An Introduction to Cryptography

Spotlight on Science

J. Robert Buchanan

Department of Mathematics

25 March 2009
What is Cryptography?

**cryptography**: study of methods for sending messages in a form that only be understood by the recipient.

**cryptanalysis**: study of methods for defeating cryptography.

**cryptology**: study of both cryptography and cryptanalysis.
Cryptology Throughout History (1 of 3)

- Subject as old as recorded history.
Subject as old as recorded history.

Julius Caesar used substitution ciphers for military and personal communication.
Enigma was patented by Arthur Scherbius in 1918 and became the foundation of German and Japanese codes during WWII. (Photo credit: JRB)
Since WWII the foundation of cryptology has been **mathematics**, particularly **number theory**.

(Photo credit: JRB)
Terminology

**Plaintext:** the un-encrypted, clearly readable form of a message.

**Ciphertext:** the encrypted form of the plaintext, readable only with the key to the cipher.
In some cryptographic systems, the key used to encipher the message must be kept hidden from the public and should be known only by the sender and recipient of the message.

There are other cryptographic systems in which each party has a **private key** (which they keep secret) and a **public key** which they publish so that others may send them encrypted messages which only they can decrypt. The public key functions as the recipient’s cryptographic “telephone number”.
Overview of Public Key Cryptography

Suppose Alice wants to send Bob a message which only Bob will be able to read.

1. Alice looks up Bob’s public key and enciphers the plaintext using his public key.

2. Alice sends Bob the ciphertext (or even publishes it in the newspaper!).

3. Bob decrypts the ciphertext using his private key.
Overview of Public Key Cryptography

Suppose Alice wants to send Bob a message which only Bob will be able to read.

1. Alice looks up Bob’s public key and enciphers the plaintext using his public key.
2. Alice sends Bob the ciphertext (or even publishes it in the newspaper!).
Overview of Public Key Cryptography

Suppose Alice wants to send Bob a message which only Bob will be able to read.

1. Alice looks up Bob’s public key and enciphers the plaintext using his public key.
2. Alice sends Bob the ciphertext (or even publishes it in the newspaper!).
3. Bob decrypts the ciphertext using his private key.
Outline

1. Introduction

2. Public Key Cryptography
   - RSA
   - Euclidean Algorithm and Friends
   - Euler Phi Function
   - How it all works

3. For More Information
One of the most popular public key cryptographic systems is known as RSA.

It was developed by

Adi Shamir
Ronald Rivest
Leonard Adelman.
A Little Number Theory I

Definition

A **prime number** is a whole number $p$ greater than 1 whose only divisors are $p$ and 1. An integer which is not prime is called a **composite number**.
Definition

If \( \gcd(a, b) = 1 \) we say that \( a \) and \( b \) are relatively prime.
Examples

Example

\[ 4 \equiv \text{gcd}(16, 60) \]
\[ 13 \equiv \text{gcd}(13, 65) \]
\[ ? \equiv \text{gcd}(72, 81) \]
Outline

1. Introduction

2. Public Key Cryptography
   - RSA
   - Euclidean Algorithm and Friends
   - Euler Phi Function
   - How it all works

3. For More Information
Euclidean Algorithm

To find the gcd\((a, b)\):

1. If \(b\) divides \(a\) evenly, \(\gcd(a, b) = b\).
2. If not, replace \(a\) by \(b\) and replace \(b\) by the remainder of \(a \div b\) and go back to step 1.

A Java applet implementing the Euclidean Algorithm is also available.
Example

Find $\gcd(105, 56)$ using the Euclidean Algorithm.
Example

Find $\gcd(105, 56)$ using the Euclidean Algorithm.

$$
\begin{array}{c|c}
 a & b \\
 105 & 56 \\
 56 & 49 \\
 49 & 7 \\
\end{array}
$$

$$\gcd(105, 56) = 7$$
Definition

If $a$ and $b$ are integers, then an expression of the form $ax + by$ where $x$ and $y$ are also integers is called a **linear combination of $a$ and $b$**.
Definition

If $a$ and $b$ are integers, then an expression of the from $ax + by$ where $x$ and $y$ are also integers is called a linear combination of $a$ and $b$.

