Chapter 6 The RSA Algorithm

Introduction

- The development of public-key cryptography is the greatest and perhaps the only true revolution in the entire history of cryptography.

Introduction

- The development of public-key cryptography is the greatest and perhaps the only true revolution in the entire history of cryptography.
- Previously, virtually all cryptosystems have been based on the elementary tools of substitution and permutation.

Introduction

- The development of public-key cryptography is the greatest and perhaps the only true revolution in the entire history of cryptography.
- Previously, virtually all cryptosystems have been based on the elementary tools of substitution and permutation.
- A major advance in symmetric cryptography occurred with the development of the rotor encryption/decryption machine.
- With the availability of computers, even more complex systems were devised (most prominently, LUCIFER which led to DES)

Introduction

- The development of public-key cryptography is the greatest and perhaps the only true revolution in the entire history of cryptography.
- Previously, virtually all cryptosystems have been based on the elementary tools of substitution and permutation.
- A major advance in symmetric cryptography occurred with the development of the rotor encryption/decryption machine.
- With the availability of computers, even more complex systems were devised (most prominently, LUCIFER which led to DES)
- Both the rotor machines and DES, although representing significant advances, still relied on the tools of substitution and permutation.

Public-Key Encryption

- Public-key algorithms are based on mathematical functions rather than substitution and permutations.

Public-Key Encryption

- Public-key algorithms are based on mathematical functions rather than substitution and permutations.
- Public-key cryptography is asymmetric, involving the use of two separate keys.

Public-Key Encryption

- Public-key algorithms are based on mathematical functions rather than substitution and permutations.
- Public-key cryptography is asymmetric, involving the use of two separate keys.
- The use of two keys has profound consequences in the area of confidentiality, key distribution and authentication.

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
    - depends on length of the key

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
  - depends on length of the key
  - depends on how much computational work is required

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
    - depends on length of the key
    - depends on how much computational work is required
- Public-key encryption is a general purpose technique that has made symmetric encryption obsolete.

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
  - depends on length of the key
  - depends on how much computational work is required
- Public-key encryption is a general purpose technique that has made symmetric encryption obsolete.
  - because of the computational overhead of current public-key encryption schemes, there seems to be no foreseeable likelihood that symmetric encryption will be abandoned

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
  - depends on length of the key
  - depends on how much computational work is required
- Public-key encryption is a general purpose technique that has made symmetric encryption obsolete.
  - because of the computational overhead of current public-key encryption schemes, there seems to be no foreseeable likelihood that symmetric encryption will be abandoned
  - ‘the restrictions on public-key cryptography to key management and signature applications is almost universally accepted’

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
  - depends on length of the key
  - depends on how much computational work is required
- Public-key encryption is a general purpose technique that has made symmetric encryption obsolete.
  - because of the computational overhead of current public-key encryption schemes, there seems to be no foreseeable likelihood that symmetric encryption will be abandoned
  - 'the restrictions on public-key cryptography to key management and signature applications is almost universally accepted'
- Key distribution is trivial when using public-key encryption, compared to the rather cumbersome hand-shaking involved with key distribution centers for symmetric encryption.

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
  - depends on length of the key
  - depends on how much computational work is required
- Public-key encryption is a general purpose technique that has made symmetric encryption obsolete.
  - because of the computational overhead of current public-key encryption schemes, there seems to be no foreseeable likelihood that symmetric encryption will be abandoned
  - 'the restrictions on public-key cryptography to key management and signature applications is almost universally accepted'
- Key distribution is trivial when using public-key encryption, compared to the rather cumbersome hand-shaking involved with key distribution centers for symmetric encryption.
  - some form of protocol is needed, generally involving a central agent

Misconceptions

- Public-key encryption is more secure from cryptoanalysis than is symmetric encryption.
  - depends on length of the key
  - depends on how much computational work is required
- Public-key encryption is a general purpose technique that has made symmetric encryption obsolete.
  - because of the computational overhead of current public-key encryption schemes, there seems to be no foreseeable likelihood that symmetric encryption will be abandoned
  - ‘the restrictions on public-key cryptography to key management and signature applications is almost universally accepted’
- Key distribution is trivial when using public-key encryption, compared to the rather cumbersome hand-shaking involved with key distribution centers for symmetric encryption.
  - some form of protocol is needed, generally involving a central agent
  - the procedures involved are not simpler nor any more efficient than those required for symmetric encryption

Key Terms

Definition

Asymmetric Keys

Two related keys, a public key and a private key, that are used to perform complementary operations, such as encryption and decryption or signature generation and signature verification.

Key Terms

Definition

Asymmetric Keys

Two related keys, a public key and a private key, that are used to perform complementary operations, such as encryption and decryption or signature generation and signature verification.

