Public Key Encryption and Digital Signatures

Public Key Encryption Overview

- Each party has a PAIR (K, K^{-1}) of keys:
  - K is the **public** key, and used for encryption
  - K^{-1} is the **private** key, and used for decryption
  - Satisfies $\mathbf{D}_{K^{-1}}[\mathbf{E}_K[M]] = M$
- Knowing the public-key K, it is computationally infeasible to compute the private key K^{-1}
  - How to check (K,K^{-1}) is a pair?
  - Offers only computational security. PK Encryption impossible when P=NP, as deriving K^{-1} from K is in NP.
- The public-key K may be made publicly available, e.g., in a publicly available directory
  - Many can encrypt, only one can decrypt
- Public-key systems aka *asymmetric* crypto systems

RSA Algorithm

- Invented in **1978** by Ron **R**ivest, Adi **S**hamir and Leonard **A**dleman
  - Published as R L Rivest, A Shamir, L Adleman, "*On Digital Signatures and Public Key Cryptosystems*", Communications of the ACM, vol 21 no 2, pp120-126, Feb 1978
- Security relies on the difficulty of factoring large composite numbers
- Essentially the same algorithm was discovered in 1973 by Clifford Cocks, who works for the British intelligence

RSA Public Key Crypto System

Key generation:

1. Select 2 large prime numbers of about the same size, p and q

   Typically each p, q has between 512 and 2048 bits

2. Compute $n = pq$, and $\Phi(n) = (q-1)(p-1)$
3. Select e, $1<e<\Phi(n)$, s.t. $\gcd(e, \Phi(n)) = 1$

   Typically e=3 or e=65537

4. Compute d, $1< d< \Phi(n)$ s.t. $(e.d) \equiv 1 \bmod \Phi(n)$

   Knowing $\Phi(n)$, d easy to compute.

Public key: (e, n)

Private key: (d,n)

RSA Description (cont.)

Encryption

Given a message M, $0 < M < n$ $\quad M \in Z_n - \{0\}$

use public key (e, n)

compute $C = M^e \bmod n$ $\quad C \in Z_n - \{0\}$

Decryption

Given a ciphertext C, use private key (d)

Compute $C^d \bmod n = (M^e \bmod n)^d \bmod n = M^{ed} \bmod n = M$

$$\text{Plaintext: } M \xrightarrow{C = M^e \bmod (n=pq)} \text{Ciphertext: } C$$

$$\text{Plaintext: } M \xleftarrow{C^d \bmod n} \text{Ciphertext: } C$$

From n, difficult to figure out p,q

From (n,e), difficult to figure d.

From (n,e) and C, difficult to figure out M s.t. $C = M^e$

RSA Example-1

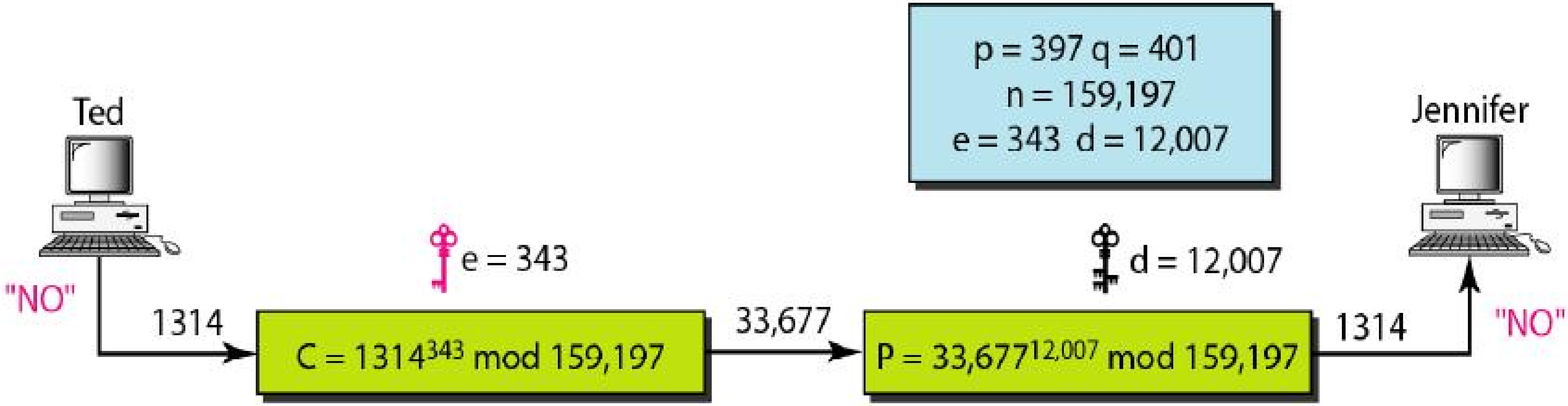
- Parameters:
  - p = 3, q = 5, q= pq = 15
  - Φ (n) = ?
- Let e = 3, what is d?
- Given M=2, what is C?
- How to decrypt?

