CENG 434 Kriptoloji – 5. Ders

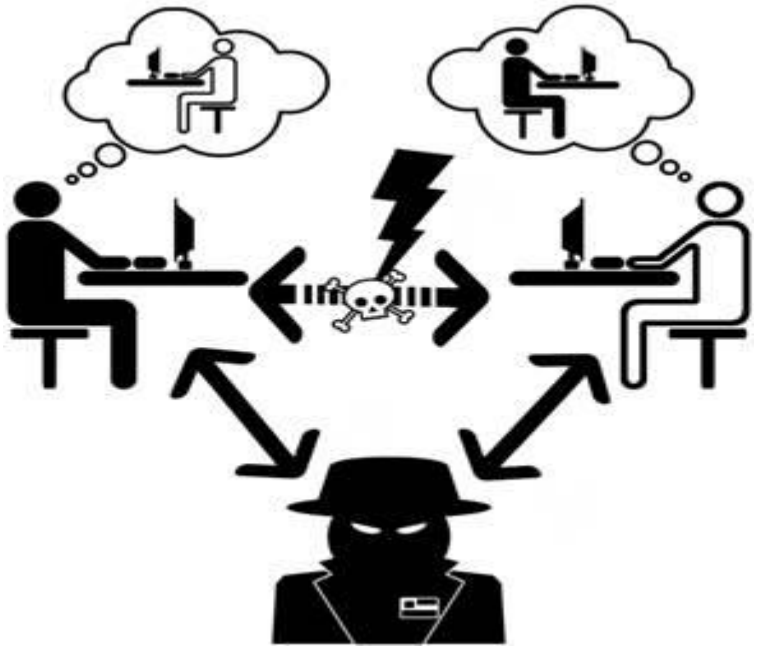
Alper UĞUR

CENG 507 :
KRİPTOGRAFİK ALGORİTMALAR VE SİSTEMLER

CENG 434:
KRİPTOLOJİ

Güvenlik Hizmetleri (Security Services)

- Gizlilik (Confidentiality)
- Bütünlük (Integrity)
- Kimlik doğrulama (Authentication)
- Ulaşılabilirlik (Availability)
- Rededememe (Non-repudiation)



Kimlik doğrulama (Authentication)

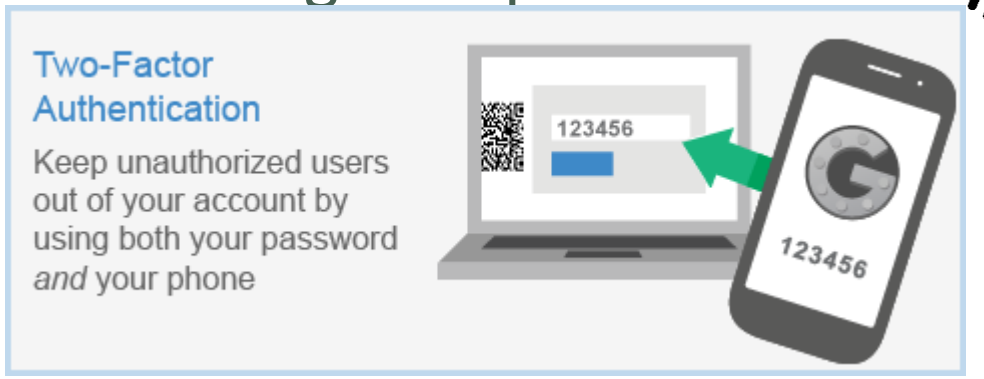
- Varlığın iddia ettiği kimliğini doğrulamak (Who are you,really?)
- Varlığın orijinalliğini doğrulamak (authentic document)

GANDALF ?

Kimlik doğrulama (Authentication)

- Elde edilen (You have)
- Sahip olunan (You own)
- Her ikisi (Both)

- Challenge-Response



"This site wants a two-factor authentication. A retina scan and a urine sample."

Simetrik şifreleme

- Kimlik doğrulama
  - Sadece anahtar sahipleri
  - $C = E_K(P)$
- Mesajın orijinalliği?
- Her anahtarı olan içeriği değiştirip gönderebilir
- Kendisinin oluşturduğunu reddedebilir

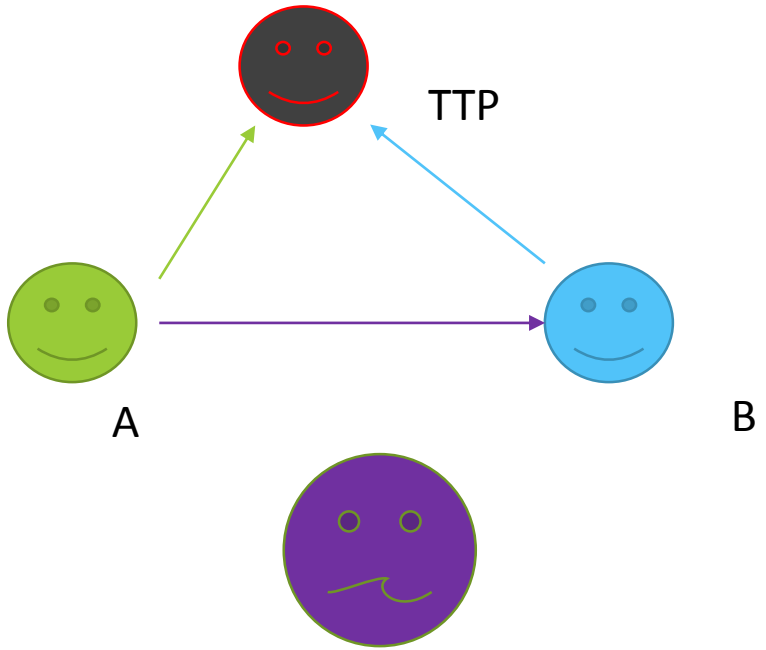
Simetrik şifreleme

- Güvenilir Üçüncü Taraf (Trusted Third Party)
- Mesajın orijinalliği?
- K1: (A,B) K2: (A,C) K3: (B,C)

