CS558: Introduction to Computer Security

Key Distribution

Key Distribution

- For symmetric encryption to work, the two parties must share a secrete key.
- Frequent key changes are usually desirable to limit the amount of data compromised if an attacker learns the key.
- Key distribution: refers to the means of delivering a key to two parties who wish to exchange data, without allowing others to see the key
- Often secure systems failure due to a break in the key distribution scheme

Key Distribution

- For two parties A and B, key distribution can be achieved in a number of ways:

Key Distribution

- For two parties A and B, key distribution can be achieved in a number of ways:

1. A can select key and physically deliver to B
2. A third party can select & physically deliver key to A & B
3. If A and B have communicated previously, A can transmit the new key to B, encrypted using the old key.

Key Distribution

- For two parties A and B, key distribution can be achieved in a number of ways:

1. A can select key and physically deliver to B
2. A third party can select & physically deliver key to A & B
3. If A and B have communicated previously, A can transmit the new key to B, encrypted using the old key.
   - If an attacker succeeds in getting one key, then all subsequent keys will be revealed

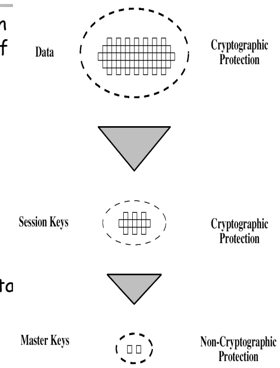
Key Distribution

- For two parties A and B, key distribution can be achieved in a number of ways:

4. If A & B have secure communications with a third party C, C can deliver a key on the encrypted links to A and B
   - ➢A key distribution center is responsible for distributing keys to pairs of users.
   - ➢Each user must share a unique key with the key distribution center for purpose of key distribution.

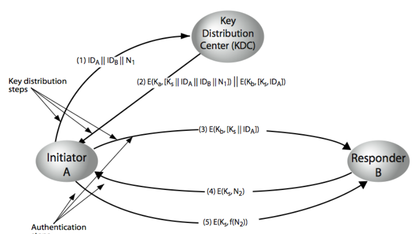
Key Hierarchy

- The use of a key distribution center is based on the use of a hierarchy of keys.
- Master key
  - ❖Used to encrypt session keys
  - ❖Shared by user & key distribution center
- Session key
  - ❖Temporary key
  - ❖Used for encryption of data between users



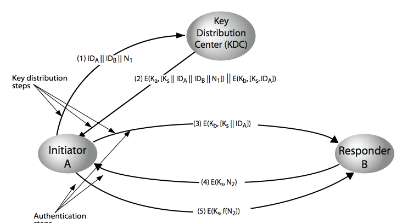
Key Distribution Scenario

- A wishes to establish a logical connection with B and requires a one-time session key to protect the data transmitted over the connection
- A shares the master key K_a with the KDC
- B shares the master key K_b with the KDC



Key Distribution Scenario

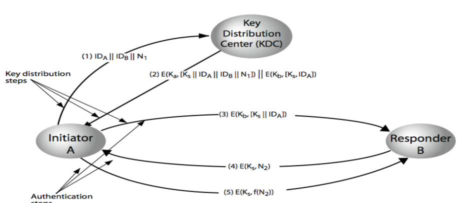
- Msg1: A issues a request to the KDC for a session key to protect a connection to B. The message includes the identity of A and B, and a unique identifier, N1 (nonce).
- Nonce: a random number that is used to demonstrate the freshness of a session – prevent replay attack



Key Distribution Scenario

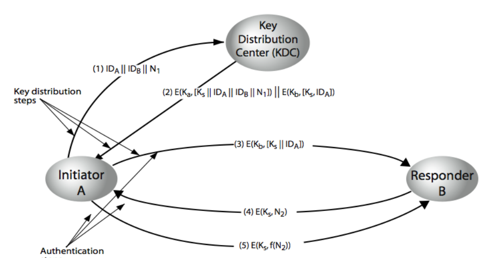
- Msg2: The KDC responds with a message encrypted using K_a

1. The one-time session key k_s
2. The original request message and the nonce
3. Two items for B, encrypted using K_b: the one-time session key k_s and an identity of A, ID_A.