Definition

Public Key Certificate

A digital document issued and digitally signed by the private key of a certification authority that binds the name of the subscriber to a public key. The certificate indicates that the subscriber identified in the certificate has sole control and access to the corresponding private key.

Key Terms

Definition

Public Key (Asymmetric) Cryptographic Algorithm

A cryptographic algorithm that uses two related keys, a public and a private key. The two keys have the property that deriving the private key from the public key is computationally infeasible.

Key Terms

Definition

Public Key (Asymmetric) Cryptographic Algorithm

A cryptographic algorithm that uses two related keys, a public and a private key. The two keys have the property that deriving the private key from the public key is computationally infeasible.

Definition

Public Key Infrastructure (PKI)

A set of policies, processes, server platforms, software and work stations used the purpose of administering certificates and public-private key pairs, including the ability to issue, maintain and revoke public key certificates.

Principles of Public-Key Cryptosystems

The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems with symmetric encryption.

Principles of Public-Key Cryptosystems

The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems with symmetric encryption.

The first problem is key distribution. Key distribution under symmetric encryption requires either

(1) that communicants already share a key, which has somehow been distributed to them; or

Principles of Public-Key Cryptosystems

The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems with symmetric encryption.

The first problem is key distribution. Key distribution under symmetric encryption requires either

(1) that communicants already share a key, which has somehow been distributed to them; or

(2) the use of a key distribution center

Principles of Public-Key Cryptosystems

The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems with symmetric encryption.

The first problem is key distribution. Key distribution under symmetric encryption requires either

(1) that communicants already share a key, which has somehow been distributed to them; or

(2) the use of a key distribution center

Whitfield Diffie (who discovered public-key encryption along with Martin Hellman while both were at Stanford) reasoned that this second requirement negated the very essence of cryptography; the ability to maintain total secrecy over your own communications.

The second problem was that of digital signatures.

Principles of Public-Key Cryptosystems

The second problem was that of digital signatures.

- Wanted widespread use, not just military

Principles of Public-Key Cryptosystems

The second problem was that of digital signatures.

- Wanted widespread use, not just military
- Needed electronic signatures to be equivalent of those on documents

Principles of Public-Key Cryptosystems

The second problem was that of digital signatures.

- Wanted widespread use, not just military
- Needed electronic signatures to be equivalent of those on documents
- Needed to be sure signature was sent by a particular person and satisfactory to both parties

Diffie and Hellman

Diffie and Hellman achieve an astounding breakthrough in 1976 by coming up with a method that addressed both problems and was radically different from all previous approaches to cryptography.

Diffie and Hellman

Diffie and Hellman achieve an astounding breakthrough in 1976 by coming up with a method that addressed both problems and was radically different from all previous approaches to cryptography.

Asymmetric algorithms, such as the RSA, rely on two different but related keys (one for encryption and one for decryption) and these algorithms have a very important characteristic: it is computationally infeasible to determine the decryption key given only the knowledge of the cryptographical algorithm and the encryption key.

Diffie and Hellman

Diffie and Hellman achieve an astounding breakthrough in 1976 by coming up with a method that addressed both problems and was radically different from all previous approaches to cryptography.

Asymmetric algorithms, such as the RSA, rely on two different but related keys (one for encryption and one for decryption) and these algorithms have a very important characteristic: it is computationally infeasible to determine the decryption key given only the knowledge of the cryptographical algorithm and the encryption key.

Another property is that either key can be used for encryption and the other for decryption.