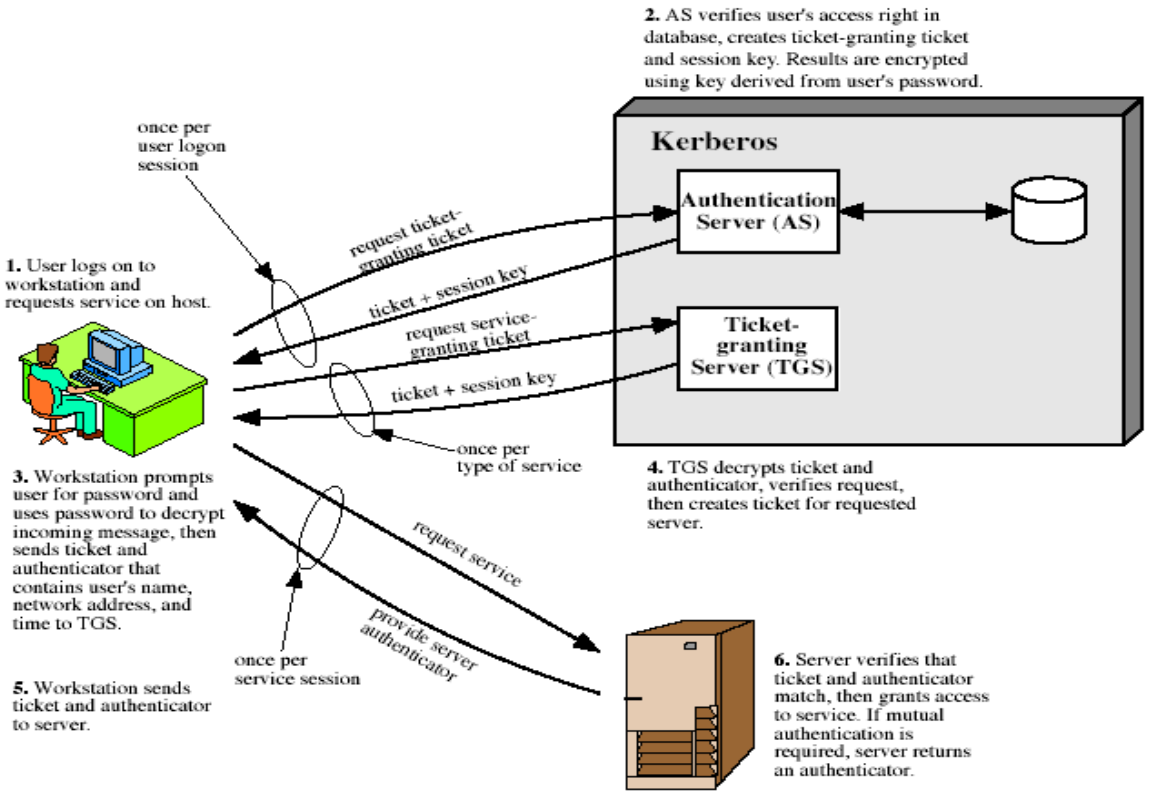
- $E_{K1}(M)$; M, $E_{K1}(M)$

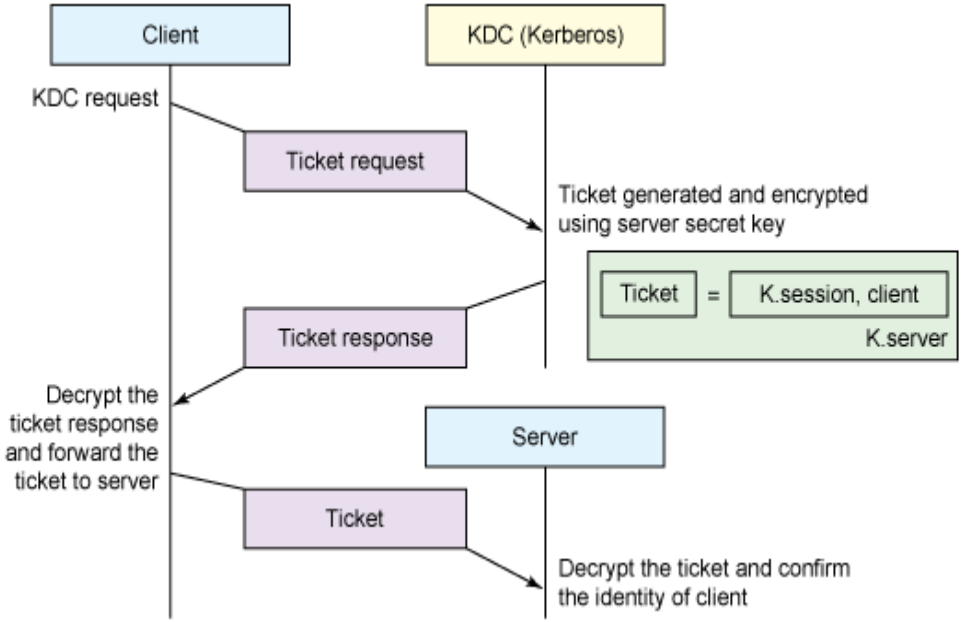
- M, $E_{K1}(M)$; M, $E_{K2}(E_{K1}(M))$

- M, $E_{K2}(M)$

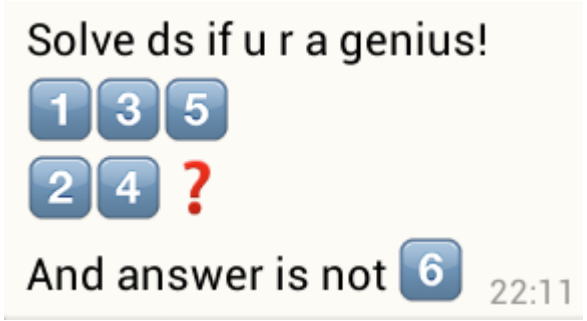


KERBEROS


2. AS verifies user's access right in database, creates ticket-granting ticket and session key. Results are encrypted using key derived from user's password.
Kerberos
Authentication Server (AS)
Ticket-granting Server (TGS)
once per user logon session
request ticket-granting ticket
ticket + session key
request service-granting ticket
ticket + session key
once per type of service
1. User logs on to workstation and requests service on host.
3. Workstation prompts user for password and uses password to decrypt incoming message, then sends ticket and authenticator that contains user's name, network address, and time to TGS.
4. TGS decrypts ticket and authenticator, verifies request, then creates ticket for requested server.
request service
provide server authenticator
once per service session
5. Workstation sends ticket and authenticator to server.
6. Server verifies that ticket and authenticator match, then grants access to service. If mutual authentication is required, server returns an authenticator.