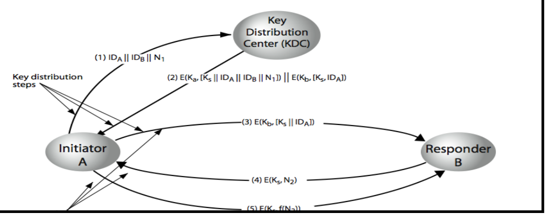
Key Distribution Scenario

- Msg3: A stores the session key for use in the upcoming session and forward to B the information that originated at the KDC for B, E(Kb,[Ks,IDA]).



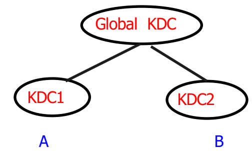
Key Distribution Scenario

- Now, a session key has been securely delivered to A and B.
- Msg4: B sends a nonce N_2 to A using the newly minted session key.
- Msg5: Also using K_s, A responds with $f(N_2)$, where f is a function that perform some transformation on N_2



Hierarchical Key Control

- It is not necessary to limit the key distribution function to a single KDC – for large networks, a hierarchy of KDCs can be established
  - E.g. local KDCs, each responsible for a small domain
  - If two entities are in different domains, then local KDCs can communicate through a global KDC.



Chapter 9
Public-Key Cryptography

Private-Key Cryptography

- Symmetric key cryptography uses one key, shared by both sender and receiver
- If this key is disclosed, communications are compromised
- Can we use symmetric key encryption to protect sender from receiver forging a message and claiming is sent by sender?

Private-Key Cryptography

- Symmetric key cryptography uses one key, shared by both sender and receiver
- If this key is disclosed, communications are compromised
- Can we use symmetric key encryption to protect sender from receiver forging a message and claiming is sent by sender?
  - John can deny sending the message. Because it is possible for Mary to forge a message, there is no way to prove that John did in fact send the message.
  - Mary may forge a different message and claim that it came from John

Public-Key Cryptography

- Public invention due to Whitfield Diffie & Martin Hellman at Stanford University in 1976.
- Public-key/two-key/asymmetric cryptography involves the use of two keys:
  - A public-key, which may be known by anybody, and can be used to encrypt messages, and verify signatures
  - A private-key, known only to the recipient, used to decrypt messages, and sign (create) signatures
- Is asymmetric because
  - Those who encrypt messages or verify signatures may not decrypt messages or create signatures

Public-key cryptography: Misconceptions

- Misconception 1: Public-key encryption is more secure from cryptanalysis than symmetric encryption

Public-key cryptography: Misconceptions

- Misconception 1: Public-key encryption is more secure from cryptanalysis than symmetric encryption
  - The security depends on the length of the key and the computational work involved in breaking a cipher.

Public-key cryptography: Misconceptions

- Misconception 1: Public-key encryption is more secure from cryptanalysis than symmetric encryption
  - The security depends on the length of the key and the computational work involved in breaking a cipher.
- Misconception 2: Public-key encryption is a general-purpose technique that has made symmetric encryption obsolete.

Public-key cryptography: Misconceptions

- Misconception 1: Public-key encryption is more secure from cryptanalysis than symmetric encryption
  - The security depends on the length of the key and the computational work involved in breaking a cipher.
- Misconception 2: Public-key encryption is a general-purpose technique that has made symmetric encryption obsolete.
  - Computation overhead of public-key encryption

Why Public-Key Cryptography?

- Developed to address two key issues:
  - Key distribution – how to have secure communications in general without having to trust a KDC
  - Digital signatures – how to verify a message comes intact from the claimed sender
- Public invention due to Whitfield Diffie & Martin Hellman at Stanford University in 1976.

Requirements for Public-Key Cryptography

- It is computationally easy for a party B to generate a pair: public key PU_b, private key PR_b
- The two keys can be applied in either order.

  $M = D(PU_b, E(PR_b, M)) = D(PR_b, E(PU_b, M))$
- It is computationally easy for sender A, knowing the public key and the message to be encrypted, M, to generate the corresponding ciphertext.

  $C = E(PU_b, M)$
- It is computationally easy for the receiver B to decrypt the resulting ciphertext using the private key to recover the original message.

  $M = D(PR_b, C) = D(PR_b, E(PU_b, M))$

Requirements for Public-Key Cryptography (Cont.)

- It is computationally infeasible for an adversary, knowing the public key PU_b, to determine the private key PR_b.
- It is computationally infeasible for an adversary, knowing the public key PU_b and the ciphertext C encrypted using PU_b, to recover the original message M.
- These are formidable requirements – only a few algorithms (e.g. RSA) have received widespread acceptance.

