Açık Anahtarlı Kriptografi ve Uygulamalar

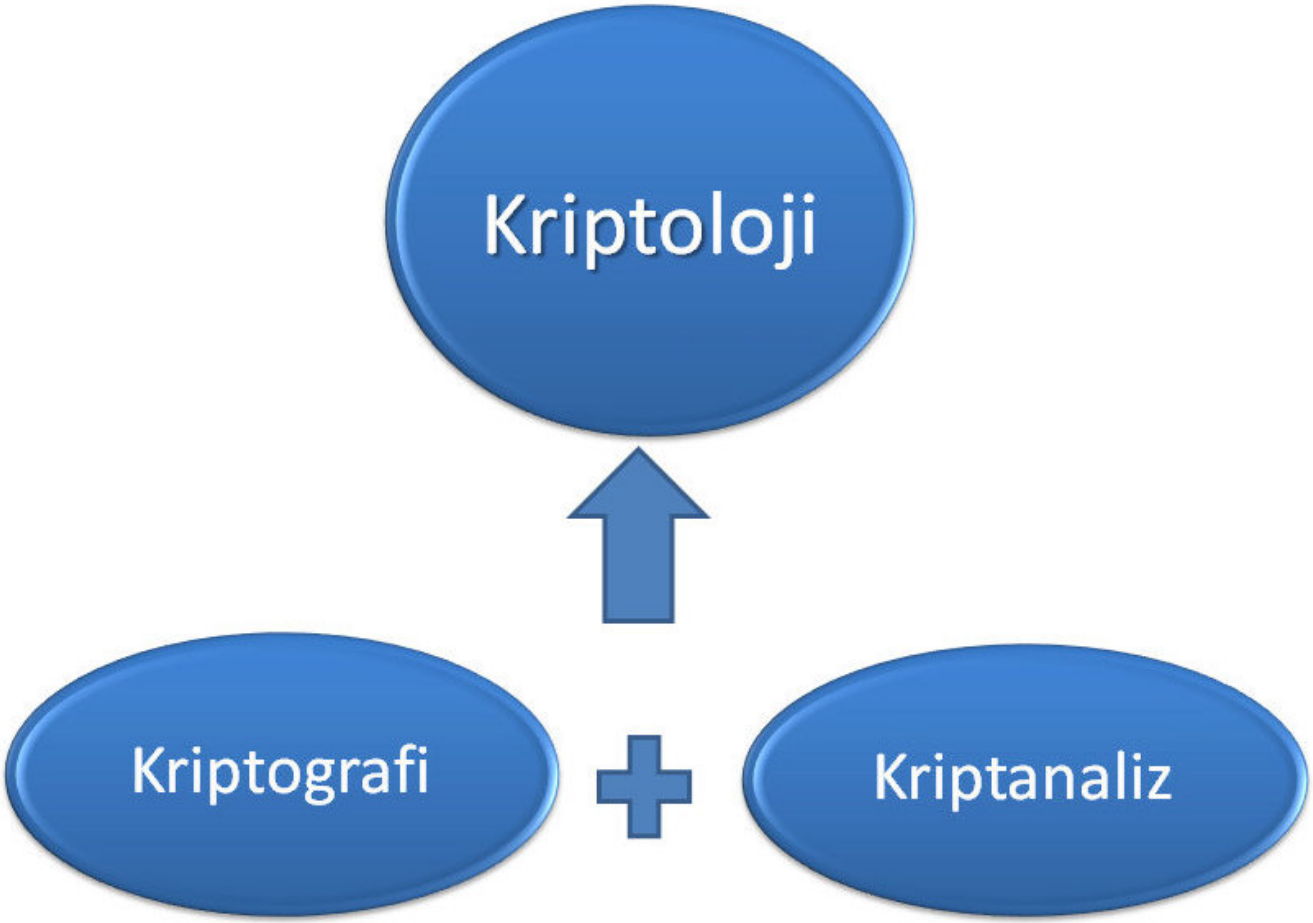
Cihangir TEZCAN

Uygulamalı Matematik Enstitüsü
Kriptografi Bölümü
Orta Doğu Teknik Üniversitesi

SEM Seminerleri

29 Ocak 2013

Temel Kavramlar



Temel Amaçlar

Gizlilik

Bilgi istenmeyen kişiler tarafından anlaşılamamalıdır.

Bütünlük

Bilginin iletilirken hiç değiştirilmemiş olduğu doğrulanmalıdır.

Kimlik Denetimi

Gönderici ve alıcı birbirlerinin kimlikleri doğrulamalıdır.