Theorem

If $a$ and $b$ are integers, then $\gcd(a, b)$ is the smallest positive integer which can be written as a linear combination of $a$ and $b$. 
Modular Arithmetic

Definition

Two integers \( n \) and \( m \) are congruent modulo \( k \) (or equivalent modulo \( k \)) written symbolically as

\[
    n \equiv m \pmod{k}
\]

when \( k \) divides \( n - m \) evenly.

Example

\[
    65 \equiv 13 \pmod{26} \quad \text{since} \quad \frac{65 - 13}{26} = 2.
\]
Modular Arithmetic

Definition

Two integers \( n \) and \( m \) are congruent modulo \( k \) (or equivalent modulo \( k \)) written symbolically as

\[ n \equiv m \pmod{k} \]

when \( k \) divides \( n - m \) evenly.

Example

\[ 65 \equiv 13 \pmod{26} \quad \text{since} \quad \frac{65 - 13}{26} = 2. \]

Corollary

*If \( a \) and \( b \) are relatively prime then \( b \) has a multiplicative inverse modulo \( a \).*
Extended Euclidean Algorithm

To find $x$ and $y$ such that

$$\gcd(a, b) = ax + by$$

use the **tabular method**.

Start the table like this:

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
<th>$d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>$a$</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>$b$</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
Extended Euclidean Algorithm

To find $x$ and $y$ such that

$$\gcd(a, b) = ax + by$$

use the **tabular method**.

Start the table like this:

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
<th>$d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>$a$</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>$b$</td>
</tr>
<tr>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

To fill in the next row:

1. Find $q$ and $r$ such that
   $$\frac{a}{b} = q + \frac{r}{b}.$$
2. Multiply next to last row by $q$ and subtract from third to last row.
3. Keep appending rows to the table until $r = 0$. 

J. Robert Buchanan (Millersville Univ.)
Example

\[
\begin{array}{ccc}
 x & y & d \\
 1 & 0 & 105 \\
 0 & 1 & 56 \\
 1 & -1 & 49 \\
 -1 & 2 & 7 \\
\end{array}
\]

\((-1)(105) + (2)(56) = 7\)

A Java applet implementing the Extended Euclidean Algorithm is also available.
Finding a Multiplicative Inverse

Verify that 23 and 29 are relatively prime and find the multiplicative inverse of 29 modulo 23.

\[
\begin{array}{ccc}
  x & y & d \\
  1 & 0 & 29 \\
  0 & 1 & 23 \\
  1 & -1 & 6 \\
  -3 & 4 & 5 \\
  4 & -5 & 1 \\
\end{array}
\]

\[
1 \equiv (4)(29) + (-5)(23) \pmod{23}
\]

\[
1 \equiv (4)(29) \pmod{23}
\]

So 4 is the multiplicative inverse of 29 modulo 23.
Outline

1 Introduction

2 Public Key Cryptography
   - RSA
   - Euclidean Algorithm and Friends
   - Euler Phi Function
   - How it all works

3 For More Information
Definition

If $\mathbb{N}$ is the set of all positive integers then for each positive integer $n$ define $\phi(n)$ to be the number of positive integers less than or equal to $n$ which are relatively prime to $n$. Such a function $\phi$ is called the Euler phi function.
Do you notice a pattern?

\[\phi(n)\]
### Euler Phi Function (2 of 2)

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\phi(n)$</th>
<th>$n$</th>
<th>$\phi(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>10</td>
<td>4</td>
</tr>
</tbody>
</table>
Euler Phi Function (2 of 2)

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\phi(n)$</th>
<th>$n$</th>
<th>$\phi(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>10</td>
<td>4</td>
</tr>
</tbody>
</table>

Lemma

If $p$ is prime, then $\phi(p) = p - 1$. 
Euler Phi Function (2 of 2)

