografi (Elliptic Curve Cryptography (ECC)) tabanlı şifreleme algoritmaları, RSA yerine tercih edilmeye başlandı.

- Dr. Scott Vanstone (Matematik ve Bilgisayar Bilimleri Profesörü ve Certicom şirketi kurucularından), ECC için açık anahtarlı kriptografinin gelecek nesli (özellikle kablosuz iletişim) olarak bahsediyor. Certicom firmasının ECC üzerine algoritmalar konusunda birçok patenti mevcuttur.

Kaynak:
http://www.sciencedirect.com/science/article/pii/S0167404803005078

Eliptik Eğri Kriptografi
-Kullanım Yararları-

- Eliptik Eğri Kriptografi tabanlı şifreleme algoritmaları, DSA ve RSA parametrelerine nazaran daha küçük parametreler kullanırlar ve eşdeğer güvenlik seviyesini sağlarlar (Bakınız: “On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography”:

http://lacal.epfl.ch/files/content/sites/lacal/files/papers/ecdl2.pdf

Eğer izleyen kaynaklardan bir veya birkaçı sınırlı ise Eliptik Eğri Kriptografi kullanın faydası olacaktır:

  - İşlemci Gücü
  - Veri depolama alanı
- Band genişliği
- Güç Tüketimi

Eliptik Eğri Kriptografi is özellikle sınırlı çevre birimlerine sahip akıllı kart, mobil cihazlar, el bilgisayarları ve diğer benzeri sistemler için tercih edilebilir.

Eliptik Eğri Kriptografi

- **Standartlar:** ANS X9F1 | CRYPTREC | IEEE P1363 | NESSIE | NSA Suite B

- Netscape Security Services (NSS), yazılım kütüphanesi kümesi olup, birçok platform bağımsız istemci-server uygulamalarını desteklemektedir. NSS içinde desteği bulunanlar: SSL v2 and v3, TLS, PKCS #5, PKCS #7, PKCS #11, PKCS #12, S/MIME, X.509 v3 sertifikaları ve diğer güvenlik ile ilgili standartlar.

Eliptik Eğri Kriptografi

- Open SSL and NSS (sürüm 3.8) Eliptik Eğri Kriptografi algoritmalarını desteklemektedir.

- Mozilla web tarayıcı Eliptik Eğri Kriptografi algoritmalarını SSL sayesinde desteklemektedir.

- E-posta imzalama/doğrulama için ECDSA kullanılabilir (Digital Signature Algorithm (DSA), Eliptik Eğri Kriptografi sürümü)

Asimetrik Sistemlerin Karşılaştırılması

	Şifreleme	İmzalama	Anahtar Paylaşımı
RSA	Evet	Evet	Evet
Diffie-Hellman	Hayır	Hayır	Evet
DSA	Hayır	Evet	Hayır
Eliptik Eğriler	Evet	Evet	Evet

Açık Anahtarlı Şifreleme ile Simetrik Şifreleme Anahtarı Paylaşımı

- Simetrik Şifreleme için, gizli anahtar (secret key) mutlaka haberleşecek iki kişi arasında güvenli bir şekilde paylaşılmalıdır (örneğin Ayşe and Bülent arasında).
- Bir yaklaşım "Diffie-Hellman anahtar değişimi algoritmasını" kullanmak. Yaygın kullanılıyor fakat basit halinde kullanımı kimlik doğrulama olmadığı için kullanılmalıdır.

Diğer bir güçlü yaklaşım açık anahtar sertifikaları (alıcının açık anahtarı edinmenin yolu) kullanmaktır:

Bülent, Ayşe ile güvenli haberleşmede bulunmak ister. Bunun için Bülent izleyen yolları takip eder:

1) Mesaj M yi hazırlar

2) Bu mesajı belirlediği gizli anahtar K (belki bir kere kullanmak üzere) ile simetrik bir şifreleme sistemi ile şifreler (AES-192 gibi): $E_K(M)=C$

3) Ek olarak gizli anahtar K yı, Ayşe'nin açık anahtarı PU_a ile bir açık anahtar şifreleme algoritması ile şifreler: $E_{PUa}(K)=K'$

4) Bülent K' ve C *yi* Ayşe'ye gönderir.

Ayşe $D_{PRa}[K']=D_{PRa}[E_{PUa}(K)]=K$ elde eder ve ayrıca M yi *hesaplar:* $D_K(C)=M$

Güvenlik Hedeflerine Ulaşmanı Yolu: Kriptografi

- Gizlilik hedefi: Simetrik şifreleme (büyük dosyalar) ve Asimetrik Şifreleme (küçük mesajlar)
- Bütünlük hedefi: Mesaj Kimlik doğrulama (MAC) algoritmaları, Özet fonksiyonları
- İnkar edilememezlik hedefi: Dijital imza ve X.509 sertifikaları (dolayısıyla Asimetrik şifreleme, özet fonksiyonları ve Açık anahtarlı Altyapısı)
  - Kimlik Denetimi hedefi: Mesaj Kimlik doğrulama (MAC) algoritmaları, Özet fonksiyonları ve dijital imza
  - Süreklilik hedefi: Erişim ve Yetkilendirme kontrolü, dolaylı yoldan: Mesaj Kimlik doğrulama (MAC) algoritmaları, Özet fonksiyonları ve dijital imza
  - İzlenebilirlik hedefi: İnkar edilememezlik kanıtı için, dijital imza

6.6.1 Kriptolama güvenliği ve Kriptoanaliz.

Şifrelenen metnin ne kadar güvenli olduğu ve çözümlenmesi için yapılacak saldırı tiplerinin neler olduğunun bilinmesi önemlidir. Geleneksel şifreleme yöntemlerine saldırı için iki adet genel yaklaşım mevcuttur.