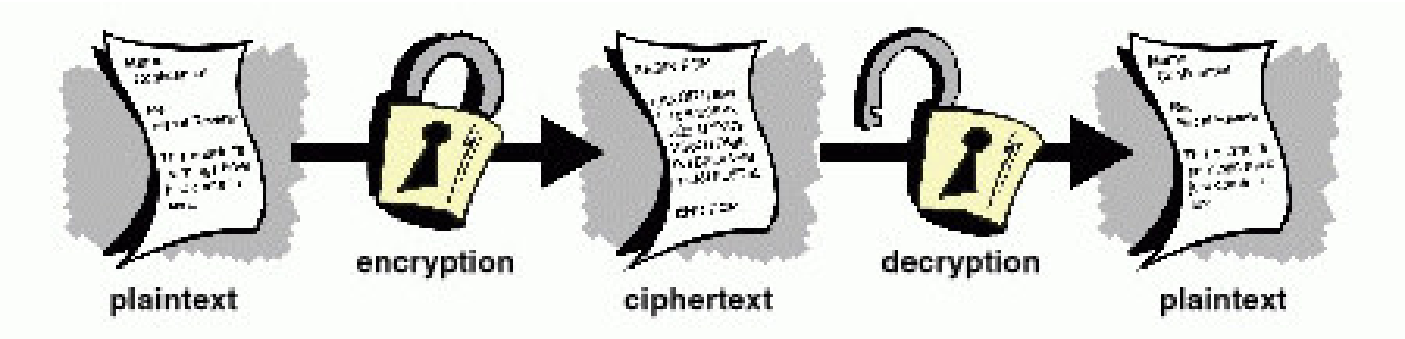
İnkar Edememe

Gönderici bilgiyi gönderdiğini inkar edememelidir.

Kriptosistem/Şifre ne demektir?

Kriptosistem/Şifre ne demektir?

- Korumak istediğiniz şey **düz metin**
- **Şifreli metin** düz metinin şifrelenmiş halidir
- Düz metinden şifreli metin oluşturan ve şifreli metni düz metne geri dönüştüren algoritmalara **kriptosistem/şifre** denir
- Şifreli metin *rastgele* (random) karakterler dizisi gibi gözükmelidir



Kerkckhoffs Prensibi

Kerkckhoffs Prensibi (1883)

Şifre gizli tutulmak zorunda olmamalıdır ve şifrenin düşman eline geçmesi hiçbir sıkıntı oluşturmamalıdır.

Yani, sistemin güvenliği tamamiyle *anahtarın* gizli tutulmasına bağlı olmalıdır.

Claude Shannon

The enemy knows the system.

3 B's of Cryptography

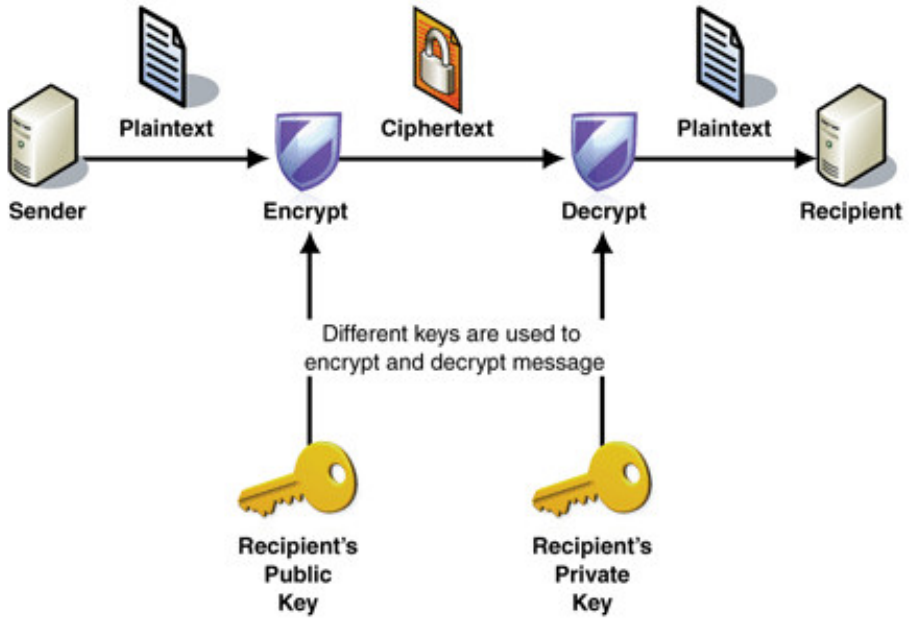
Bribe, Burglary, Blackmail

Simetrik Kriptosistemler

- Şifreleme ve deşifreleme için kullanılan anahtarlar aynı ya da birbiriyle yakın ilişkilidir
- Yani, anahtarın biri, diğer anahtardan kolaylıkla elde edilebilmelidir (in polynomial time)