Ingredients

- Plaintext

Ingredients

- Plaintext
- Encryption algorithm

Ingredients

- Plaintext
- Encryption algorithm
- Public and private keys

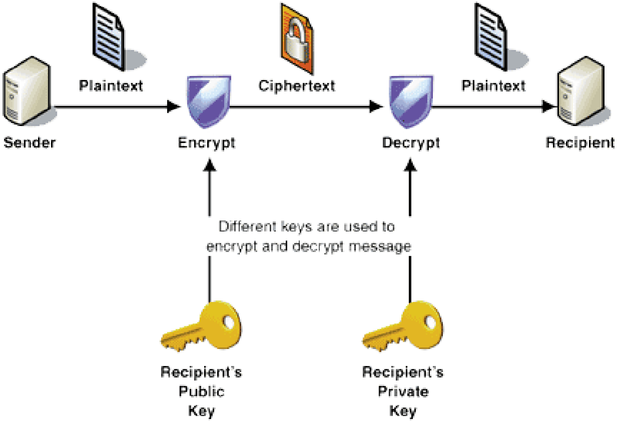
Ingredients

- Plaintext
- Encryption algorithm
- Public and private keys
- Ciphertext

Ingredients

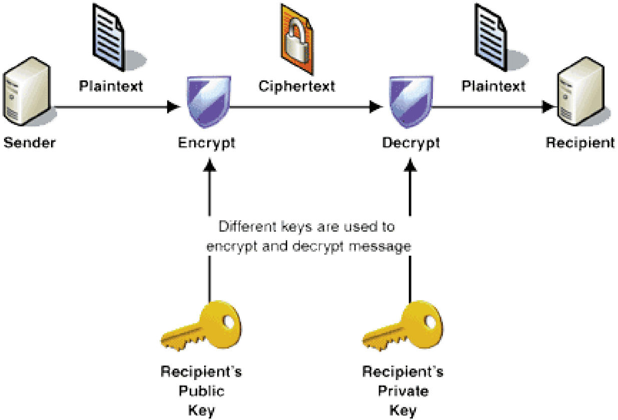
- Plaintext
- Encryption algorithm
- Public and private keys
- Ciphertext
- Decryption algorithm

Public-Key Encryption Scheme



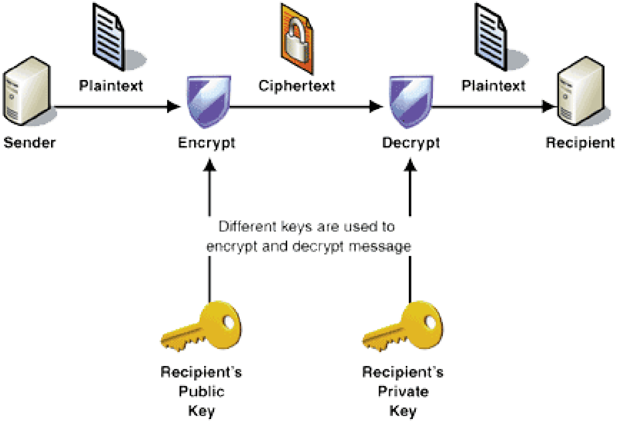
1. Each user generates a pair of keys to be used for the encryption and decryption of messages.

Public-Key Encryption Scheme



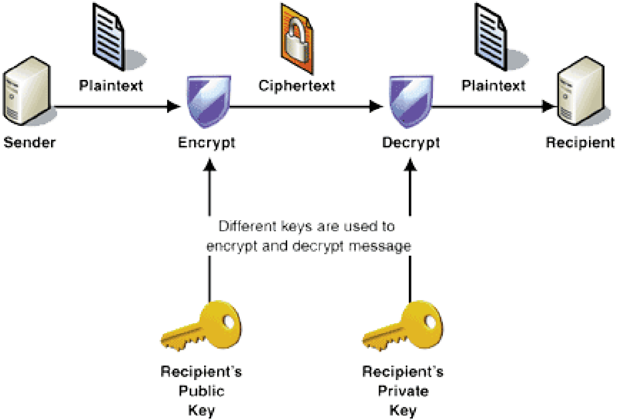
2. Each of the users places one of the keys in a public key register or other accessible file. This is the public key. The companion key is kept private. Each user maintains a collection of public keys obtained from others.

Public-Key Encryption Scheme



3. If I want to send a confidential message to you, I encrypt with your public key.

Public-Key Encryption Scheme



4. When you receive the ciphertext, you decrypt with your private key. No other recipient can decrypt the message because only you know your private key.

Rivest, Shamir and Adleman

Each asymmetric (public-key) cipher depends on the practical irreversibility of some process, usually referred to as a trapdoor.

Rivest, Shamir and Adleman

Each asymmetric (public-key) cipher depends on the practical irreversibility of some process, usually referred to as a trapdoor.

The RSA cipher uses the fact that, while not hard to compute $n = pq$ of two large primes p and q (perhaps $\approx 10^{80}$ or higher), to factor a very large integer $n = 10^{160}$ into its prime factors is essentially impossible.

Rivest, Shamir and Adleman

Each asymmetric (public-key) cipher depends on the practical irreversibility of some process, usually referred to as a trapdoor.

The RSA cipher uses the fact that, while not hard to compute $n = pq$ of two large primes p and q (perhaps $\approx 10^{80}$ or higher), to factor a very large integer $n = 10^{160}$ into its prime factors is essentially impossible.

The pioneering paper by Diffie and Hellman in effect challenged cryptologists to come up with a cryptographic algorithm that met the requirements for public-key systems.

Rivest, Shamir and Adleman

Each asymmetric (public-key) cipher depends on the practical irreversibility of some process, usually referred to as a trapdoor.

The RSA cipher uses the fact that, while not hard to compute $n = pq$ of two large primes p and q (perhaps $\approx 10^{80}$ or higher), to factor a very large integer $n = 10^{160}$ into its prime factors is essentially impossible.

