# Açık Anahtarlı Şifreleme ve RSA

Açık anahtarlı kriptografinin gelişimi, tüm kriptografi tarihindeki en büyük ve belki de tek gerçek devrimdir.

İlk başlangıcından modern zamanlara kadar, neredeyse tüm kriptografik sistemler, **yerine koyma ve permütasyon araçlarına dayanmaktadır.**

Bilgisayarların kullanılabilirliği ile daha da karmaşık sistemler tasarlandı, bunların en belirgini IBM'de Lucifer'in Veri Şifreleme Standardı (DES) ile sonuçlanan çabasıydı. DES önemli ilerlemeleri temsil etmesine rağmen, yine de yerine koyma ve permütasyona dayanıyordu.

Açık anahtarlı kriptografi, daha önce yapılan her şeyden radikal bir ayrılma sağlar.

Birincisi, **açık anahtar algoritmaları yerine koyma ve permütasyon yerine matematiksel fonksiyonlara dayalıdır**.

Daha da önemlisi, açık anahtarlı şifreleme, yalnızca bir anahtar kullanan simetrik şifrelemenin aksine, **iki ayrı anahtarın kullanımını içeren asimetriktir.**

İki anahtarın kullanımının gizlilik, anahtar dağıtımı ve kimlik doğrulama alanlarında önemli sonuçları vardır.

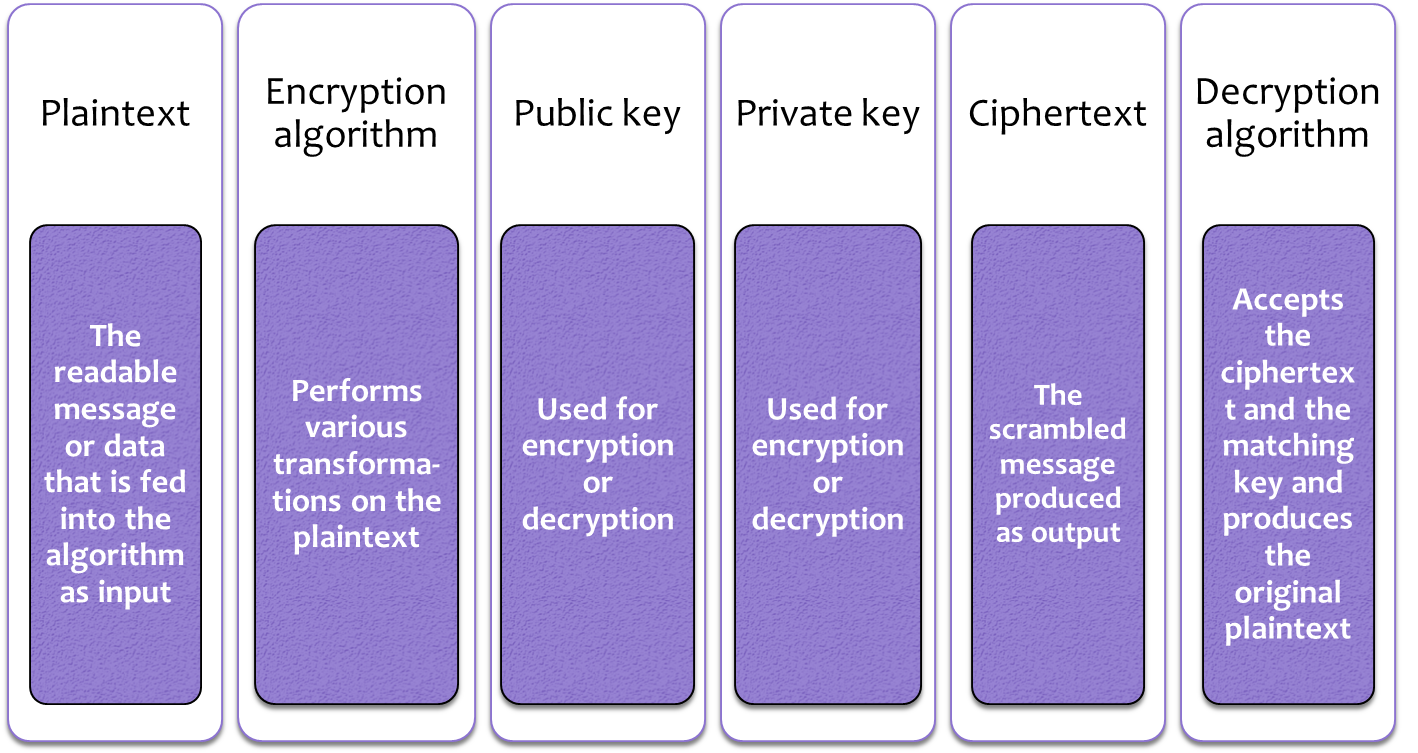
Bu bölüm, açık anahtar şifrelemesine genel bir bakış sağlar.