Asimetrik/Açık Anahtarlı Kriptosistemler

- Şifreleme anahtarı herkesin ulaşabileceği şekilde açıktadır
- Deşifreleme anahtarını elde etmek **zordur**



Asimetrik/Açık Anahtarlı Kriptosistemler

İlgili Dersler

- **MATH 365** Elementary Number Theory I
- **MATH 368** Field Extensions and Galois Theory
- **MATH 473** Ideals, Varieties and Algorithms
- **MATH 476** Algebraic Curves
- **MATH 523** Algebraic Number Theory
- **MATH 551** Algebraic Geometry

Özet Fonksiyonlar

Özet Fonksiyonlar

- Herhangi uzunluktaki bir girdiden, sabit uzunlukta bir çıktı (özet) üreten fonksiyonlardır

Özet Fonksiyonlar

Özet Fonksiyonlar

- Herhangi uzunluktaki bir girdiden, sabit uzunlukta bir çıktı (özet) üreten fonksiyonlardır
- İki farklı girdiden aynı çıktıyı üretmek (çakışma elde etmek) zor olmalıdır. Yani $H(x) = H(x')$ eşitliğini sağlayacak herhangi x ve x' değeri bulmak zor olmalıdır (collision)

Özet Fonksiyonlar

Özet Fonksiyonlar

- Herhangi uzunluktaki bir girdiden, sabit uzunlukta bir çıktı (özet) üreten fonksiyonlardır
- İki farklı girdiden aynı çıktıyı üretmek (çakışma elde etmek) zor olmalıdır. Yani $H(x) = H(x')$ eşitliğini sağlayacak herhangi x ve x' değeri bulmak zor olmalıdır (collision)
- Verilen bir özet değerini verecek bir girdi bulmak zor olmalıdır (pre-image)

Özet Fonksiyonlar

Özet Fonksiyonlar

- Herhangi uzunluktaki bir girdiden, sabit uzunlukta bir çıktı (özet) üreten fonksiyonlardır
- İki farklı girdiden aynı çıktıyı üretmek (çakışma elde etmek) zor olmalıdır. Yani $H(x) = H(x')$ eşitliğini sağlayacak herhangi x ve x' değeri bulmak zor olmalıdır (collision)
- Verilen bir özet değerini verecek bir girdi bulmak zor olmalıdır (pre-image)
- Verilen bir girdi ve karşılık gelen özeti için, aynı özeti verecek ikinci bir girdi bulmak zor olmalıdır (second pre-image)

Özet Fonksiyonlar

Bazı Özet Fonksiyonlar

- MD4 (Ron Rivest, 1990lar) **kırıldı**
- MD5 (Ron Rivest, 1990lar) **kırıldı**
- SHA-0, SHA-1 (NSA, 1990lar, MD4/MD5 tabanlı) **kırıldı**
- SHA-2 (NSA)
- Keccak (Ekim 2012, NIST yarışması kazananı)

Özet Fonksiyonlar

Bazı Özet Fonksiyonlar

- MD4 (Ron Rivest, 1990lar) **kırıldı**
- MD5 (Ron Rivest, 1990lar) **kırıldı**
- SHA-0, SHA-1 (NSA, 1990lar, MD4/MD5 tabanlı) **kırıldı**
- SHA-2 (NSA)
- Keccak (Ekim 2012, NIST yarışması kazananı)
  - 64 başvuru, 51'i ilk aşamaya yükseldi, 3'ü Türk
  - Shamata (Orhun Kara), **pratik olarak kırıldı**
  - Sarmal (Onur Özen, Kerem Varıcı, Çelebi Kocair), **teorik zayıflık**
  - Hamsi (Özgül Küçük), ilk 14'e kaldı ama **çok yavaş**

Doğum Günü Paradoksu

Doğum Günü Paradoksu

- Bir odada 2 kişinin doğum gününün aynı olma ihtimalinin %50'den fazla olması için odada en az kaç kişi bulunmalıdır?