Client
KDC (Kerberos)
KDC request
Ticket request
Ticket generated and encrypted using server secret key
Ticket = K.session, client
K.server
Ticket response
Decrypt the ticket response and forward the ticket to server
Server
Ticket
Decrypt the ticket and confirm the identity of client

Rasgele Sayı Üreteçleri



- Math.rnd(seed);

- Rasgelelilik (Randomness)
  - Tek bir sayıdan bahsetmek yerine, bir dizi sayı söz konusu

  - Düzgün dağılım (Uniform distribution)

  sayıların dağılımı, ortay çıkma sıklıkları
  - Bağımsızlık (Independence)

  Dizideki hiçbir sayı diğerlerinden çıkarım yapılarak tahmin edilemez

Sözde Rasgele Sayı Üretimi Pseudo-random number generators (PRNGs)

«Eskimiş yöntemler»

- Kareortası yöntemi
  1. Başlangıç tohumu (4 basamaklı tamsayı)
  2. Karesini al
  3. Ortasındaki 4 basamaklı sayıyı al
  4. Bu sayıyı yeni Başlangıç tohumu olarak ata
  5. Sayıyı 10.000'e böl.
  6. Sonuç rasgele sayın olacak
  7. Yeni üretmek için 2'ye geri dön.

$s_0 = 5197$

s_1: $5197^2 = 27\underline{0088}09 \rightarrow s_1 = 0088$, $R_1 = 0.0088$

s_2: $0088^2 = 00\underline{0077}44 \rightarrow s_2 = 0077$, $R_2 = 0.0077$

s_3: $0077^2 = 00\underline{0059}29 \rightarrow s_3 = 0059$, $R_3 = 0.0059$

$s_i = 6500$

s_{i+1}: $6500^2 = 42\underline{2500}00 \rightarrow s_{i+1} = 2500$, $R_{i+1} = 0.0088$

s_{i+2}: $2500^2 = 06\underline{2500}00 \rightarrow s_{i+2} = 2500$, $R_{i+1} = 0.0088$

Midsquare method:

1. Start with an initial seed (e.g. a 4-digit integer).
2. Square the number.
3. Take the middle 4 digits.
4. This value becomes the new seed. Divide the number by 10,000. This becomes the random number. Go to 2.

Doğrusal uyumlu üreteçler (Linear congruential generator)

4 tamsayı

- m mod $m > 0$
- a çarpan (katsayı) $0 , 0 < a < m$
- c artım (eklenen) $0, 0 < c < m$
- X_0 başlangıç değeri $0, 0 < X_0 < m$

4 integer

- m the modulus $m > 0$
- a the multiplier $0 , 0 < a < m$
- c the increment $0, 0 < c < m$
- X_0 the starting value $0, 0 < X_0 < m$

The algorithm is

$X_n + 1 = (aX_n + c) \bmod m$

Where n>0

Algoritma: n>0 olmak üzere

$X_{n+1} = (aX_n + c) \bmod m$

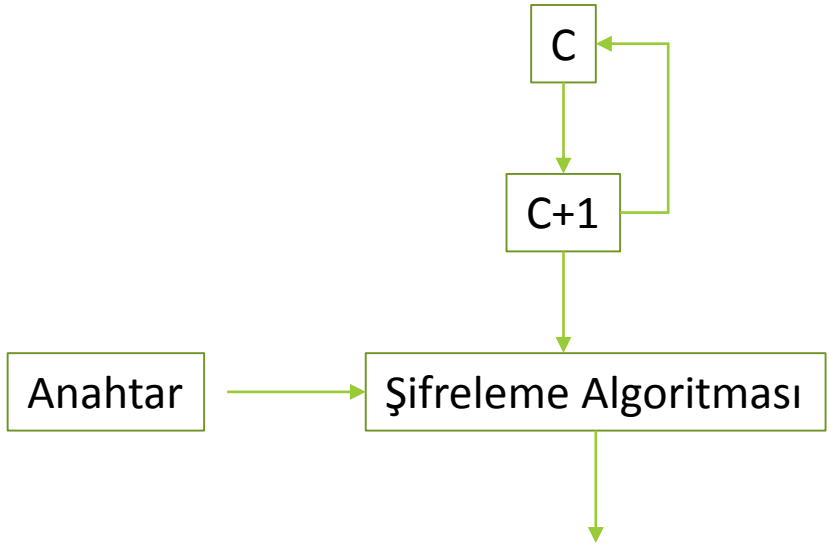
- a=1, c=1 ?
- a=7 c=0 m= 32 X_0 =1
  - {7,17,23,1,7,..}
- a= 5
  - {5,25,29,17,21,9,13,1,5,..}

Doğrusal uyumlu üreteçler (Linear congruential generator)

Lagged Fibonacci generator (LFG)