Public-Key Cryptosystems: Secrecy

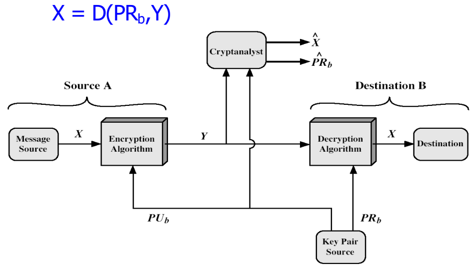
- A produces plaintext X = [X1,X2,…,Xn]
- The message is intended for destination B.
- A has two keys: a public key PU_a, and a private key PR_a.
- B has two keys: a public key PU_b, and a private key PR_b.

Public-Key Cryptosystems: Secrecy

- A forms the ciphertext Y = [Y1,Y2,…,Yn]:

  $Y = E(PU_b, X)$
- The receiver is able to invert the transformation

  $X = D(PR_b, Y)$



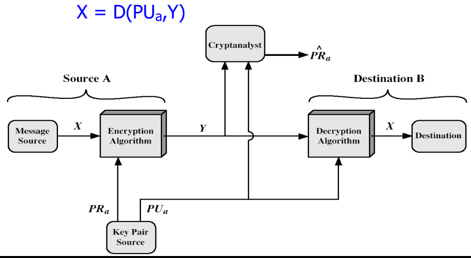
Public-Key Cryptosystems: Digital Signature

Public-Key Cryptosystems: Digital Signature

- A prepares a message to B and encrypts it using A's private key before transmitting it.

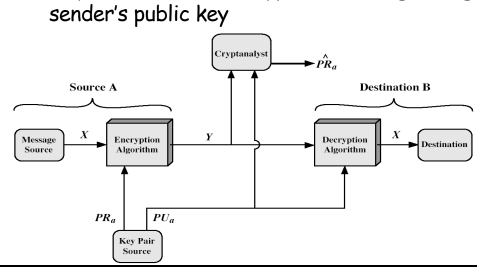
  $Y = E(PR_a, X)$
- B decrypts the message using A's public key

  $X = D(PU_a, Y)$



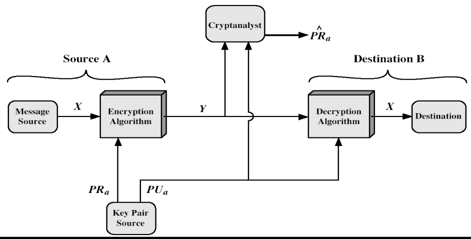
Public-Key Cryptosystems: Digital Signature

- Does not provide confidentiality.
  - The message being sent is safe from alteration but not from eavesdropping.
  - Any observer can decrypt the message using the sender's public key



Public-Key Cryptosystems: Digital Signature

- Because the message was encrypted using A's private key, only A could have prepared the message
  - Serves as digital signature.
  - It is impossible to alter the message without knowing A's private key ➔ data integrity

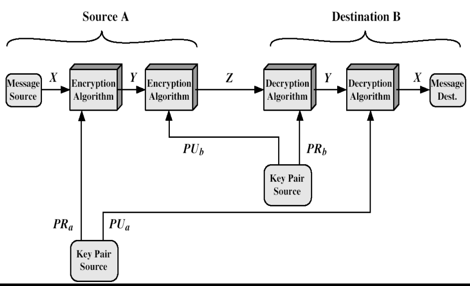


Public-Key Cryptosystems: Digital Signature and Secrecy

Public-Key Cryptosystems: Digital Signature and Secrecy

- Double use of the public-key scheme:

$$Z = E(PU_b, E(PR_a, X))$$
$$X = D(PU_a, D(PR_b, Z))$$



Conventional vs. Public-Key Encryption

Conventional Encryption	Public-Key Encryption
Needed to Work:	*Needed to Work:*
1. The same algorithm with the same key is used for encryption and decryption.	1. One algorithm is used for encryption and decryption with a pair of keys, one for encryption and one for decryption.
2. The sender and receiver must share the algorithm and the key.	2. The sender and receiver must each have one of the matched pair of keys (not the same one).
Needed for Security:	*Needed for Security:*
1. The key must be kept secret.	1. One of the two keys must be kept secret.
2. It must be impossible or at least impractical to decipher a message if no other information is available.	2. It must be impossible or at least impractical to decipher a message if no other information is available.
3. Knowledge of the algorithm plus samples of ciphertext must be insufficient to determine the key.	3. Knowledge of the algorithm plus one of the keys plus samples of ciphertext must be insufficient to determine the other key.