Doğum Günü Paradoksu

Doğum Günü Paradoksu

- Bir odada 2 kişinin doğum gününün aynı olma ihtimalinin %50'den fazla olması için odada en az kaç kişi bulunmalıdır?
- **Doğru cevap:** 23

Doğum Günü Paradoksu

Doğum Günü Paradoksu

- Bir odada 2 kişinin doğum gününün aynı olma ihtimalinin %50'den fazla olması için odada en az kaç kişi bulunmalıdır?
- **Doğru cevap:** 23
- Yani *yaklaşık olarak* bir yıldaki günlerin karekökü. Aynı cevap bir özet fonksiyonda çakışma bulmak için geçerlidir
- Bu yüzden özet uzunluğu en az 128-bit olmalıdır.

Asimetrik Kriptosistemler

Asimetrik Kriptosistemler

- 1976 yılında Diffie ve Hellman tarafından önerilmiştir
- Daha önceden hiç görüşmemiş kişilerin güvenli şekilde haberleşmesini sağlar
- Kimlik doğrulama ve inkar edememe problemlerine çözüm sunar
- Anahtar dağıtımı problemine çözüm sunar

Asimetrik Kriptosistemler

Asimetrik Kriptosistemler

- 1976 yılında Diffie ve Hellman tarafından önerilmiştir
- Daha önceden hiç görüşmemiş kişilerin güvenli şekilde haberleşmesini sağlar
- Kimlik doğrulama ve inkar edememe problemlerine çözüm sunar
- Anahtar dağıtımı problemine çözüm sunar
- Şifreleme ve deşifreleme için farklı anahtarlar kullanır
- Deşifreleme anahtarını, şifreleme anahtarından elde etmek mümkün değildir
- Şifreleme anahtarını herkese duyurmakta sakınca yoktur, bu sayede herkes size şifreli mesaj atabilir
- Çoğu asimetrik kriptosistemin güvenliği matematiksel problemlere dayanır
- Bilinen tüm asimetrik kriptosistemler işlemci zamanı ve bandwidth açısından çok pahalıdır

RSA

RSA (Rivest-Shamir-Adleman)

- Çok büyük (en az 512-bitlik) iki p ve q asal sayısı üretilir, $n = pq$ diyelim

RSA

RSA (Rivest-Shamir-Adleman)

- Çok büyük (en az 512-bitlik) iki p ve q asal sayısı üretilir, $n = pq$ diyelim
- $ed \equiv 1 \bmod (p-1)(q-1)$ denkliğini sağlayan e ve d sayıları seçilir ($e = 65537$ şifreleme işlemlerini kolaylaştırdığı için tavsiye edilir)

RSA

RSA (Rivest-Shamir-Adleman)

- Çok büyük (en az 512-bitlik) iki p ve q asal sayısı üretilir, $n = pq$ diyelim
- $ed \equiv 1 \bmod (p-1)(q-1)$ denkliğini sağlayan e ve d sayıları seçilir ($e = 65537$ şifreleme işlemlerini kolaylaştırdığı için tavsiye edilir)
- Açık anahtar: $< e, n >$, gizli anahtar: $< d, n >$

RSA

RSA (Rivest-Shamir-Adleman)

- Çok büyük (en az 512-bitlik) iki p ve q asal sayısı üretilir, $n = pq$ diyelim
- $ed \equiv 1 \bmod (p-1)(q-1)$ denkliğini sağlayan e ve d sayıları seçilir ($e = 65537$ şifreleme işlemlerini kolaylaştırdığı için tavsiye edilir)
- Açık anahtar: $< e, n >$, gizli anahtar: $< d, n >$
- Mesaj m'i şifrelemek için $c = m^e \bmod n$, deşifrelemek için $m = c^d \bmod n$ işlemleri uygulanır

RSA

RSA (Rivest-Shamir-Adleman)

- Çok büyük (en az 512-bitlik) iki p ve q asal sayısı üretilir, $n = pq$ diyelim
- $ed \equiv 1 \bmod (p-1)(q-1)$ denkliğini sağlayan e ve d sayıları seçilir ($e = 65537$ şifreleme işlemlerini kolaylaştırdığı için tavsiye edilir)
- Açık anahtar: $< e, n >$, gizli anahtar: $< d, n >$
- Mesaj m'i şifrelemek için $c = m^e \bmod n$, deşifrelemek için $m = c^d \bmod n$ işlemleri uygulanır
- Sistemin güvenliği *çoğunlukla* n sayısının çarpanlarına ayrılmasına dayanır

RSA

RSA (Rivest-Shamir-Adleman)

- Çok büyük (en az 512-bitlik) iki p ve q asal sayısı üretilir, $n = pq$ diyelim
- $ed \equiv 1 \bmod (p-1)(q-1)$ denkliğini sağlayan e ve d sayıları seçilir ($e = 65537$ şifreleme işlemlerini kolaylaştırdığı için tavsiye edilir)
- Açık anahtar: $< e, n >$, gizli anahtar: $< d, n >$
- Mesaj m'i şifrelemek için $c = m^e \bmod n$, deşifrelemek için $m = c^d \bmod n$ işlemleri uygulanır
- Sistemin güvenliği *çoğunlukla* n sayısının çarpanlarına ayrılmasına dayanır
- Bu tarz büyük asal sayılar bulmak kolaydır ama n'i çarpanlarına ayırmanın zor olduğuna **inanılır**

Çarpanlara Ayırmak Neden Zor?