The pioneering paper by Diffie and Hellman in effect challenged cryptologists to come up with a cryptographic algorithm that met the requirements for public-key systems.

One of the first successful attempts to the challenge was developed at MIT in 1977 by Rivest, Shamir and Adleman and was first published in 1978. Since that time, the RSA algorithm has reigned supreme as the most widely accepted public-key scheme.

RSA Scheme

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and $n - 1$ for some n. A typical size for n is 1024 bits, or 309 decimal digits. That is, n is less than 2^{1024}.

RSA Scheme

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and $n - 1$ for some n. A typical size for n is 1024 bits, or 309 decimal digits. That is, n is less than 2^{1024}.

- RSA makes use of an expression with exponentials.

RSA Scheme

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and $n-1$ for some n. A typical size for n is 1024 bits, or 309 decimal digits. That is, n is less than 2^{1024}.

- RSA makes use of an expression with exponentials.
- Plaintext is encrypted in blocks having a binary value less than some number n. ($\leq \log_2(n) + 1$)

RSA Scheme

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and $n - 1$ for some n. A typical size for n is 1024 bits, or 309 decimal digits. That is, n is less than 2^{1024}.

- RSA makes use of an expression with exponentials.
- Plaintext is encrypted in blocks having a binary value less than some number n. ($\leq \log_2(n) + 1$)
- In practice, the block size is i bits, where $2^i < n \leq 2^{i+1}$.

RSA Scheme

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and $n-1$ for some n. A typical size for n is 1024 bits, or 309 decimal digits. That is, n is less than 2^{1024}.

- RSA makes use of an expression with exponentials.
- Plaintext is encrypted in blocks having a binary value less than some number n. ($\leq \log_2(n) + 1$)
- In practice, the block size is i bits, where $2^i < n \leq 2^{i+1}$.

Encryption and decryption are of the following form, for some plaintext block M and ciphertext block C.

$$C \equiv M^e \pmod{n}$$

$$M \equiv C^d \pmod{n} equiv (M^e)^d \pmod{n} \equiv M^{ed} \pmod{n}$$

RSA Scheme

- Both the sender and receiver must know the value of n.

RSA Scheme

- Both the sender and receiver must know the value of n.
- Only the sender knows the value of e.

RSA Scheme

- Both the sender and receiver must know the value of n.
- Only the sender knows the value of e.
- Only the receiver knows the value of d.

RSA Scheme

- Both the sender and receiver must know the value of n.
- Only the sender knows the value of e.
- Only the receiver knows the value of d.
- This is a public-key encryption algorithm with a public key of $PU = \{e, n\}$ and a private key of $PR = \{d, n\}$.

Requirements

For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

1. It is possible to find values of of e, d, n such that $M^{ed} \pmod{n} \equiv M$ for all $M < n$

Requirements

For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

1. It is possible to find values of of e, d, n such that $M^{ed} \pmod{n} \equiv M$ for all $M < n$
2. It is relatively easy to calculate $M^e \pmod{n}$ and $C^d \pmod{n}$ for all values of $M < n$

Requirements

For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

1. It is possible to find values of of e, d, n such that $M^{ed} \pmod{n} \equiv M$ for all $M < n$
2. It is relatively easy to calculate $M^e \pmod{n}$ and $C^d \pmod{n}$ for all values of $M < n$
3. It is infeasible to determine d given e and n

First Requirement

We need to find a relationship of the form

$$M^{ed} \pmod{n} \equiv M$$

The preceding relationship holds if d and e are multiplicative inverses of each other, modulo $\phi(n)$, where $\phi(n)$ is the Euler totient function.

First Requirement

We need to find a relationship of the form

$$M^{ed} \pmod{n} \equiv M$$

The preceding relationship holds if d and e are multiplicative inverses of each other, modulo $\phi(n)$, where $\phi(n)$ is the Euler totient function.

Euler Totient Function

$\phi(n)$ of a positive integer n is defined to be the number of positive integers less than or equal to n that are relatively prime to n.

First Requirement

We need to find a relationship of the form

$$M^{ed} \pmod{n} \equiv M$$

The preceding relationship holds if d and e are multiplicative inverses of each other, modulo $\phi(n)$, where $\phi(n)$ is the Euler totient function.

Euler Totient Function

$\phi(n)$ of a positive integer n is defined to be the number of positive integers less than or equal to n that are relatively prime to n.

When p is prime, however, this reduces to

$$\phi(p) = p\left(1 - \frac{1}{p}\right) = p\left(\frac{p-1}{p}\right) = p - 1$$

So, for our situation we have

$$n = pq \Rightarrow \phi(n) = \phi(pq) = (p-1)(q-1)$$

Euler Totient Function

Example

$\phi(10) = 4$ because $\{1, 3, 7, 9\}$ are all relatively prime with 10.