Chapter 8
Introduction to Number Theory

Prime Numbers

- Prime numbers play a critical role both in number theory and in cryptography

Prime Numbers

- Prime numbers play a critical role both in number theory and in cryptography
- An integer $p > 1$ is a prime number if and only if its only divisors are ± 1 and $\pm p$
  - Eg. 2,3,5,7 are prime, 4,6,8,9,10 are not
- Any integer $a > 1$ can be factored in a unique way as $a = p_1^{a1} p_2^{a2} \ldots p_t^{at}$
  - $p_1 < p_2 < \ldots < p_t$ are prime numbers and a_i are positive integers.
  - eg. 91= ; 3600=

Prime Numbers

- Prime numbers play a critical role both in number theory and in cryptography
- An integer $p > 1$ is a prime number if and only if its only divisors are ± 1 and $\pm p$
  - Eg. 2,3,5,7 are prime, 4,6,8,9,10 are not
- Any integer $a > 1$ can be factored in a unique way as $a = p_1^{a1} p_2^{a2} \ldots p_t^{at}$
  - $p_1 < p_2 < \ldots < p_t$ are prime numbers and a_i are positive integers.

  - eg. 91=7 * 13 ; 3600=

Prime Numbers

- Prime numbers play a critical role both in number theory and in cryptography
- An integer $p > 1$ is a prime number if and only if its only divisors are ± 1 and $\pm p$
  - Eg. 2,3,5,7 are prime, 4,6,8,9,10 are not
- Any integer $a > 1$ can be factored in a unique way as $a = p_1^{a1} p_2^{a2} \ldots p_t^{at}$
  - $p_1 < p_2 < \ldots < p_t$ are prime numbers and a_i are positive integers.

  - eg. $91=7 * 13$; $3600=2^4 * 3^2 * 5^2$

Greatest Common Divisor (gcd)

- The greatest common divisor of integers a and b, expressed gcd(a,b):

Greatest Common Divisor (gcd)

- The greatest common divisor of integers a and b, expressed gcd(a,b):

  - The largest positive integer that divides both numbers without remainder

Greatest Common Divisor (gcd)

- The greatest common divisor of integers a and b, expressed gcd(a,b):
  - The largest positive integer that divides both numbers without remainder

- Can determine the greatest common divisor by comparing their prime factorizations and using least powers
  - eg. $300=2^1 x 3^1 x 5^2$, $18=2^1 x 3^2$ hence

gcd(18,300)=

Greatest Common Divisor (gcd)

- The greatest common divisor of integers a and b, expressed gcd(a,b):
  - The largest positive integer that divides both numbers without remainder
- Can determine the greatest common divisor by comparing their prime factorizations and using least powers
  - eg. $300=2^1 \times 3^1 \times 5^2$, $18=2^1 \times 3^2$ hence

  $gcd(18,300)= 2^1 \times 3^1 \times 5^0=6$

Greatest Common Divisor (gcd)

- gcd(x,y) = x if y ==0

  = gcd(y, (x mod y)) if x>=y and y>0

e.g.

gcd(300, 18) = gcd(18, (300 mod 18))
= gcd(18, 12)
= gcd(12, (18 mod 12))
= gcd(12, 6)
= gcd(6, 0) = 6

Fermat's Theorem

- Fermat's Theorem: If p is a prime number and a < p is a positive integer not divisible by p, then

  $a^{p-1} \bmod p = 1$.
  - E.g. p =3, a = 2 → $a^{p-1} \bmod p = 4 \bmod 3 = 1$.
- Also $a^p \bmod p = a$
- Useful in public key and primality testing

Euler Totient Function ø(n)

- Euler Totient Function ø(n): the number of positive integers less than n and relatively prime to n.
  - m is a relatively prime to n if gcd(m,n)=1
  - ø(37)

Euler Totient Function ø(n)

- Euler Totient Function ø(n): the number of positive integers less than n and relatively prime to n.
  - m is a relatively prime to n if gcd(m,n)=1
  - ø(37) = 36: all integers from 1 through 36 are relatively prime to 37.
  - For a prime number p, ø(p)

Euler Totient Function ø(n)

- Euler Totient Function ø(n): the number of positive integers less than n and relatively prime to n.
  - m is a relatively prime to n if gcd(m,n)=1
  - ø(37) = 36: all integers from 1 through 36 are relatively prime to 37.
  - For a prime number p, ø(p) = p-1
  - ø(35) =

Euler Totient Function ø(n)

- Euler Totient Function $ø(n)$: the number of positive integers less than n and relatively prime to n.
  - m is a relatively prime to n if $\gcd(m,n)=1$
  - $ø(37) = 36$: all integers from 1 through 36 are relatively prime to 37.
  - For a prime number p, $ø(p) = p-1$
  - $ø(35) = 24$:
    - 1, 2, 3, 4, 6, 8, 9, 11, 12, 13, 16, 17, 18, 19, 22, 23, 24, 26, 27, 29, 31, 32, 33, 34.