Ardından, açık anahtarlı şifreleme için uygulanabilir olduğu gösterilen en önemli şifreleme/şifre çözme algoritması olan RSA algoritmasını inceliyoruz.

# Açık Anahtarlı Şifreleme Sistemi

Asimetrik algoritmalar, şifreleme için bir anahtara ve şifre çözme için farklı ancak ilgili bir anahtara dayanır.

Bu algoritmalar aşağıdaki önemli özelliklere sahiptir.

Açık anahtarlı bir şifreleme şemasının altı bileşeni vardır:

**Temel adımlar aşağıdaki gibidir.**

1. Her kullanıcı, mesajların şifrelenmesi ve şifresinin çözülmesi için kullanılacak bir çift anahtar oluşturur.

2. Her kullanıcı, iki anahtardan birini genel bir kayda (public register) veya başka bir erişilebilir dosyaya yerleştirir. Bu genel anahtardır. Tamamlayıcı anahtar gizli tutulur. Şekil 9.1a'nın önerdiği gibi, her kullanıcı diğerlerinden alınan bir bir genel anahtar koleksiyonuna sahiptir.

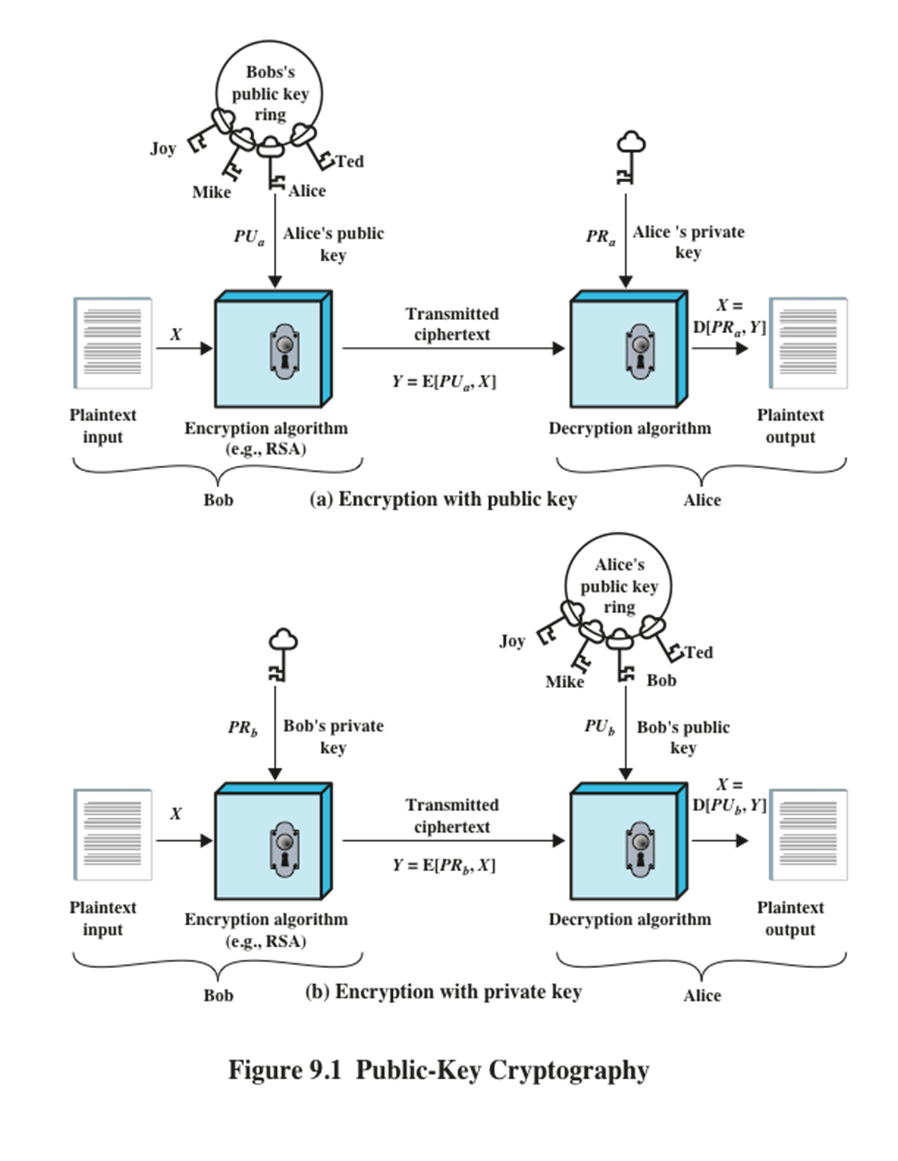
3. Bob, Alice'e gizli bir mesaj göndermek isterse, Bob mesajı Alice'in genel anahtarını kullanarak şifreler.