Euler Totient Function

Example

$\phi(10) = 4$ because $\{1, 3, 7, 9\}$ are all relatively prime with 10.

How the function works: If $n = p_1^{k_1} \cdot p_2^{k_2} \dots p_m^{k_m}$, then

$$\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\dots\left(1 - \frac{1}{p_m}\right)$$

So in our example, we have

$$\phi(10) = 10\left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{5}\right) = 10\left(\frac{1}{2}\right)\left(\frac{4}{5}\right) = 4$$

Relationship between e and d

The relationship between e and d can be expressed as

$$ed \pmod{\phi(n)} \equiv 1$$

Relationship between e and d

The relationship between e and d can be expressed as

$$ed \pmod{\phi(n)} \equiv 1$$

$$ed \equiv 1 \pmod{\phi(n)}$$

Relationship between e and d

The relationship between e and d can be expressed as

$$\begin{aligned} ed \pmod{\phi(n)} &\equiv 1 \\ ed &\equiv 1 \pmod{\phi(n)} \\ d &\equiv e^{-1} \pmod{\phi(n)} \end{aligned}$$

Relationship between e and d

The relationship between e and d can be expressed as

$$\begin{aligned} ed \pmod{\phi(n)} &\equiv 1 \\ ed &\equiv 1 \pmod{\phi(n)} \\ d &\equiv e^{-1} \pmod{\phi(n)} \end{aligned}$$

Note that this is only true if d (and therefore e as well) are relatively prime to $\phi(n)$. Equivalently, $(\phi(n), d) = 1$.

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen
- $n = pq$, public and calculated

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen
- $n = pq$, public and calculated
- e, with $(\phi(n), e) = 1$, $1 < e < \phi(n))$, public and chosen

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen
- $n = pq$, public and calculated
- e, with $(\phi(n), e) = 1$, $1 < e < \phi(n))$, public and chosen
- $d \equiv e^{-1} \pmod{\phi(n)}$, private and calculated

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen
- $n = pq$, public and calculated
- e, with $(\phi(n), e) = 1$, $1 < e < \phi(n))$, public and chosen
- $d \equiv e^{-1} \pmod{\phi(n)}$, private and calculated

The private key consists of $\{d, n\}$ and the public key consists of $\{e, n\}$.

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen
- $n = pq$, public and calculated
- e, with $(\phi(n), e) = 1$, $1 < e < \phi(n))$, public and chosen
- $d \equiv e^{-1} \pmod{\phi(n)}$, private and calculated

The private key consists of $\{d, n\}$ and the public key consists of $\{e, n\}$.

Suppose that User A has published its public key and that User B wishes to send a message M to A. Then B calculates $C \equiv M^e \pmod{\phi(n)}$ and transmits C.

RSA Scheme

The ingredients are:

- p, q, two private numbers that are private and chosen
- $n = pq$, public and calculated
- e, with $(\phi(n), e) = 1$, $1 < e < \phi(n))$, public and chosen
- $d \equiv e^{-1} \pmod{\phi(n)}$, private and calculated

The private key consists of $\{d, n\}$ and the public key consists of $\{e, n\}$.

Suppose that User A has published its public key and that User B wishes to send a message M to A. Then B calculates $C \equiv M^e \pmod{\phi(n)}$ and transmits C.

On receipt of this ciphertext, User A decrypts by calculating $M \equiv C^d \pmod{\phi(n)}$.

RSA Example

Example

1. Select two prime numbers, $p = 17$ and $q = 11$

RSA Example

Example

1. Select two prime numbers, $p = 17$ and $q = 11$
2. Calculate $n = pq = 17 \cdot 11 = 187$

RSA Example

Example

1. Select two prime numbers, $p = 17$ and $q = 11$
2. Calculate $n = pq = 17 \cdot 11 = 187$
3. Calculate $\phi(n) = (p - 1)(q - 1) = 16 \cdot 10 = 160$

RSA Example

Example

1. Select two prime numbers, $p = 17$ and $q = 11$
2. Calculate $n = pq = 17 \cdot 11 = 187$
3. Calculate $\phi(n) = (p-1)(q-1) = 16 \cdot 10 = 160$
4. Select e such that $(e, 160) = 1$ and $e < 160$ - choose $e = 7$