Euler Totient Function ø(n)

- Two prime numbers p and q with $p \neq q$, then
  $ø(pq) = ø(p)*ø(q) = (p-1)*(q-1)$
  - The set of integers less than pq is $\{1, 2, \ldots, pq-1\}$

Euler Totient Function ø(n)

- Two prime numbers p and q with $p \neq q$, then
  $ø(pq) = ø(p)*ø(q) = (p-1)*(q-1)$
  - The set of integers less than pq is $\{1, 2, \ldots, pq-1\}$
  - The integers in this set that are not relatively prime to n: $\{p, 2p, \ldots, (q-1)p\}$ and $\{q, 2q, \ldots, (p-1)q\}$

Euler Totient Function ø(n)

- Two prime numbers p and q with $p \neq q$, then
  $ø(pq) = ø(p)*ø(q) = (p-1)*(q-1)$
  - The set of integers less than pq is $\{1, 2, \ldots, pq-1\}$
  - The integers in this set that are not relatively prime to p*q: $\{p, 2p, \ldots, (q-1)p\}$ and $\{q, 2q, \ldots, (p-1)q\}$

$$\begin{aligned} ø(pq) &= (pq - 1) - [(q-1) + (p-1)] \\ &= pq - p - q + 1 \\ &= (p-1) * (q-1) \\ &= ø(p)*ø(q) \end{aligned}$$

- E.g. $ø(21) = (3-1)*(7-1) = 2*6 = 12$

Euler's Theorem

- Euler's Theorem: for every a and n that are relatively prime, $a^{ø(n)} \bmod n = 1$
  - a=3;n=10;

  $ø(10)=4$; $3^4 \bmod 10 = 81 \bmod 10 = 1$
  - a=2;n=11;

  $ø(11)=10$; $2^{10} \bmod 11 = 1024 \bmod 11 = 1$

Primality Testing

- For many cryptographic algorithms, it is necessary to select one or more very large prime numbers at random

Primality Testing

- For many cryptographic algorithms, it is necessary to select one or more very large prime numbers at random
- Naïve algorithm: divide by all numbers in turn less than the square root of the number
  - Only works for small numbers

Miller Rabin Algorithm

- Background
  - $n-1 = 2^k q$ with $n > 3$, n odd, $k > 0$, q odd
    - Divide $(n-1)$ by 2 until the result is an odd number.
- Property
  - Let $n > 2$ be a prime number, a be an integer $1 < a < n-1$, and $n-1 = 2^k q$. Then one of the following two conditions is true: 1) $a^q \bmod n = 1$ or 2) there exists $1 \le j \le k$ such that $a^{(2^{j-1} q)} \bmod n = n - 1$.

Miller Rabin Algorithm

- Background
  - $n-1 = 2^k q$ with $n > 3$, n odd, $k > 0$, q odd
    - Divide $(n-1)$ by 2 until the result is an odd number.
- Property
  - Let $n > 2$ be a prime number, a be an integer $1 < a < n-1$, and $n-1 = 2^k q$. Then one of the following two conditions is true: 1) $a^q \bmod n = 1$ or 2) there exists $1 \le j \le k$ such that $a^{(2^{j-1} q)} \bmod n = n - 1$.

However, if the above condition is met, n may not be a prime.

E.g. $n=2047=23*89$, then $n-1 = 2*1023$.