Çarpanlara Ayırmak Neden Zor?

- Bazı örnekler
  - $5283065753709209 = 59957 \times 88114244437$: 1930larda 25 dakikada

Çarpanlara Ayırmak Neden Zor?

Çarpanlara Ayırmak Neden Zor?

- Bazı örnekler
  - $5283065753709209 = 59957 \times 88114244437$: 1930larda 25 dakikada
  - $2^{128} + 1$, 39 basamaklı: 1970'de IBM bilgisayar ile bir kaç saat içerisinde

Çarpanlara Ayırmak Neden Zor?

Çarpanlara Ayırmak Neden Zor?

- Bazı örnekler
  - $5283065753709209 = 59957 \times 88114244437$: 1930larda 25 dakikada
  - $2^{128} + 1$, 39 basamaklı: 1970'de IBM bilgisayar ile bir kaç saat içerisinde
  - 100 basamaklı sayılar: 1988'de İnternet üzerinden 2 haftada içerisinde

Çarpanlara Ayırmak Neden Zor?

Çarpanlara Ayırmak Neden Zor?

- Bazı örnekler
  - $5283065753709209 = 59957 \times 88114244437$: 1930larda 25 dakikada
  - $2^{128} + 1$, 39 basamaklı: 1970'de IBM bilgisayar ile bir kaç saat içerisinde
  - 100 basamaklı sayılar: 1988'de İnternet üzerinden 2 haftada içerisinde
  - 200 basamaklı sayılar: Çok büyük bilgisayar ağları kullanarak aylar sürer
  - 300 basamaklı sayılar: Günümüzde güvenliler

Çarpanlara Ayırmak Neden Zor?

Çarpanlara Ayırmak Neden Zor?

- Bazı örnekler
  - $5283065753709209 = 59957 \times 88114244437$: 1930larda 25 dakikada
  - $2^{128} + 1$, 39 basamaklı: 1970'de IBM bilgisayar ile bir kaç saat içerisinde
  - 100 basamaklı sayılar: 1988'de İnternet üzerinden 2 haftada içerisinde
  - 200 basamaklı sayılar: Çok büyük bilgisayar ağları kullanarak aylar sürer
  - 300 basamaklı sayılar: Günümüzde güvenliler
- Çarpanlara ayırmak neden zor: **Kimse bilmiyor**
- Mevcut algoritmalar yeterince hızlı değil
- En hızlı algoritma: Number Field Sieve (1989)
- Kuantum bilgisayarlarında çarpanlara ayırmak çok kolay!!!

Çarpanlara Ayırmak Neden Zor?

RSA Number	Decimal digits	Binary digits	Cash prize offered	Factored on	Factored by
RSA-100	100	330		April 1991	Arjen K. Lenstra
RSA-110	110	364		April 1992	Arjen K. Lenstra and M.S. Manasse
RSA-120	120	397		June 1993	T. Denny et al.
RSA-129	129	426	$100 USD	April 1994	Arjen K. Lenstra et al.
RSA-130	130	430		April 10, 1996	Arjen K. Lenstra et al.
RSA-140	140	463		February 2, 1999	Herman J. J. te Riele et al.
RSA-150[1]	150	496		April 16, 2004	Kazumaro Aoki et al.
RSA-155	155	512		August 22, 1999	Herman J. J. te Riele et al.
RSA-160	160	530		April 1, 2003	Jens Franke et al., University of Bonn
RSA-170	170	563		*open*	
RSA-576	174	576	$10,000 USD	December 3, 2003	Jens Franke et al., University of Bonn
RSA-180	180	596		*open*	
RSA-190	190	629		*open*	
RSA-640	193	640	$20,000 USD	November 2, 2005	Jens Franke et al., University of Bonn
RSA-200	200	663		May 9, 2005	Jens Franke et al., University of Bonn

Problemler

Problemler

1. Şifreleme işlemleri çok masraflı, tarafların işlem gücü, çok fazla şifreleme işlemi yapmak için yeterli değil
2. Mesajın düşündüğümüz kişi tarafından yollandığından nasıl emin olabiliriz?
3. Mesajın kimden geldiğinden emin olunsa bile, başka birine bunu ispatlayabilir miyiz?