4. Alice mesajı aldığında, özel anahtarını kullanarak mesajın şifresini çözer. Başka hiçbir alıcı mesajın şifresini çözemez çünkü Alice'in özel anahtarını yalnızca Alice bilir.

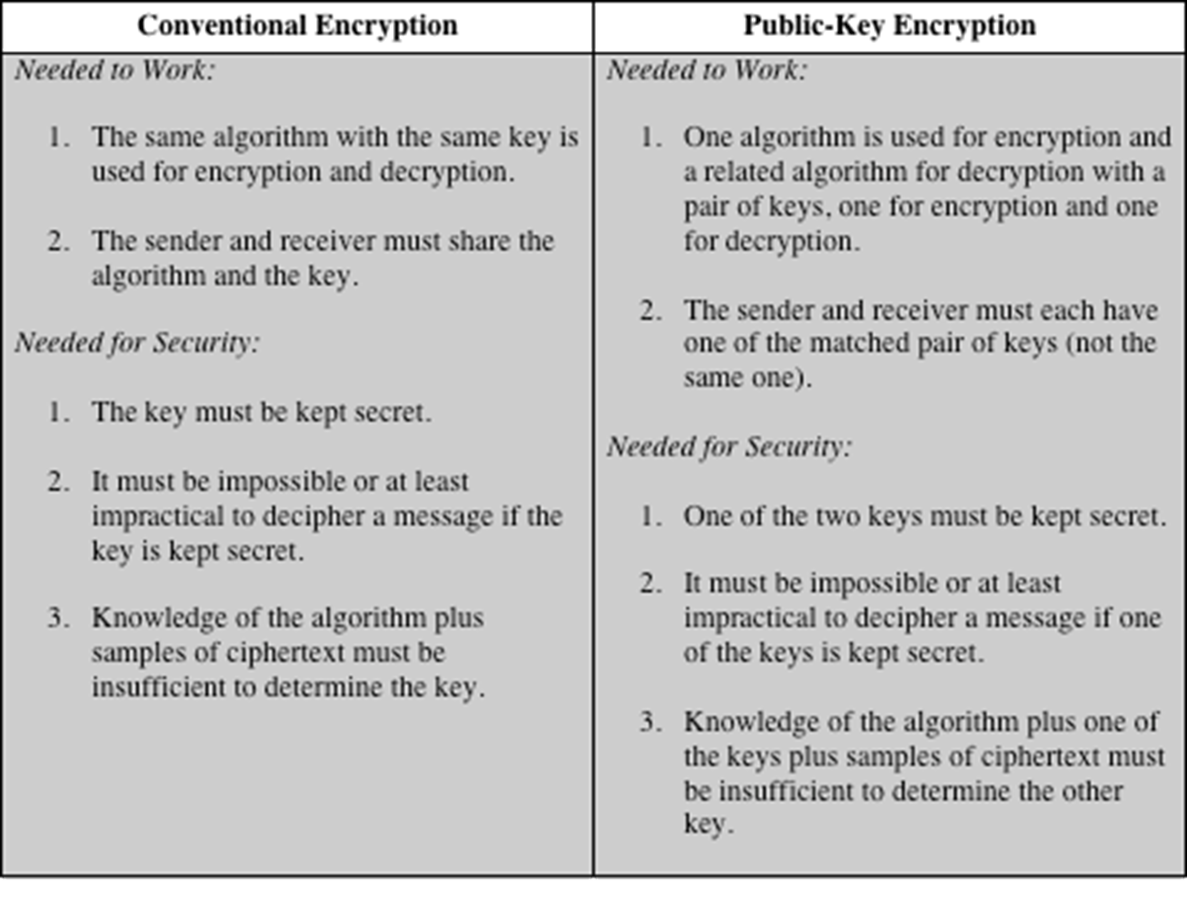
Bu yaklaşımla, tüm katılımcıların ortak anahtarlara erişimi vardır ve özel anahtarlar her katılımcı tarafından yerel olarak üretilir ve bu nedenle asla dağıtılmaları gerekmez.