$2^{1023} \bmod 2047 = 1$, but 2047 is not a prime

Miller Rabin Algorithm

- Algorithm: check if n is a prime

1. Find integers $k > 0$, q odd, so that $(n-1)=2^k q$
2. Select a random integer $1<a<n-1$
3. if $a^q \bmod n = 1$ then return ("maybe prime");
4. for j = 1 to k do
   if $a^{2^{j-1} q} \bmod n = n-1$ then return("maybe prime")

//n is definitely not prime

5. return ("not prime")

Probabilistic Considerations

- It was shown that given an odd number n that is not prime and a randomly chosen integer $1 < a < n-1$, the probability that the algorithm fails to detect that n is not a prime is $< 1/4$

Probabilistic Considerations

- It was shown that given an odd number n that is not prime and a randomly chosen integer $1 < a < n-1$, the probability that the algorithm fails to detect that n is not a prime is $< 1/4$
- Hence if repeat test with different a, then chance n is prime after t tests is:
  - Pr(n maybe a prime after t tests) = $(1/4)^t$
  - eg. for $t=10$ this probability is $< 10^{-6}$

Section 9.2 The RSA Algorithm

RSA

- By Rivest, Shamir & Adleman of MIT in 1977
- Best known & widely used public-key scheme
- The RSA scheme is a block cipher
  - A typical size is 1024 bits.

Algorithm

- Each block has a value less than some number n
- Encryption and decryption are of the following form for some plaintext block M and ciphertext block C.

  $C = M^e \bmod n$

  $M = C^d \bmod n$

  Property of modular arithmetic

  $[(a1 \bmod n) * ......* (am \bmod n)] \bmod n$
  $= (a1*...*am) \bmod n$

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

Determining e and d

- Find values of e, d, n s.t. $M^{ed} \bmod n = M$ for all $M < n$.
- Theorem:

  If $e*d=1+k.ø(n)$ (or $e*d \bmod ø(n) = 1$) where $\gcd(e,ø(n)) = 1$, then $M^{ed} \bmod n = M$.

  The proof is given at the end of the slides

RSA Algorithm

Theorem:

If $e*d=1+k.ø(n)$ (or $e*d \bmod ø(n) = 1$) where $\gcd(e,ø(n)) = 1$, then $M^{ed} \bmod n = M$.

- Find values of e, d, n such that $M^{ed} \bmod n = M$ for all $M < n$
  - Selecting two large primes p and q
  - Computing $n=p*q$
  - $ø(n)=(p-1)(q-1)$
  - Selecting at random the encryption key e where $1<e<ø(n)$, $\gcd(e,ø(n)) = 1$
  - Solve following equation to find decryption key d

    $e*d \bmod ø(n) = 1$ and $0 \le d \le n$

RSA Use

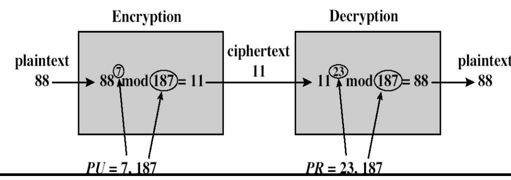
- To encrypt a message M the sender:
  - Obtains public key of recipient $PU=\{e,n\}$
  - Computes: $C = M^e \bmod n$, where $0 \le M<n$
- To decrypt the ciphertext C the owner:
  - Uses their private key $PR=\{d,n\}$
  - Computes: $M = C^d \bmod n$
- Can also use the private key to encrypt the message and use the public key to decrypt the message

RSA Example - Key Setup

1. Select primes: $p=17$ & $q=11$
1. Compute $n = p*q = 17 \times 11 = 187$
1. Compute $ø(n)=(p-1)(q-1)=16 \times 10=160$
1. Select e: $gcd(e,160)=1$; choose $e=7$
1. Determine d: $d*e \bmod 160 = 1$ and $d < 160$. Value is $d=23$ since $23*7=161= 160+1$
1. Publish public key $PU=\{7,187\}$
1. Keep private key $PR=\{23,187\}$

RSA Example - En/Decryption

- Sample RSA encryption/decryption is:
- Given message $M = 88$ (nb. $88<187$)
- Encryption:
  $C = 88^7 \bmod 187 = 11$
- Decryption:
  $M = 11^{23} \bmod 187 = 88$



RSA Requirements

- Encryption and decryption are of the following form for some plaintext block M and ciphertext block C.
  $C = M^e \bmod n$
  $M = C^d \bmod n = M^{ed} \bmod n$
- The following requirements must be met:
  - Requirement 1: It is possible to find values of e, d, n such that $M^{ed} \bmod n = M$ for all $M < n$
  - Requirement 2: It is relatively easy to calculate $M^e \bmod n$ and $C^d \bmod n$ for all values of $M < n$
  - Requirement 3: It is infeasible to determine d given e and n

RSA Security

- Possible approaches to attacking RSA are:
  - Brute force attacks
  - Mathematical attacks:
    - Factoring n into its two prime factors
    - Determine $ø(n)$ directly without determining p and q.
    - Determine d directly.