RSA Example

Example

1. Select two prime numbers, $p = 17$ and $q = 11$
2. Calculate $n = pq = 17 \cdot 11 = 187$
3. Calculate $\phi(n) = (p-1)(q-1) = 16 \cdot 10 = 160$
4. Select e such that $(e, 160) = 1$ and $e < 160$ - choose $e = 7$
5. Determine d such that $de \equiv 1 \pmod{160}$ and $d < 160$. the correct value is $d = 23$, because $23 \cdot 7 = 161 = (1 \cdot 160) + 1$; d can be calculated using the Euclidean Algorithm

RSA Example

Example

1. Select two prime numbers, $p = 17$ and $q = 11$
2. Calculate $n = pq = 17 \cdot 11 = 187$
3. Calculate $\phi(n) = (p-1)(q-1) = 16 \cdot 10 = 160$
4. Select e such that $(e, 160) = 1$ and $e < 160$ - choose $e = 7$
5. Determine d such that $de \equiv 1 \pmod{160}$ and $d < 160$. the correct value is $d = 23$, because $23 \cdot 7 = 161 = (1 \cdot 160) + 1$; d can be calculated using the Euclidean Algorithm

The resulting keys are the public key $\{7, 187\}$ and the private key $\{23, 187\}$.

RSA Example - Encryption

So suppose we wanted to use these keys for a plaintext input of $M = 88$.

RSA Example - Encryption

So suppose we wanted to use these keys for a plaintext input of $M = 88$.

We need to calculate $C \equiv 88^7 \pmod{187}$.

RSA Example - Encryption

So suppose we wanted to use these keys for a plaintext input of $M = 88$.

We need to calculate $C \equiv 88^7 \pmod{187}$.

$$\begin{aligned} 88^7 \pmod{187} = {} & [(88^4 \pmod{187}) \times (88^2 \pmod{187}) \\ & \times [88^1 \pmod{187})] \pmod{187} \end{aligned}$$

RSA Example - Encryption

So suppose we wanted to use these keys for a plaintext input of $M = 88$.

We need to calculate $C \equiv 88^7 \pmod{187}$.

$$\begin{aligned} 88^7 \pmod{187} = {} & [(88^4 \pmod{187}) \times (88^2 \pmod{187}) \\ & \times [88^1 \pmod{187})] \pmod{187} \end{aligned}$$

$$\begin{aligned} 88^1 \pmod{187} &\equiv 88 \\ 88^2 \pmod{187} \equiv 7744 \pmod{187} &\equiv 77 \\ 88^4 \pmod{187} \equiv 77^2 \pmod{187} &\equiv 132 \\ 88^7 \pmod{187} \equiv (88 \times 77 \times 132) \pmod{187} &\equiv 11 \end{aligned}$$

RSA Example - Decryption

We calculate $M \equiv 11^{23} \pmod{187}$.

RSA Example - Decryption

We calculate $M \equiv 11^{23} \pmod{187}$.

$$11^{23} = 11^1 \times 11^2 \times 11^4 \times 11^8 \times 11^8$$

$$\begin{aligned}
11^1 \pmod{187} &\equiv 11 \\
11^2 \pmod{187} &\equiv 121 \\
11^4 \pmod{187} \equiv 14641 \pmod{187} &\equiv 55 \\
11^8 \pmod{187} \equiv 55^2 \pmod{187} &\equiv 33
\end{aligned}$$

RSA Example - Decryption

We calculate $M \equiv 11^{23} \pmod{187}$.

$$11^{23} = 11^1 \times 11^2 \times 11^4 \times 11^8 \times 11^8$$

$$\begin{aligned} 11^1 \pmod{187} &\equiv 11 \\ 11^2 \pmod{187} &\equiv 121 \\ 11^4 \pmod{187} \equiv 14641 \pmod{187} &\equiv 55 \\ 11^8 \pmod{187} \equiv 55^2 \pmod{187} &\equiv 33 \end{aligned}$$

So,

$$11^{23} \pmod{187} \equiv (11 \times 121 \times 55 \times 33^2) \pmod{187} \equiv 88$$

RSA Security

The security of RSA more or less depends upon the difficulty of factorization of integers into primes. This seems to be a genuinely difficult problem.

RSA Security

The security of RSA more or less depends upon the difficulty of factorization of integers into primes. This seems to be a genuinely difficult problem.

More precisely, security of RSA depends on a much more special problem, the difficulty of factoring integers of the special form $n = pq$ into primes.

RSA Security

The security of RSA more or less depends upon the difficulty of factorization of integers into primes. This seems to be a genuinely difficult problem.

More precisely, security of RSA depends on a much more special problem, the difficulty of factoring integers of the special form $n = pq$ into primes.

The reason that difficulty of factorization makes RSA secure is that for n the product of two big primes (with the primes being secret), it seems hard to compute $\phi(n)$ when only n is given.

Finding p and q

If we know the factorization $n = pq$, it is easy to calculate $\phi(n)$ by $\phi(n) = \phi(pq) = (p-1)(q-1)$.