Bir kullanıcının özel anahtarı korunduğu ve gizli kaldığı sürece, gelen iletişim güvenlidir.

Herhangi bir zamanda, bir sistem kendi özel anahtarını değiştirebilir ve eski ortak(genel, açık) anahtarının yerine tamamlayıcı ortak anahtarı yayınlayabilir.



# Geleneksel ve Açık Anahtarlı Şifreleme



# RSA algoritması

RSA (Rivest-Shamir-Adelman, 1978) algoritması asimetrik bir şifreleme algoritmasıdır.

**Bir şifreleme / şifre çözme anahtar çifti tasarlamak için,**

* iki büyük asal sayı, **p** ve **q,** seçilir ve
* (p-1) (q-1) 'e göre nispeten asal(relative prime) bir **e** tam sayısı seçilir (e ve (p-1)(q-1), 1'den başka ortak faktöre sahip değildir yani, gcd((p-1)(q-1)), e) =1).
* Son olarak, **e.d mod ((p-1) (q-1)) = 1** olacak şekilde bir **d** tamsayısı hesaplanır.

Bu demek oluyor ki





Yani **e** ve **d** çarpımsal ters modulo (p-1) (q-1)’dir . Modüler aritmetik kurallarına göre, bu yalnızca d (ve bu nedenle e) 'nin (p-1) (q-1)' e göre nispeten asal(relative prime) olması durumunda geçerlidir.

Aynı zamanda, gcd ((p-1) (q-1), d) = 1.

**Bir anahtar (e, N)** ve **diğer anahtar (d, N)’dir**, burada **N = p \* q** ve modul olarak adlandırılır.

**Örnek1:**

p = 7 ve q = 13 seçilebilir.

Sonra N = p \* q = 91 ve

(p-1) (q-1) = (7-1) (13-1) = (6) (12) = 72.

e’yi 1 < *e* < (p-1)(q-1) bu aralıkta seçin

e = 5 seçebiliriz (72 ile nispeten asaldır) ve d = 29 olur, çünkü e \* d = 145 ve . Burada d mod 72’de e’ nin tersidir.

Daha sonra, bir anahtar, K1 = (e, N) = (5, 91) ve diğer anahtar, K2 = (d, N) = (29, 91).

Şifrelenecek mesaj (düz metin), **her blok M, 0 ile (N-1) arasında bir tam sayı** olarak ele alınabilecek şekilde bloklara bölünür.

M'yi şifreli metin bloğu, C, olarak **şifrelemek(encrypt)** için.

 gerçekleştirilir.

C **şifresini çözmek(decrypt)** için

 gerçekleştirilir.

Protokol doğru çalışıyor çünkü



* Hem gönderenin hem de alıcının **N** değerini bilmesi gerekir. Gönderen **e** değerini, alıcı yalnızca **d** değerini bilir.
* RSA algoritması hakkında daha fazla ayrıntı, William Stallings’in Cryptography ve Network Security ders kitabında bulunabilir.

**Örneğe dönersek, M = 2 olduğunu farz edelim.**

Daha sonra M'yi şifrelemek için

 olarak hesaplıyoruz.

Böylece, C = 32. C'nin şifresini çözmek için

 hesaplıyoruz.

Bu da düz metin mesajı olan M’dir.