Kriptoanaliz: Kriptoanalitik saldırılar, algoritmanın özelliği, şifresiz metnin genel karakteristiği hakkındaki bilgilere ve şifresiz metin–şifreli metin çiftinin bazı örneklerine dayanır. Bu saldırı sonucunda kullanılan anahtar veya şifresiz metin, algoritmanın eksikliklerine dayanılarak elde edilmeye çalışılır.

Deneme-Yanılma(Brute-Force Attack) saldırısı: Saldırgan mümkün olan bütün anahtar kombinasyonlarını, şifresiz metin elde edilene kadar şifreli metni çözmek için dener. Ortalama olarak bütün anahtar kombinasyonlarının yarısı başarılı bir saldırı için denenmelidir.

Saldırı Tipi	Kriptoanalist'in bildiği
Sadece Şifreli Metne (ciphertext only)	Kriptolama algoritması Kodu çözülecek şifreli metin (istatistiksel Saldırı, brute force)
Bilinen Düz metin (known plaintext)	Kriptolama algoritması Kodu çözülecek şifreli metin Gizli anahtar ile şifrelenen bir veya daha fazla düz-şifreli metin çifti(Şifreye saldırı için kullanılır.)
Seçilen Düz metin (chosen plaintext)	Kriptolama algoritması Kodu çözülecek şifreli metin Kriptoanalist tarafından seçilen açık metin, bununla birlikte açık metnin gizli anahtar ile üretilen şifreli hali
Şeçilen Şifreli metin (chosen ciphertext)	Kriptolama algoritması Kodu çözülecek şifreli metin Kriptoanalist tarafından seçilen kuvvetle muhtemel şifreli metin ve karşılığı olan, gizli anahtar ile üretilen çözümlenmiş açık metin.
Şeçilen metin (chosen text)	Kriptolama algoritması Kodu çözülecek şifreli metin Kriptoanalist tarafından seçilen açık metin, bununla birlikte açık metnin gizli anahtar ile üretilen şifreli hali Kriptoanalist tarafından seçilen kuvvetle muhtemel şifreli metin ve karşılığı olan, gizli anahtar ile üretilen çözümlenmiş açık metin.

Tablo 6.1: Şifrelenen mesaja karşı yapılan saldırı Tipleri

İki farklı temel yöntem ile şifreler güvenli olabilir.

Mutlak güvenlik

- Bilgisayar gücü ne kadar fazla olursa olsun şifre hiçbir şekilde kırılamaz.

Hesaplamaya bağlı güvenlik

Bir şifreleme algoritması aşağıdaki kriterleri sağlıyor ise hesaplamaya bağlı güvenli(computationally secure) dır.

- Şifrenin kırılmasının maliyeti şifrelenmiş bilginin değerinden fazla ise
- Şifreyi kırmak için gereken zaman, bilginin yaralı ömründen fazla ise.

Hesaplamaya bağlı güvenlikte verilen bilgisayar gücü sınırları(örn. Evrenin yaşından daha fazla hesaplama zamanı gerekir gibi), içinde şifre kırılamaz.

Hesaplamaya bağlı güvenlik için şifreleme algoritması ve kullanılan anahtar uzunluğu önemlidir. Şifreleme algoritmasının kriptoanalist tarafından bilindiği kabul edilerek şifre uzunluğu ve bilgisayarın hesaplama gücüne bağlı olarak şifrelerin çözümleme süreleri Tablo 6.2'de gösterilmiştir. Çözümleme süresi için gerekli olacak zaman hesabı ortalama olarak alternatif şifre sayısının yarısı kadardır. Bilgisayar hesaplama gücünü ise paralel mimarili tasarım ile artırmak mümkün olmaktadır.

Anahtar Uzunluğu(bit)	Alternatif Anahtar Sayısı	1 çözümleme/µs hızında gereken zaman	10^6 çözümleme/µs hızında gereken zaman
24	$2^{24} = 1.6 \times 10^7$	2^{23} µs = 8.4 saniye	8.4 µsaniye
32	$2^{32} = 4.3 \times 10^9$	2^{31} µs = 35.8 dakika	2.15 milisaniye
48	$2^{48} = 2.8 \times 10^{14}$	2^{47} µs = 4.46 yıl	2.35 dakika
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} µs = 1142 yıl	10 saat
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} µs = 5.4×10^{24} yıl	5.4×10^{18} yıl
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} µs = 5.9×10^{36} yıl	5.9×10^{30} yıl
26 karakter permutasyonu	$26! = 4 \times 10^{26}$	2×10^{26} µs = 6.4×10^{12} yıl	6.4×10^6 yıl

Tablo 6.2. : Anahtar uzunluklarına göre hesaplamaya bağlı güvenlik