Finding p and q

If we know the factorization $n = pq$, it is easy to calculate $\phi(n)$ by $\phi(n) = \phi(pq) = (p-1)(q-1)$.

If we know n and $\phi(n)$, we can find p and q as well. The trick is based on the fact that p and q are roots of

$$x^2 - (p+q)x + pq = 0$$

Finding p and q

If we know the factorization $n = pq$, it is easy to calculate $\phi(n)$ by $\phi(n) = \phi(pq) = (p-1)(q-1)$.

If we know n and $\phi(n)$, we can find p and q as well. The trick is based on the fact that p and q are roots of

$$x^2 - (p+q)x + pq = 0$$

Already, $pq = n$, so we can express $p+q$ in terms of n and $\phi(n)$. Since

$$\begin{aligned}\phi(n) &= (p-1)(q-1) \\ &= pq - (p+q) + 1 \\ &= n - (p+q) + 1\end{aligned}$$

we can rearrange to get $p + q = n - \phi(n) + 1$.

Finding p and q

If we know the factorization $n = pq$, it is easy to calculate $\phi(n)$ by $\phi(n) = \phi(pq) = (p-1)(q-1)$.

If we know n and $\phi(n)$, we can find p and q as well. The trick is based on the fact that p and q are roots of

$$x^2 - (p+q)x + pq = 0$$

Already, $pq = n$, so we can express $p+q$ in terms of n and $\phi(n)$. Since

$$\begin{aligned} \phi(n) &= (p-1)(q-1) \\ &= pq - (p+q) + 1 \\ &= n - (p+q) + 1 \end{aligned}$$

we can rearrange to get $p + q = n - \phi(n) + 1$.
Therefore, p and q are the roots of

$$x^2 - (n - \phi(n) + 1)x + n = 0$$

Which can be solved using the quadratic formula.

Key Generation and Management

To set up a modulus $n = pq$ from secret primes p and q and to determine a key pair e and d with $ed \equiv 1 \pmod{\phi(n)}$, requires first of all two large primes p and q, each at least as large as 10^{80}.

Key Generation and Management

To set up a modulus $n = pq$ from secret primes p and q and to determine a key pair e and d with $ed \equiv 1 \pmod{\phi(n)}$, requires first of all two large primes p and q, each at least as large as 10^{80}.

Since the security of RSA is based upon the intractability of factoring, it is very lucky that primality testing is much easier than factorization into primes. That is, we can obtain many 'large' primes cheaply despite the fact that we can't factor large $n = pq$.

Key Generation and Management

To set up a modulus $n = pq$ from secret primes p and q and to determine a key pair e and d with $ed \equiv 1 \pmod{\phi(n)}$, requires first of all two large primes p and q, each at least as large as 10^{80}.

Since the security of RSA is based upon the intractability of factoring, it is very lucky that primality testing is much easier than factorization into primes. That is, we can obtain many 'large' primes cheaply despite the fact that we can't factor large $n = pq$.

The decryption key d (private) can be chosen first, after pq.

Key Generation and Management

To set up a modulus $n = pq$ from secret primes p and q and to determine a key pair e and d with $ed \equiv 1 \pmod{\phi(n)}$, requires first of all two large primes p and q, each at least as large as 10^{80}.

Since the security of RSA is based upon the intractability of factoring, it is very lucky that primality testing is much easier than factorization into primes. That is, we can obtain many 'large' primes cheaply despite the fact that we can't factor large $n = pq$.

The decryption key d (private) can be chosen first, after pq.

For there to be a corresponding encryption key e it must be that d is relatively prime to $(p-1)(q-1)$, and the Euclidean Algorithm gives an efficient means to compute e. If $gcd > 1$, we can guess another d.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.
2. For technical reasons, some people have recently recommended $2^{16} + 1 = 65537$ (prime) as the encryption exponent.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.
2. For technical reasons, some people have recently recommended $2^{16} + 1 = 65537$ (prime) as the encryption exponent.
3. Both $p - 1$ and $q - 1$ should have at least one very large prime factor, since there are factorization attacks if $p - 1$ and $q - 1$ have only smallish primes.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.
2. For technical reasons, some people have recently recommended $2^{16} + 1 = 65537$ (prime) as the encryption exponent.
3. Both $p - 1$ and $q - 1$ should have at least one very large prime factor, since there are factorization attacks if $p - 1$ and $q - 1$ have only smallish primes.
4. The primes p and q should not be 'close' to each other - there are factorization attacks that succeed in this case.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.
2. For technical reasons, some people have recently recommended $2^{16} + 1 = 65537$ (prime) as the encryption exponent.
3. Both $p - 1$ and $q - 1$ should have at least one very large prime factor, since there are factorization attacks if $p - 1$ and $q - 1$ have only smallish primes.
4. The primes p and q should not be 'close' to each other - there are factorization attacks that succeed in this case.
5. The ratio $\frac{p}{q}$ should not be 'close' to a rational number with smallish numerator and denominator since Lehmer's Continuous Fraction Factorization attack on $n = pq$ will succeed.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.
2. For technical reasons, some people have recently recommended $2^{16} + 1 = 65537$ (prime) as the encryption exponent.
3. Both $p - 1$ and $q - 1$ should have at least one very large prime factor, since there are factorization attacks if $p - 1$ and $q - 1$ have only smallish primes.
4. The primes p and q should not be 'close' to each other - there are factorization attacks that succeed in this case.
5. The ratio $\frac{p}{q}$ should not be 'close' to a rational number with smallish numerator and denominator since Lehmer's Continuous Fraction Factorization attack on $n = pq$ will succeed.
6. p and q should differ in length by only a few digits.