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\phi(n)$</th>
<th>$n$</th>
<th>$\phi(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>10</td>
<td>4</td>
</tr>
</tbody>
</table>

**Lemma**

*If $p$ is prime, then $\phi(p) = p - 1$.***

**Lemma**

*If $p$ and $q$ are distinct primes, then

$$\phi(pq) = \phi(p)\phi(q) = (p - 1)(q - 1).$$*
Outline

1. Introduction

2. Public Key Cryptography
   - RSA
   - Euclidean Algorithm and Friends
   - Euler Phi Function
   - How it all works

3. For More Information
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes \( p \) and \( q \),

2. let \( p = 1009 \) and \( q = 2003 \).

3. forms the product \( n = pq \),

4. computes \( \phi(n) \),

5. selects integer \( e \) such that \( \gcd(e, \phi(n)) = 1 \),

6. finds the multiplicative inverse \( d \) of \( e \) \((mod \ \phi(n))\),

7. private key is \( d \), public key is \( (e, n) \).
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes \( p \) and \( q \),
2. forms the product \( n = pq \),

1. let \( p = 1009 \) and \( q = 2003 \).
2. \( n = 2021027 \)
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes \( p \) and \( q \),
2. forms the product \( n = pq \),
3. computes \( \phi(n) \),
4. picks an integer \( e \) such that \( \gcd(e, \phi(n)) = 1 \),
5. finds the multiplicative inverse \( d \) of \( e \) modulo \( \phi(n) \),
6. private key is \( d \), public key is \( (e, n) \).

Example:

1. let \( p = 1009 \) and \( q = 2003 \).
2. \( n = 2021027 \)
3. \( \phi(2021027) = (1009 - 1)(2003 - 1) = 2018016 \)
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes $p$ and $q$,
2. forms the product $n = pq$,
3. computes $\phi(n)$,
4. selects integer $e$ such that $\gcd(e, \phi(n)) = 1$,

1. let $p = 1009$ and $q = 2003$.
2. $n = 2021027$
3. $\phi(2021027) = (1009 - 1)(2003 - 1) = 2018016$
4. $e = 1003$ (for example)
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes $p$ and $q$,
2. forms the product $n = pq$,
3. computes $\phi(n)$,
4. selects integer $e$ such that $\gcd(e, \phi(n)) = 1$,
5. finds the multiplicative inverse $d$ of $e \pmod{\phi(n)}$,

1. let $p = 1009$ and $q = 2003$.
2. $n = 2021027$
3. $\phi(2021027) = (1009 - 1)(2003 - 1) = 2018016$
4. $e = 1003$ (for example)
5. $d = 1311811$ (for this $e$)

Remark: only the Key Utility knows $p$ and $q$. 
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes \( p \) and \( q \),
2. forms the product \( n = pq \),
3. computes \( \phi(n) \),
4. selects integer \( e \) such that \( \gcd(e, \phi(n)) = 1 \),
5. finds the multiplicative inverse \( d \) of \( e \) \( \pmod{\phi(n)} \),
6. private key is \( d \), public key is \( (e, n) \).

Example:

1. let \( p = 1009 \) and \( q = 2003 \).
2. \( n = 2021027 \)
3. \( \phi(2021027) = (1009 - 1)(2003 - 1) = 2018016 \)
4. \( e = 1003 \) (for example)
5. \( d = 1311811 \) (for this \( e \))
6. private key is 1311811, public key is \( (1003, 2021027) \).
Getting Public and Private Keys

A public agency (call it the “Key Utility”)

1. selects two large primes \( p \) and \( q \),
2. forms the product \( n = pq \),
3. computes \( \phi(n) \),
4. selects integer \( e \) such that \( \gcd(e, \phi(n)) = 1 \),
5. finds the multiplicative inverse \( d \) of \( e \) \( \pmod{\phi(n)} \),
6. private key is \( d \), public key is \( (e, n) \).

Remark: only the Key Utility knows \( p \) and \( q \).