Elektronik İmza

Elektronik İmza

- RSA tersten kullanılabilir: şifreleme gizli anahtarla, deşifreleme açık anahtarla yapılabilir

Elektronik İmza

Elektronik İmza

- RSA tersten kullanılabilir: şifreleme gizli anahtarla, deşifreleme açık anahtarla yapılabilir
- Bu sayede *elektronik imza* elde edilir: sadece gizli anahtar sahibi mesajlarını imzalayabilir, herkes imzanın doğruluğunu kontrol edebilir

Elektronik İmza

Elektronik İmza

- RSA tersten kullanılabilir: şifreleme gizli anahtarla, deşifreleme açık anahtarla yapılabilir
- Bu sayede *elektronik imza* elde edilir: sadece gizli anahtar sahibi mesajlarını imzalayabilir, herkes imzanın doğruluğunu kontrol edebilir
- Mesajının tamamını imzalamak çok zahmetlidir, bu yüzden mesajın sadece *özeti* imzalanır

Elektronik İmza

Elektronik İmza

- RSA tersten kullanılabilir: şifreleme gizli anahtarla, deşifreleme açık anahtarla yapılabilir
- Bu sayede *elektronik imza* elde edilir: sadece gizli anahtar sahibi mesajlarını imzalayabilir, herkes imzanın doğruluğunu kontrol edebilir
- Mesajının tamamını imzalamak çok zahmetlidir, bu yüzden mesajın sadece *özeti* imzalanır
- Bu sayede inkar edememe (non-repudiation) problemi çözülmüş olur

Elektronik İmza

Elektronik İmza

- RSA tersten kullanılabilir: şifreleme gizli anahtarla, deşifreleme açık anahtarla yapılabilir
- Bu sayede *elektronik imza* elde edilir: sadece gizli anahtar sahibi mesajlarını imzalayabilir, herkes imzanın doğruluğunu kontrol edebilir
- Mesajının tamamını imzalamak çok zahmetlidir, bu yüzden mesajın sadece *özeti* imzalanır
- Bu sayede inkar edememe (non-repudiation) problemi çözülmüş olur
- **Not:** Aslında pratikte işler bu kadar da kolay değil

Ayrık Logaritma Problemi

Ayrık Logaritma Problemi

- $Z_p^* = \{1, 2, \ldots, p-1\}$

Ayrık Logaritma Problemi

Ayrık Logaritma Problemi

- $Z_p^* = \{1, 2, \ldots, p-1\}$
- Z_p^*'da g ve y verilmiş ve $g^x = y \bmod p$ ise, x kaçtır?

Ayrık Logaritma Problemi

Ayrık Logaritma Problemi

- $Z_p^* = \{1, 2, \ldots, p-1\}$
- Z_p^*'da g ve y verilmiş ve $g^x = y \bmod p$ ise, x kaçtır?
- $x = \log_g y$ çözülmesi zor bir problemdir.
- En iyi algoritma: Function Field Sieve (Adleman 1994)

Ayrık Logaritma Problemi

Ayrık Logaritma Problemi

- $Z_p^* = \{1, 2, \ldots, p-1\}$
- Z_p^*'da g ve y verilmiş ve $g^x = y \bmod p$ ise, x kaçtır?
- $x = \log_g y$ çözülmesi zor bir problemdir.
- En iyi algoritma: Function Field Sieve (Adleman 1994)

Example

Z_{17}'de $g = 3$, $y = 11$, $3^x = 11 \bmod 17$ is x kaçtır?

Ayrık Logaritma Problemi

Ayrık Logaritma Problemi

- $Z_p^* = \{1, 2, \ldots, p-1\}$
- Z_p^*'da g ve y verilmiş ve $g^x = y \bmod p$ ise, x kaçtır?
- $x = \log_g y$ çözülmesi zor bir problemdir.
- En iyi algoritma: Function Field Sieve (Adleman 1994)

Example

Z_{17}'de $g = 3$, $y = 11$, $3^x = 11 \bmod 17$ is x kaçtır?

Ne kadar zor?

- Problemin zorluğu üzerinde çalışılan gruba bağlıdır.
- örn: Function Field Sieve gibi Index Calculus Algoritmaları eliptik eğrilerden elde edilen gruplarda geçerli değildir.