**Çalışma:** M = 5 için hesaplamaları gösterin.

**p ve q’yu elde etmek son derece zordur, bu nedenle, yalnızca gizli anahtar K2 (özel anahtar, private key, PR)’yi bilmek, bir mesajın doğru bir şekilde şifresini çözebilir.**

**RSA şeması(scheme) bileşenleri şunlardır:**

iki asal sayı *p*, *q*  (özel(private), seçilir(chosen))

*N* = *pq* (genel(public), heaplanır(calculated))

*e*, with gcd((p-1)(q-1), *e*) = 1; 1< *e* < (p-1)(q-1) (public, chosen)

*d* ≡ *e*-1 (mod (p-1)(q-1)) (private, calculated)

Euler totient fonksiyonunu düşünün

N=p\*q

(N)= (p)\* (q), (totient fonksiyonu)

(N) = (p - 1) (q - 1)

**şifreleme için genel anahtar** => yalnızca özel anahtarın sahibi şifreyi çözebilir => gizlilik

**şifreleme için özel anahtar** => genel anahtarı bilen herhangi bir kişi şifresini çözebilir => gizlilik yok, ancak **gönderenin kimlik doğrulaması** için kullanılabilir = dijital imza

**Örnek 2:** Farzadelim ki anahtarlar aşağıdaki gibi üretilir.

**1.** İki asal sayı seçin, *p* = 17 ve *q* = 11.

2. *N* = *pq* = 17 \* 11 = 187 hesaplayın.

3. *φ* (*N*) = (*p* - 1) (*q* - 1) = 16 \* 10 = 160 hesaplayın.

4. *e*'yi, *e*'nin *φ* (*N*) = 160'a göre en yüksek ve *φ* (*N*) 'den küçük olması için *e* = 7'yi olacak şekilde seçin.

5. d'yi ≡ 1 (mod 160) ve d <160 olacak şekilde belirleyin. Doğru değer d = 23'tür, çünkü 23 \* 7 = 161 = (1 x 160) + 1; d, genişletilmiş Euclid algoritması kullanılarak hesaplanabilir.

Sonuçta ortaya çıkan anahtarlar genel anahtar *PU* = {7, 187} ve özel anahtar *PR* = {23, 187} 'dir.

Bu anahtarları M = 88 düz metin girişi için kullanın.

Şifreleme için, *C* = 887 mod 187'yi hesaplamamız gerekiyor.

Modüler aritmetiğin özelliklerini kullanarak, bunu aşağıdaki gibi yapabiliriz.

887 mod 187 = [(884 mod 187) \* (882 mod 187)

\* (881 mod 187)] mod 187

881 mod 187 = 88

882 mod 187 = 7744 mod 187 = 77

884 mod 187 = 59,969,536 mod 187 = 132

887 mod 187 = (88 \* 77 \* 132) mod 187 = 894,432 mod 187 = 11

Şifre çözme için *M* = 1123 mod 187 hesaplnır:

1123 mod 187 = [(111 mod 187) \* (112 mod 187) \* (114 mod 187) \* (118 mod 187) \* (118 mod 187)] mod 187