1. let \( p = 1009 \) and \( q = 2003 \).
2. \( n = 2021027 \)
3. \( \phi(2021027) = (1009 - 1)(2003 - 1) = 2018016 \)
4. \( e = 1003 \) (for example)
5. \( d = 1311811 \) (for this \( e \))
6. private key is 1311811, public key is \( (1003, 2021027) \).
Exchanging Messages

Suppose Alice and Bob have the following keys.

<table>
<thead>
<tr>
<th>Key Holder</th>
<th>Public Key</th>
<th>Private Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>$(e_a, n_a) = (1003, 2021027)$</td>
<td>$d_a = 1311811$</td>
</tr>
<tr>
<td>Bob</td>
<td>$(e_b, n_b) = (10007, 2021027)$</td>
<td>$d_b = 1197863$</td>
</tr>
</tbody>
</table>
Exchanging Messages

Suppose Alice and Bob have the following keys.

<table>
<thead>
<tr>
<th>Key Holder</th>
<th>Public Key</th>
<th>Private Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>((e_a, n_a) = (1003, 2021027))</td>
<td>(d_a = 1311811)</td>
</tr>
<tr>
<td>Bob</td>
<td>((e_b, n_b) = (10007, 2021027))</td>
<td>(d_b = 1197863)</td>
</tr>
</tbody>
</table>

Private key \(d_a\) is known only to Alice (and the Key Utility), while private key \(d_b\) is known only to Bob (and the Key Utility).
Exchanging Messages

Suppose Alice and Bob have the following keys.

<table>
<thead>
<tr>
<th>Key Holder</th>
<th>Public Key</th>
<th>Private Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>$(e_a, n_a) = (1003, 2021027)$</td>
<td>$d_a = 1311811$</td>
</tr>
<tr>
<td>Bob</td>
<td>$(e_b, n_b) = (10007, 2021027)$</td>
<td>$d_b = 1197863$</td>
</tr>
</tbody>
</table>

Private key $d_a$ is known only to Alice (and the Key Utility), while private key $d_b$ is known only to Bob (and the Key Utility).

If Alice wants to send the message “hi” to Bob, she looks up Bob’s public key $(e_b, n_b)$ and must do two things.
Encrypting Using RSA

1. Alice converts “hi” to a number, $m$.

<table>
<thead>
<tr>
<th>Letter</th>
<th>h</th>
<th>i</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCII Code</td>
<td>104</td>
<td>105</td>
</tr>
</tbody>
</table>

$m = 104105$
Encrypting Using RSA

1. Alice converts “hi” to a number, $m$.

<table>
<thead>
<tr>
<th>Letter</th>
<th>h</th>
<th>i</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCII Code</td>
<td>104</td>
<td>105</td>
</tr>
</tbody>
</table>

$m = 104105$

2. Alice creates the ciphertext $c$ using the formula

$$c \equiv m^{e_b} \pmod{n_b}$$

$$c = 104105^{10007} \pmod{2021027}$$

$$c = 1086444$$

and sends it to Bob. Alice can try this Java applet if she needs help with modular exponentiation.
Encrypting Using RSA

1. Alice converts “hi” to a number, $m$.

<table>
<thead>
<tr>
<th>Letter</th>
<th>h</th>
<th>i</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCII Code</td>
<td>104</td>
<td>105</td>
</tr>
</tbody>
</table>

$m = 104105$

2. Alice creates the ciphertext $c$ using the formula

$$c \equiv m^{eb} \pmod{n_b}$$

$$c = 104105^{10007} \pmod{2021027}$$

$$c = 1086444$$

and sends it to Bob. Alice can try this Java applet if she needs help with modular exponentiation.

Remark: if $m > n_b$ Alice breaks the $m$ into chunks smaller than $n_b$. 
Bob Decrypts the Ciphertext

When Bob receives the ciphertext $c$ he uses his private key $d_b$ and computes

$$m' \equiv c^{d_b} \pmod{n_b}$$

$$m' = 1086444^{1197863} \pmod{2021027}$$

$$m' = 104105.$$
When Bob receives the ciphertext $c$ he uses his private key $d_b$ and computes

$$m' \equiv c^{d_b} \pmod{n_b}$$

$$m' = 1086444^{1197863} \pmod{2021027}$$

$$m' = 104105.$$
The security of RSA depends on the difficulty in factoring large numbers like $n$. 