Blum Shub Shub

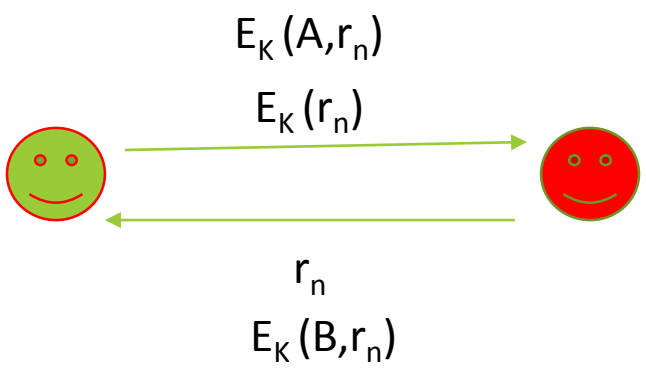
Kriptografik Üreteçler

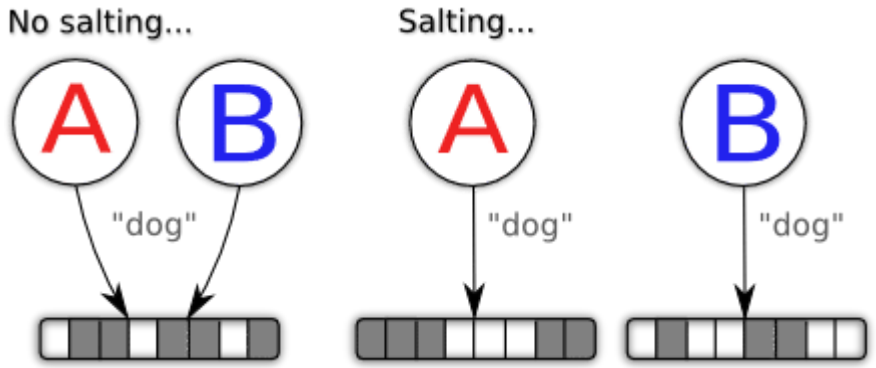


Rasgele Sayı Üreteçleri

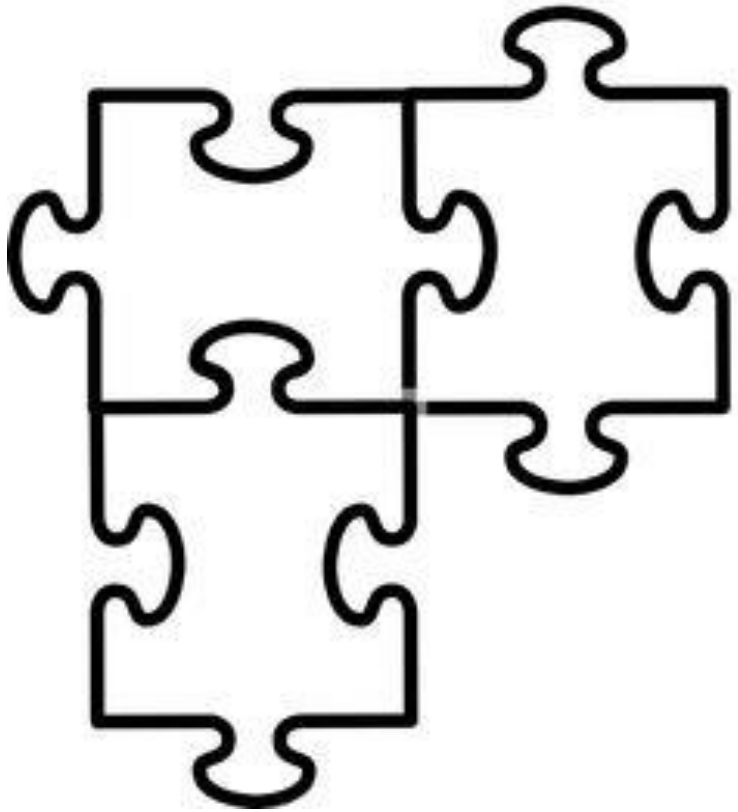
- Kimlik doğrulama
- Meydan okuma-Cevap
- Protokol güvenliği
- «Tuz'la da kokmasın»

Authentication
Challenge-Response
Protocol security
Salting passwords





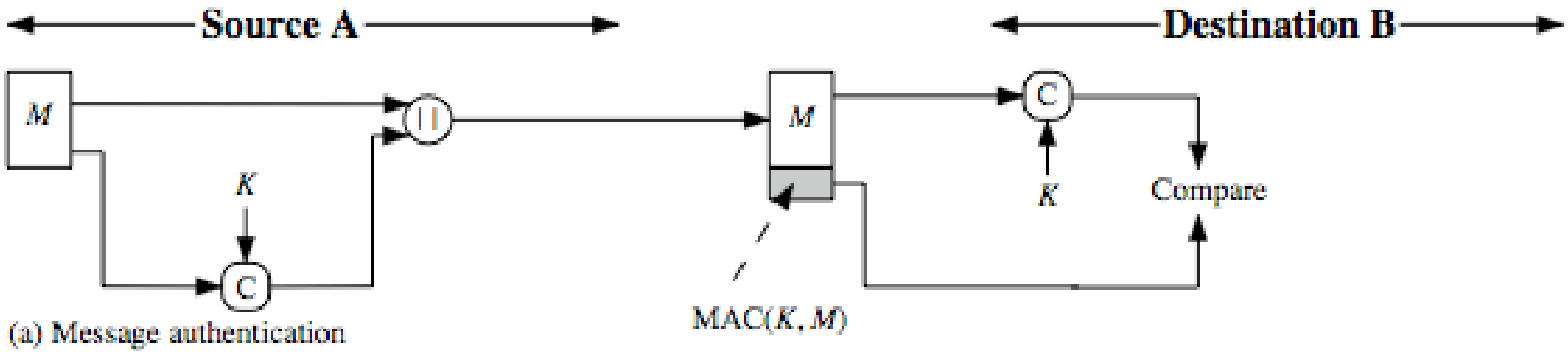
Ara - 10dk



Bütünlük(Integrity)

- MAC (Message Authentication Code)
- MAC = C(K,M)
- Mesaj özeti HASH

a small fixed-sized block of data
generated from message + secret key
MAC = C(K,M)
appended to message when sent

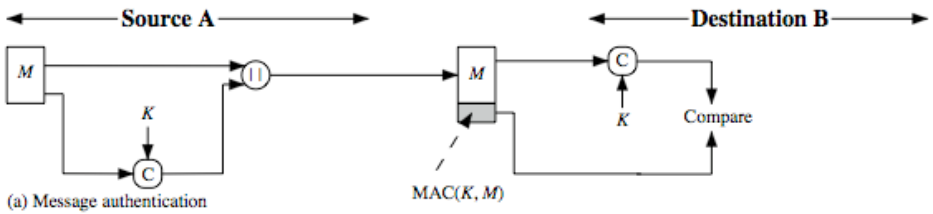


Bütünlük(Integrity)

- Aynı özete sahip başka bir mesaj bulunamamalı
- Düzgün dağılım
- Çığ etkisi
- Tersinir olmayan bir fonksiyon olmalı
- MAC = C(K,M) C^{-1} (MAC) = ~~K,M~~