111 mod 187 = 11

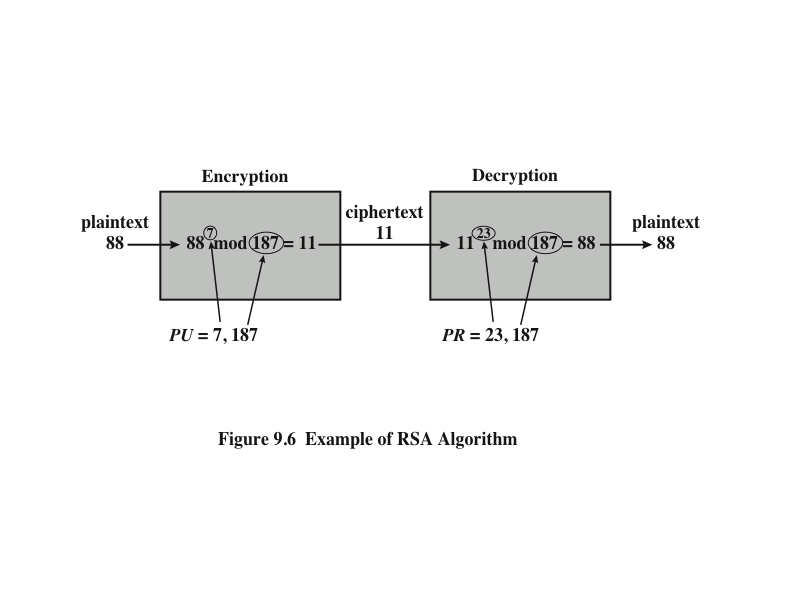
112 mod 187 = 121

114 mod 187 = 14,641 mod 187 = 55

118 mod 187 = 214,358,881 mod 187 = 33

1123 mod 187 = (11 \* 121 \* 55 \* 33 \* 33) mod 187

= 79,720,245 mod 187 = 88



**Örnek :** Örnek 1'de p=7, q=13, N=91 ve φ (N)=(p-1)(q-1)=72 olduğunu varsayalım. d=5 (ki bu 72'ye göre asaldır) ve e=29 seçebiliriz, çünkü e\*d=145 ve



gcd(d, φ (N))=1=> göreceli olarak asal ve ters var

gcd=Euclidean Algorithm(a,b)

A=a; B=b;

1: If B==0 then return gcd=A;

R= A mod B;

A=B;

B=R;

goto 1;

gcd(72,5)

A=72, B=5

R=A mod B = 72 mod 5

Q=floor(A/B)=floor(72/5)=floor(14.4)= 14

R=A-Q\*B=72-14\*5=2

A=B=5; B=R=2

R= 5 mod 2=1

A=2; B=1;

R=2 mod 1=0

A=1; B=0=>gcd=A=1

Extented Euclidean Algoritmasını kullanarak 5’in mod 72’de tersini hesaplayalım.

gcd(m,b), m=72, b=5

A=(1,0,72), B=(0,1,5)

Q=floor(72/5)=14

T=A-Q\*B=(1-14\*0, 0-14\*1, 72-14\*5)= (1, -14, 2)

A=(0,1,5), B=(1, -14, 2)

Q=floor(5/2)=2

T=A-Q\*B=(0-2\*1, 1-2\*(-14), 5-2\*2)=(-2, 29, 1)

A=(1,-14,2), B=(-2,29,1)

Inverse is B2=29

29\*5=145 mod 72 =1 valid inverse

Extended Euclidean Algorithm(m,b) to find b^(-1) mod m

A=(A1,A2.A3)=(1,0,m); B=(B1,B2,B3)=(0,1,b);

1: If B3==0 then return gcd=A3, inverse does not exist;

If B3==1 then return inverse is B2 it is b^(-1) mod m

Q=floor(A3/B3)

T= A-Q\*B=(A1-Q\*B1, A2-Q\*B2, A3-Q\*B3);

A=B;

B=T; !! order matters

Goto 1;

O halde, bir anahtar K1=(29,91) ve diğeri K2=(5,91) olur. Şifrelenecek mesaj, her blok M'nin 0 ile (N-1) arasında bir tamsayı olarak ele alınabileceği şekilde bloklara bölünür.

M'yi şifreli metin bloğu C'ye şifrelemek için,



C'nin şifresini çözmek için



Örneğe dönersek, M=2 olduğunu varsayalım.

Ardından, M'yi şifrelemek için hesaplarız



Böylece, C=32. C'nin şifresini çözmek için



düz metin mesajı olan M.

p ve q'nun elde edilmesi son derece zordur, bu nedenle, yalnızca K2 gizli anahtarını bilen alıcı bir mesajın şifresini doğru bir şekilde çözebilir..

2^29??

29=16+8+4+1=2^4+2^3+2^2+2^0=11101

2^29=2^(16+8+4+1)=2^16\*2^8\*2^4\*2=16\*74\*16\*2=256\*74\*2 mod 91 =74\*74\*2 mod 91 = 16\*2=32

2^16, 2^8, 2^4

2^2=4 mod 91=4

2^4=2^(2\*2)=(2^2)^2=4^2=16 mod 91 =16

2^8=(2^4)^2=16^2=256 mod 91 =(182+74) mod 91 =74

2^16=(2^8)^2=74\*74 mod 91= 5476 mod 91 =16

Decryption

32^5=32^(4+1)=32^4\*32 mod 91=74\*32 mod 91=(74\*2)\*16 mod 91 = 148\*16 mod 91=57\*16 mod 91 = 114\*8 mod 91 = 23\*8 mod 91 = 92\*2 mod 91 = 1\*2 mod 91 = 2

32^2=1024 mod 91 = 23

32^4=(32^2)^2=23^2=529 mod 91 = 74