RSA Example-2

- $p = 11$, $q = 7$, $n = 77$, $\Phi(n) = 60$
- $d = 13$, $e = 37$ ($ed = 481$; $ed \bmod 60 = 1$)
- Let $M = 15$. Then $C \equiv M^e \bmod n$
  - $C \equiv 15^{37} \pmod{77} = 71$
- $M \equiv C^d \bmod n$
  - $M \equiv 71^{13} \pmod{77} = 15$

RSA Example-3

- Jennifer creates a pair of keys for herself. She chooses p = 397 and q = 401. She calculates n = 159,197 and Φ = 396 · 400 = 158,400. She then chooses e = 343 and d = 12,007. Show Show Ted can send a message to Radha if he knows e and n.



Let us give a realistic example. We randomly chose an integer of 512 bits. The integer p is a 159-digit number.

p = 96130345313583504574191581280615427909309845594996215822583150879647940
455056470638491257160180347503120986660649242019180878066742109606335
4219926661209

The integer q is a 160-digit number.

q = 12060191957231446918276794204450896001555925054637033936061798321731482
148483764659215389453209175225273226830107120695604602513887145524969000
359660045617

We calculate n. It has 309 digits:

$$
\begin{aligned}
\mathbf{n} = {} & 1159350417396761496889250986461588752377145737545414477548552613761478\\
& 8540832635081727687881596832516846884930062548576411125016241455233918\\
& 2927162507656772727460097082714127730434960500556347274566628060099924\\
& 0371029914244722922157727985317270338393813346926841373276220009666766\\
& 71831831088373420823444370953
\end{aligned}
$$

We calculate Φ. It has 309 digits:

$$
\begin{aligned}
\phi = {} & 1159350417396761496889250986461588752377145737545414477548552613761478\\
& 8540832635081727687881596832516846884930062548576411125016241455233918\\
& 2927162507656751054233608492916752034482627988117554787657013923444405\\
& 7169895817281960982263610754672118646121713591073586406140088851702653\\
& 77277264467341066243857664128
\end{aligned}
$$

We choose e = 35,535. We then find d.

e = 35535

d = 580083028600377639360936612896779175946690620896509621804228661113805938528223587317062869100300217108590443384021707298690876006115306202524959884448047568240966247081485817130463240644077704833134010850947385295645071936774061197326557424237217617674620776371642076003370853332885321447088595513667029483 1

Alice wants to send the message "THIS IS A TEST" which can be changed to a numeric value by using the 00–26 encoding scheme (26 is the space character).

P = 1907081826081826002619041819

The ciphertext calculated by Alice is $C = P^e$, which is.

C = 47530912364622682720636555061054518094237179607049171652323924305445296061319932856661784341835911415119741125200568297979457173603610127821884789274156609048002350719071527718591497518846588863210114835410336165789846796838676373376577746562507928052114814184404814184430812773059004692874248559166462108656

Bob can recover the plaintext from the ciphertext by using $P = C^d$, which is

P = 1907081826081826002619041819

The recovered plaintext is THIS IS A TEST after decoding.

Applications of RSA

- RSA can be used to encrypt and decrypt actual messages; however, it is very slow if the message is long.
- RSA, therefore, is useful for short messages or a symmetric key to be used for a symmetric-key cryptosystem.