- N byte -> 256bit

- knowing a message and MAC, is infeasible to find another message with same MAC
- MACs should be uniformly distributed
- MAC should depend equally on all bits of the message



Özetleme Fonksiyonları (Hash Functions)

- Girdinin boyutu sınırlı olmamalı
  (No limitation for the size of input)
- Çıktının boyutu sabit olmalı
  (fixed-length output)
- H(M) fonksiyonu hesaplanması kolay olmalı
  (H(M) easy to calculate, implement)
- **Tek yönlülük:** H(M) =h ise bilinen h de M nin hesaplanması mümkün olmamalı
  One-way: H(M)=h where it is infeasible to computationally find M from h
- **Zayıf çakışma dayanıklılığı (weak collision resistance)**
  - H(M') = H(M) , M' ≠ M
- **Güçlü çakışma dayanıklılığı (strong collision resistance)**
  - (M,M') where H(M)=H(M')



Güvercin yuvası prensibi (Pigeon Hole Principle)

Güvercin Yuvası				

- Güvercin sayısı ve yuva sayısı
- # of pigeon and # of holes

- Boş kalma (any unoccupied?)
- 1+ güvercin yerleşmesi
- (more than one pigeon in one hole)

http://www.matematikdunyasi.org/arsiv/PDF/03-I-50-51-GuvercinYuvasi.pdf

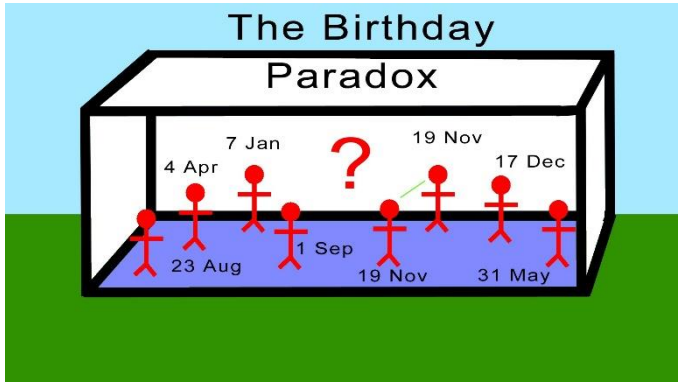
Doğumgünü İkilemi (Birthday Paradox)

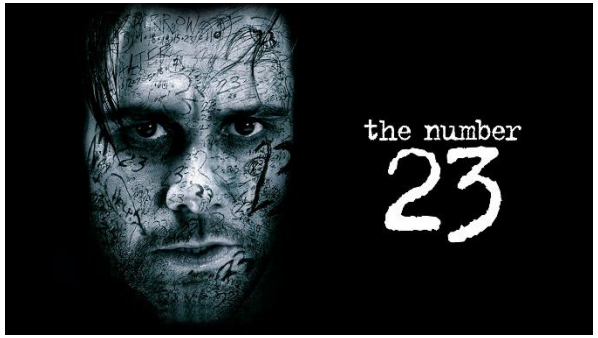
- Bir odada doğumgünü aynı olan iki kişinin olma olasılığı nedir?
(%100 : odadaki kişi sayısı?)
(%50: odadaki kişi sayısı?)

What's the probability for two person having same birthday?
(%100 : # of person in the room?)
(%50: # of person in the room?)

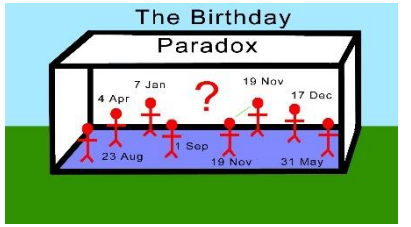
$$22 + 21 + 20 + \cdots + 1 = 253$$

$\binom{n}{2}=\binom{23}{2}$= 23.22/2 =253





Doğumgünü İkilemi (Birthday Paradox)



- Bir odada doğumgünü aynı olan iki kişinin olma olasılığı nedir?
(%100 : odadaki kişi sayısı?)
(%50: odadaki kişi sayısı?)

What's the probability for two person having same birthday?
(%100 : # of person in the room?)
(%50: # of person in the room?)

- P(o)+P(o') =1
- P(o) =1-P(o')

- 365/365 , 364/365, 363/365,…, 1/365

- Örn: 30 kişi
- (365.364.363….336)/ 365^{30}

- = (365!)/(335!)(365^{30})

(335! = (365-30)!)

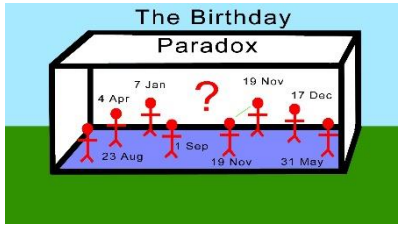
- =0,29

- *P(o') =0,29 P(o) = 1-0,29 =0,71*

365!/(342!)(365^{23}) =0,493
1-0,493=0,51

http://www.geeksforgeeks.org/birthday-paradox/

Doğumgünü İkilemi (Birthday Paradox)