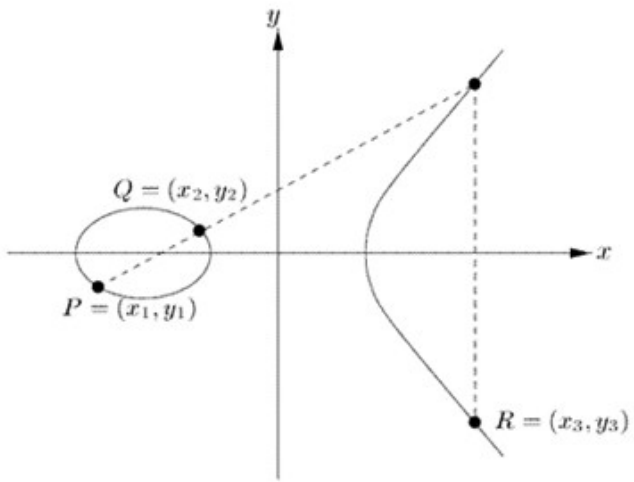
Eliptik Eğriler

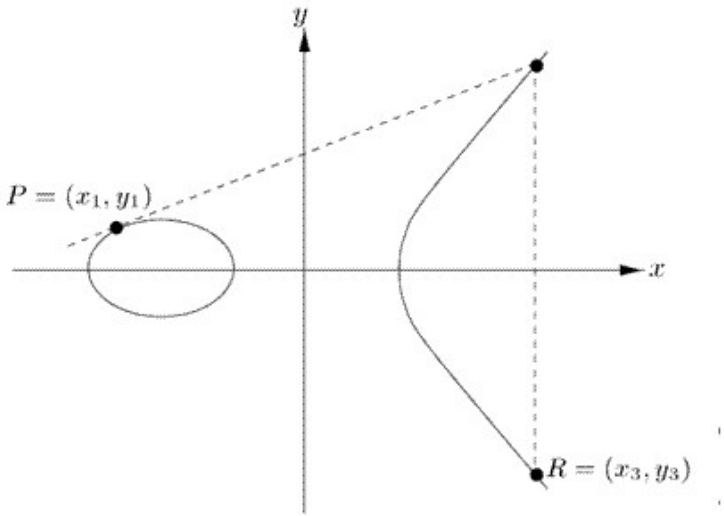
Eliptik Eğriler

- $E(F_p)$'ye ait tüm bu noktalar toplamsal grup oluşturur.
- Bu grubun aritmetiğinde iki temel işlem tanımlanır:

Toplama : $P \neq Q \Rightarrow P + Q = R$

İki kat alma : $2P = R$





Diffie-Hellman Anahtar Değişimi

Diffie-Hellman Anahtar Değişimi

A ve B kişileri ortak bir anahtar kullanmak için sırasıyla şu işlemleri yaparlar

1. g ve p sayıları belirlenir (herkese açık bilgi)

Diffie-Hellman Anahtar Değişimi

Diffie-Hellman Anahtar Değişimi

A ve B kişileri ortak bir anahtar kullanmak için sırasıyla şu işlemleri yaparlar

1. g ve p sayıları belirlenir (herkese açık bilgi)
2. A rastgele bir x seçer ve $X = g^x$'i hesaplayıp, X değerini B'ye gönderir

Diffie-Hellman Anahtar Değişimi

Diffie-Hellman Anahtar Değişimi

A ve B kişileri ortak bir anahtar kullanmak için sırasıyla şu işlemleri yaparlar

1. g ve p sayıları belirlenir (herkese açık bilgi)
2. A rastgele bir x seçer ve $X = g^x$'i hesaplayıp, X değerini B'ye gönderir
3. B rastgele bir y seçer ve $Y = g^y$'i hesaplayıp, Y değerini A'ye gönderir

Diffie-Hellman Anahtar Değişimi

Diffie-Hellman Anahtar Değişimi

A ve B kişileri ortak bir anahtar kullanmak için sırasıyla şu işlemleri yaparlar

1. g ve p sayıları belirlenir (herkese açık bilgi)
2. A rastgele bir x seçer ve $X = g^x$'i hesaplayıp, X değerini B'ye gönderir
3. B rastgele bir y seçer ve $Y = g^y$'i hesaplayıp, Y değerini A'ye gönderir
4. A $K = Y^x$ değerini, B de $K = X^y$ değerini hesaplar

Diffie-Hellman Anahtar Değişimi

Diffie-Hellman Anahtar Değişimi

A ve B kişileri ortak bir anahtar kullanmak için sırasıyla şu işlemleri yaparlar

1. g ve p sayıları belirlenir (herkese açık bilgi)
2. A rastgele bir x seçer ve $X = g^x$'i hesaplayıp, X değerini B'ye gönderir
3. B rastgele bir y seçer ve $Y = g^y$'i hesaplayıp, Y değerini A'ye gönderir
4. A $K = Y^x$ değerini, B de $K = X^y$ değerini hesaplar
5. $K = g^{xy}$ ortak anahtar olarak kullanılır

Bu anahtar herhangi bir simetrik kriptosistem anahtarı olarak kullanılabilir.