Notes

1. Sometimes the encryption exponent is taken to be 3 with p and q not $1 \pmod 3$.
2. For technical reasons, some people have recently recommended $2^{16} + 1 = 65537$ (prime) as the encryption exponent.
3. Both $p - 1$ and $q - 1$ should have at least one very large prime factor, since there are factorization attacks if $p - 1$ and $q - 1$ have only smallish primes.
4. The primes p and q should not be 'close' to each other - there are factorization attacks that succeed in this case.
5. The ratio $\frac{p}{q}$ should not be 'close' to a rational number with smallish numerator and denominator since Lehmer's Continuous Fraction Factorization attack on $n = pq$ will succeed.
6. p and q should differ in length by only a few digits.
7. $gcd(p - 1, q - 1)$ should be small.

Attacking RSA

At present, it seems that attacks on RSA will succeed only when RSA is improperly implemented, for example, with too small modulus. That is, there seems to be exploitable weaknesses when certain avoidable mistakes are made.

Attacking RSA

At present, it seems that attacks on RSA will succeed only when RSA is improperly implemented, for example, with too small modulus. That is, there seems to be exploitable weaknesses when certain avoidable mistakes are made.

If the key size $n = pq$ is too small, a brute force attack to factor n may succeed in a smaller time than one would want.

Mathematical Attacks

This is equivalent to factoring the product of two primes.

Mathematical Attacks

This is equivalent to factoring the product of two primes.

1. Factor n into p and q. This enables the calculation of $\phi(n)$, which enables the determination of $d \equiv e^{-1} \pmod{\phi(n)}$.

Mathematical Attacks

This is equivalent to factoring the product of two primes.

1. Factor n into p and q. This enables the calculation of $\phi(n)$, which enables the determination of $d \equiv e^{-1} \pmod{\phi(n)}$.
2. Determine $\phi(n)$ directly without determining p or q.

Mathematical Attacks

This is equivalent to factoring the product of two primes.

1. Factor n into p and q. This enables the calculation of $\phi(n)$, which enables the determination of $d \equiv e^{-1} \pmod{\phi(n)}$.
2. Determine $\phi(n)$ directly without determining p or q.
3. Determine d directly without first determining $\phi(n)$.

Timing Attack

This is analagous to a burglar guessing a combination for a safe by observing how long it takes for someone to turn the dial from number to number. This would be a computer attack where we measure how long it takes for exponentiation.

Timing Attack

This is analagous to a burglar guessing a combination for a safe by observing how long it takes for someone to turn the dial from number to number. This would be a computer attack where we measure how long it takes for exponentiation.

Steps against this kind of attack:

1. Constant exponentiation time

Timing Attack

This is analagous to a burglar guessing a combination for a safe by observing how long it takes for someone to turn the dial from number to number. This would be a computer attack where we measure how long it takes for exponentiation.

Steps against this kind of attack:

1. Constant exponentiation time
2. Random delay

Timing Attack

This is analagous to a burglar guessing a combination for a safe by observing how long it takes for someone to turn the dial from number to number. This would be a computer attack where we measure how long it takes for exponentiation.

Steps against this kind of attack:

1. Constant exponentiation time
2. Random delay
3. Blinding: multiply the ciphertext by a random number before exponentiation. This prevents bit by bit analysis since the attacker won't know the true cipher bits.

Chosen Ciphertext Attack

An adversary chooses a number of ciphertexts and is given corresponding plaintexts decrypted by the private key.

Chosen Ciphertext Attack

An adversary chooses a number of ciphertexts and is given corresponding plaintexts decrypted by the private key.

Not feasible if a real situation where a hacker is trying to defeat RSA.