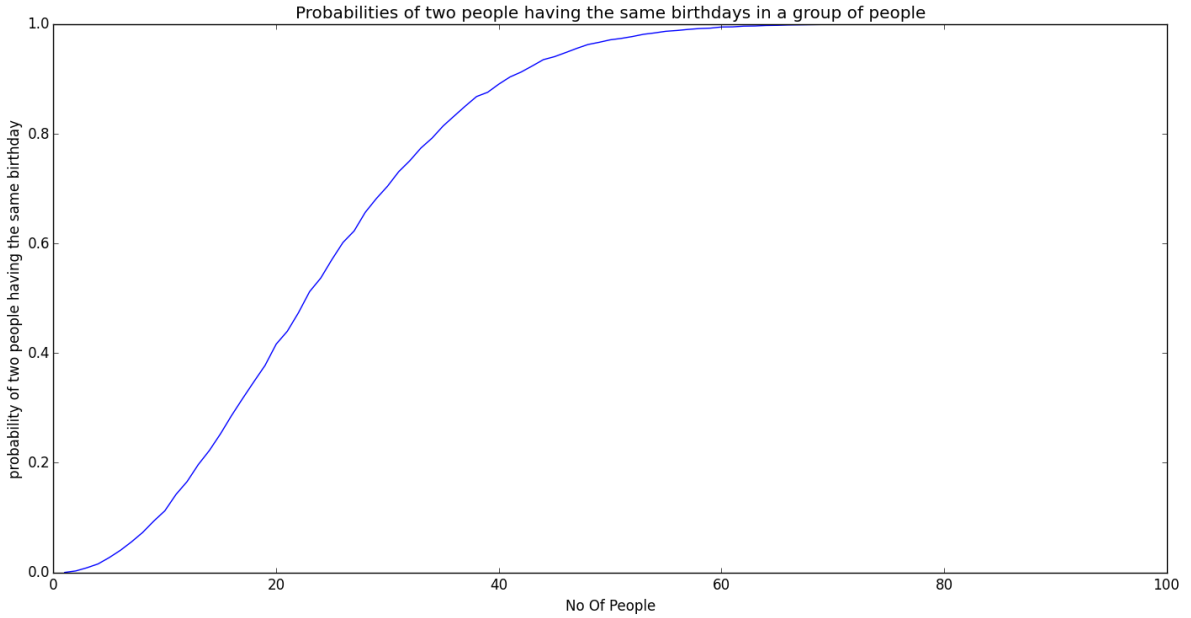


- Bir odada doğumgünü aynı olan iki kişinin olma olasılığı nedir?
(%100 : odadaki kişi sayısı?)
(%50: odadaki kişi sayısı?)

What's the probability for two person having same birthday?
(%100 : # of person in the room?)
(%50: # of person in the room?)

- P(o)+P(o') =1
- P(o) =1-P(o')

- 365/365 , 364/365, 363/365,…, 1/365



$365!/(342!)(365^{23}) = 0{,}493$
$1-0{,}493=0{,}51$

Özetleme Fonksiyonların Güvenliği (Security of Hash Functions)

- Özet uzunluğu yeterli mi?

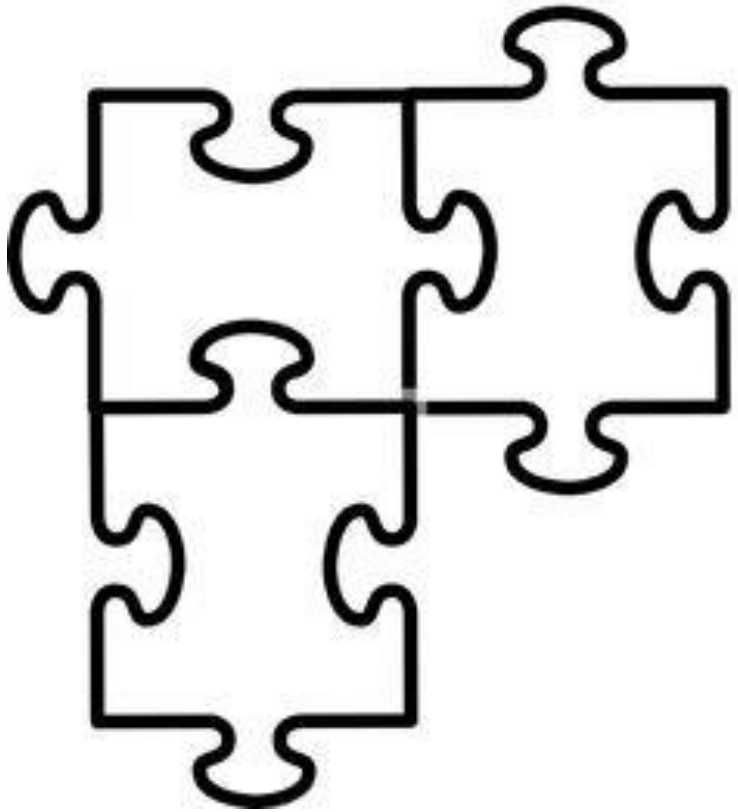
- MD5 sözlük saldırısı (dictionary attack) (16 karakter)
- SHA-1 (128bit)

MD5:5f4dcc3b5aa765d61d8327deb882cf99

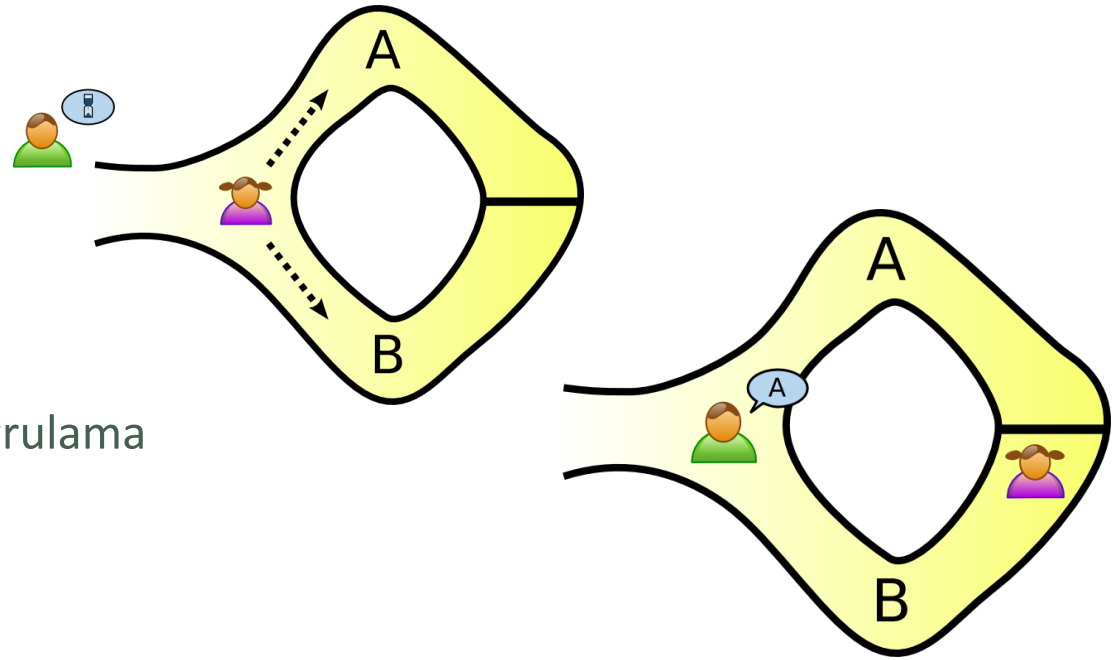
http://md5.gromweb.com/

- SHA-3 Yarışma (hamsi : Özgül Küçük, spectral: Çetin Kaya KOÇ)