Asimetrik Sistemlerin Karşılaştırılması

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

Bazı (eski) tablolar

Tablo 1. Kriptografik Algoritmalar ve Sağladıkları Bilgi Güvenliği Kavramları

	Simetrik Şifreler	**Asimetrik Şifreler**	**Anahtarsız Sistemler**
Gizlilik	Sağlar	Sağlar	-
Bütünlük	-	-	Sağlar
Kimlik Denetimi	-	Sağlar	-
İnkar Edememe	-	Sağlar	-
Performans	Hızlı	Yavaş	Hızlı
Önerilen Algoritmalar	DEA, 3DEA, AES	RSA, DSA, ECDSA	SHA serisi, RIPEMD

Bazı (eski) tablolar

Tablo 1. Güvenlik Seviyelerine Uygun Olarak Kullanılan Kriptografik Algoritmalar, Anahtarların bit Uzunlukları ve Son Geçerlilik Tarihleri

Güvenlik Seviyeleri / *Algoritmalar*	Sınıflandırılmamış	Gizli(Kısa Süreli Koruma)	Gizli(Orta Vadeli Koruma)	Gizli(Uzun Süreli Koruma)	Çok Gizli(Askeri Gizlilik Düzeyi)
Simetrik					
2DES	80				
3DES		112			
AES			128	192	256
Asimetrik					
DSA[1]	1024 -160	2048 - 224	3072 - 256	7680 - 384	15360 - 512
RSA	1024	2048	3072	7680	15360
ECDSA	160	224	256	384	512
EC	160	224	256	384	512
Diffie-Hellman	1024	2048	3072	7680	15360
Özet Fonksiyonları					
RIPEMD	160				
SHA		224	256	384	512
Son Geçerlilik Tarihi*	2010	2020	2030	2050	**

[1] İlk değer grubun boyutunu, ikinci değer anahtar uzunluğunu göstermektedir.
* Günümüz koşullarına uygun olarak yaklaşık değerler verilmiştir.
** Kuantum bilgisayarlara karşı iyi koruma sağlamaktadır.

Bazı (eski) tablolar

Tablo 3 RSA Sisteminin Şifreleme Performansı

Algoritma	İşlem/milisaniye	İşlem/megahertz
RSA 1024 bit şifreleme	0.07	0.13
RSA 1024 bit şifre çözme	1.54	2.78
RSA 2048 bit şifreleme	0.15	0.28
RSA 2048 bit şifre çözme	5.95	10.89

Tablo 2 ve 3'de verilen değerler AMD Opteron 2.4 GHz işlemcili bir bilgisayarda cryptopp kütüphanesi kullanılarak elde edilmiştir [1].

Bazı (eski) tablolar

Tablo 4 E-imza Algoritmalarının Performansları

Algoritma	imza sayısı /dakika	Algoritma	imza sayısı /dakika
RSA 1024 bit imzalama		**ECDSA 192 bit imzalama**	
RSA 1024 bit imza doğrulama		**ECDSA 192 bit imza doğrulama**	
RSA 2048 bit imzalama	2940	**ECDSA 224 bit imzalama**	105840
RSA 2048 bit imza doğrulama	26880	**ECDSA 224 bit imza doğrulama**	47520
RSA 3072 bit imzalama	480	**ECDSA 256 bit imzalama**	54000
RSA 3072 bit imza doğrulama	11280	**ECDSA 256 bit imza doğrulama**	22800
RSA 7680 bit imzalama	60	**ECDSA 384 bit imzalama**	30960
RSA 7680 bit imza doğrulama	2160	**ECDSA 384 bit imza doğrulama**	11040
RSA 15360 bit imzalama	60	**ECDSA 521 bit imzalama**	14400
RSA 15360 bit imza doğrulama	480	**ECDSA 521 bit imza doğrulama**	5280

Tablo 4'deki veriler Pentium IV 3.0 GHz işlemcili bir bilgisayarda elde edilmiştir.

Teşekkürler

Sorular?