Ara - 10dk



Zero Knowledge Proof



- Meydan okuma
- Etkileşimli doğrulama

- Amaç: Başka bir bilgi açığa çıkarmadan bir durumu doğrulama

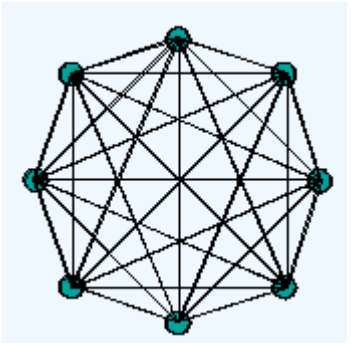
(The goal is to prove a statement without leaking extra information)

- **Bütünlük (Completeness):**
- Eğer sonuç doğru ise doğrulayan yanıltılmayacaktır.
- if the statement is true, the honest verifier (that is, one following the protocol properly) will be convinced of this fact by an honest prover.
- **Geçerlilik(Soundness):**
- Eğer sonuç yanlış ise doğrulayan doğruluğa ikna edilemeyecektir.
  - (Her zaman düşük bir olasılık vardır)
  - if the statement is false, no prover, even if it doesn't follow the protocol, can convince the honest verifier that it is true,
  - except with some small probability

- **Zero-Knowledge:** Sonuç doğru ise doğrulayan bu durumdan ekstra bir bilgi öğrenememelidir.
- If the statement is true, verifier learns anything other than this fact.

Anahtar Yönetimi (Key Management)

- Anahtarın saklanması (Key Storage)
- Anahtarların değişimi (Key Exchange)
- Anahtarların yenilenmesi (Key Renewal)
- Anahtarların iptali (Key Revocation)

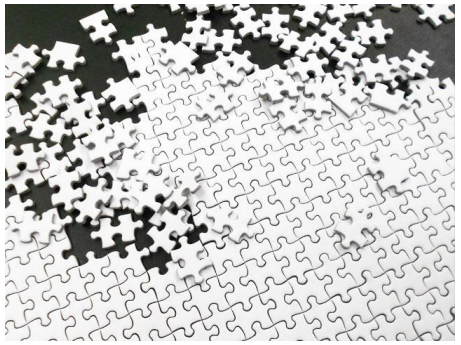


Simetrik Şifreleme: anahtarlar ortak

CENG 507 :
KRİPTOGRAFİK ALGORİTMALAR VE SİSTEMLER

CENG 434:
KRİPTOLOJİ

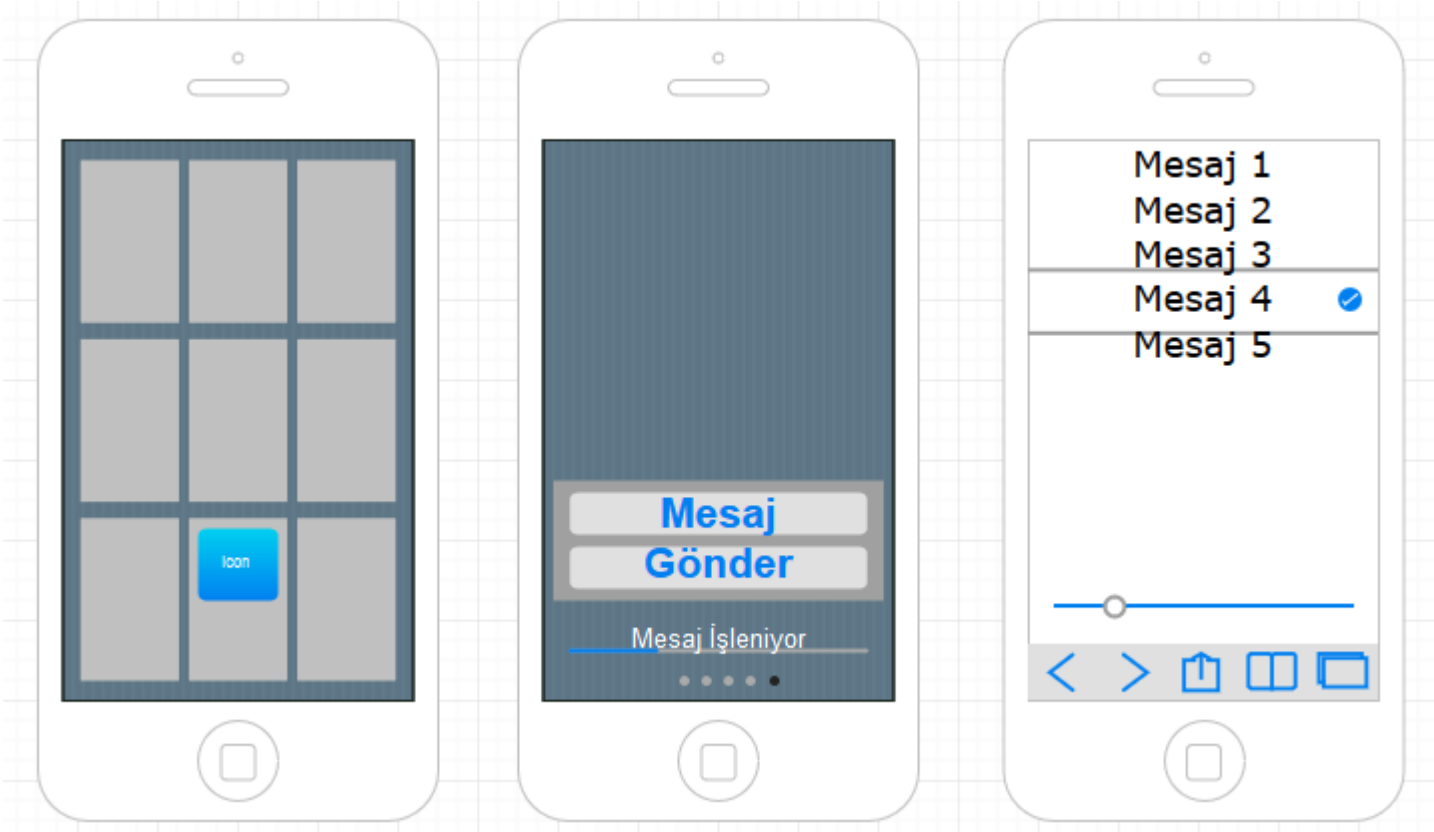
Araştırma ve Proje detayları için EDS'yi takip edin.

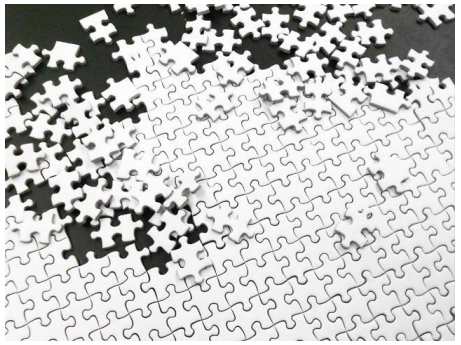


Proje

- Kullanıcı girişi
- Bir metin
- $Kullanıcı_A$
- $Kullanıcı_B$

- Tasarımı
- Uygulama
- Test senaryosu

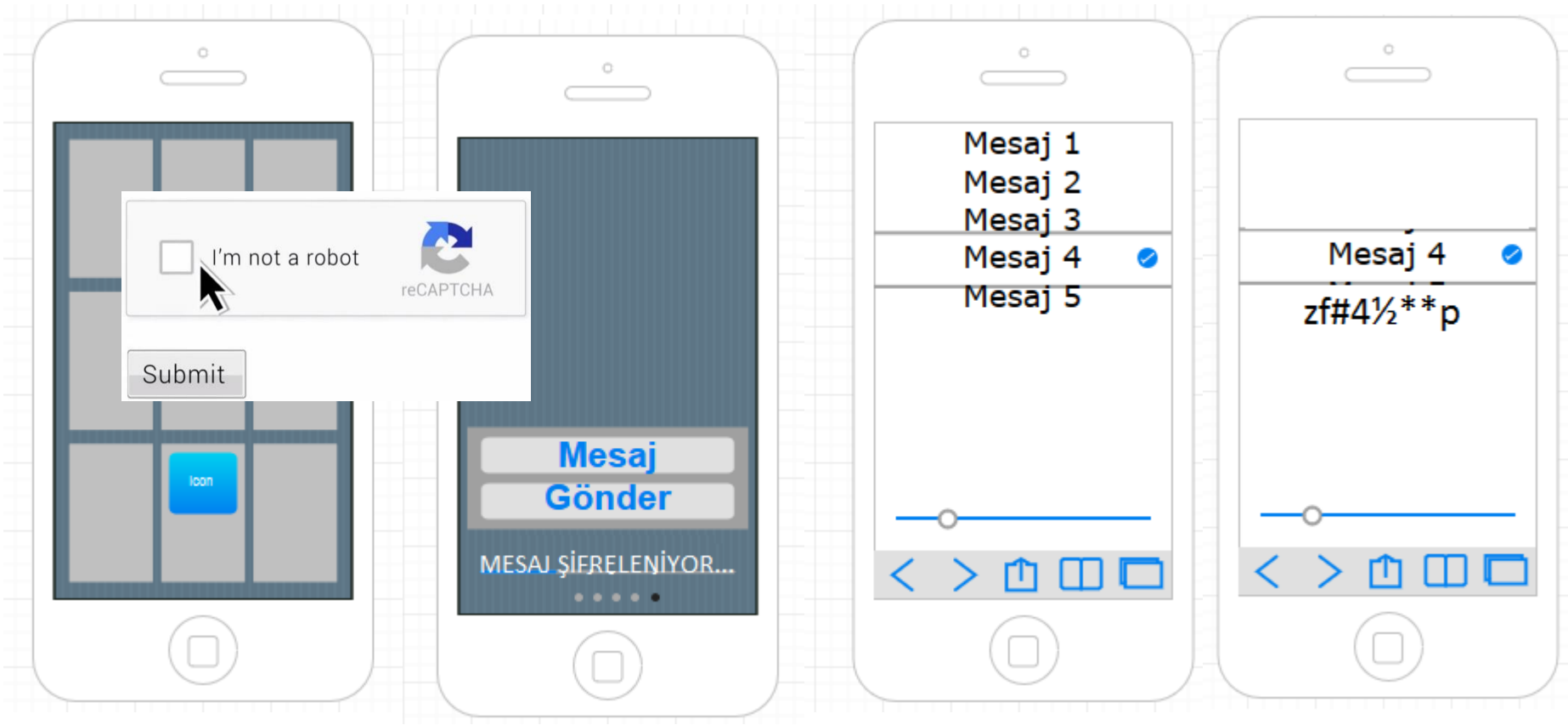


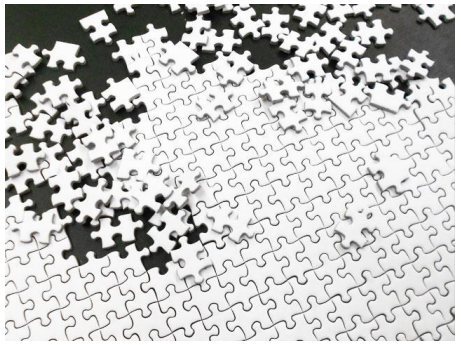


Proje

- Kullanıcı Girişi
- 3 Hatalı Giriş (bekle)
- 5 Hatalı Giriş (kilitle)
- Parola değiştirme
- Anahtar saklama
- Anahtar değişimi

- Tasarımı
- Uygulama
- Test senaryosu





Araştırma + Sunum

Bireysel

- Simetrik Şifreleme Algoritmaları
- Kimlik Doğrulama Mimarileri
- Rasgele Sayı Üreteçleri
- Özetleme Fonksiyonları
- Kriptanaliz