If $n = pq$ where $p$ and $q$ are prime numbers with more than 500 digits each then $n$ has more than 1000 digits. It takes approximately $4.7 \times 10^{22}$ years to factor a 500-digit composite number using today's computers. As computers get faster, we can keep increasing $n$. 

J. Robert Buchanan (Millersville Univ.)
Is RSA Secure?

- The security of RSA depends on the difficulty in factoring large numbers like $n$.
- If $n = pq$ where $p$ and $q$ are prime numbers with more than 500 digits each then $n$ has more than 1000 digits.
Is RSA Secure?

- The security of RSA depends on the difficulty in factoring large numbers like $n$.
- If $n = pq$ where $p$ and $q$ are prime numbers with more than 500 digits each then $n$ has more than 1000 digits.
- It takes approximately $4.7 \times 10^{22}$ years to factor a 500-digit composite number using today’s computers.
Is RSA Secure?

- The security of RSA depends on the difficulty in factoring large numbers like $n$.
- If $n = pq$ where $p$ and $q$ are prime numbers with more than 500 digits each then $n$ has more than 1000 digits.
- It takes approximately $4.7 \times 10^{22}$ years to factor a 500-digit composite number using today’s computers.
- As computers get faster, we can keep increasing $n$. 
Will We Run Out of Primes?

Theorem

*There are infinitely many prime numbers.*
Will We Run Out of Primes?

Theorem

There are infinitely many prime numbers.

Proof.

- Suppose there are only finitely many primes $p_1 < p_2 < \cdots < p_k$. 

Form the number $q = (p_1)(p_2) \cdots (p_k) + 1 > p_k$. This number is either prime or composite. 

No (known) prime evenly divides $q$. 

Either $q$ itself is prime or there is a prime larger than $p_k$ which divides $q$. 

J. Robert Buchanan (Millersville Univ.)
Will We Run Out of Primes?

Theorem

There are infinitely many prime numbers.

Proof.

- Suppose there are only finitely many primes \( p_1 < p_2 < \cdots < p_k \).
- Form the number \( q = (p_1)(p_2) \cdots (p_k) + 1 > p_k \). This number is either prime or composite.

\[ \equiv q \pmod{p_i} \]

No (known) prime evenly divides \( q \).

Either \( q \) itself is prime or there is a prime larger than \( p_k \) which divides \( q \).
Will We Run Out of Primes?

Theorem
There are infinitely many prime numbers.

Proof.
- Suppose there are only finitely many primes \( p_1 < p_2 < \cdots < p_k \).
- Form the number \( q = (p_1)(p_2) \cdots (p_k) + 1 > p_k \). This number is either prime or composite.
- \( 1 \equiv q \pmod{p_i} \)
Will We Run Out of Primes?

Theorem

*There are infinitely many prime numbers.*

Proof.

- Suppose there are only finitely many primes $p_1 < p_2 < \cdots < p_k$.
- Form the number $q = (p_1)(p_2) \cdots (p_k) + 1 > p_k$. This number is either prime or composite.
- $1 \equiv q \pmod{p_i}$
- No (known) prime evenly divides $q$. 
Will We Run Out of Primes?

Theorem

*There are infinitely many prime numbers.*

Proof.

- Suppose there are only finitely many primes $p_1 < p_2 < \cdots < p_k$.
- Form the number $q = (p_1)(p_2)\cdots(p_k) + 1 > p_k$. This number is either prime or composite.
- $1 \equiv q \pmod{p_i}$
- No (known) prime evenly divides $q$.
- Either $q$ itself is prime or there is a prime larger than $p_k$ which divides $q$. 
Where Can You Learn More?

- The modern language of cryptology is mathematics. Major in mathematics in college. Take a course in number theory.
- The National Security Agency employs and educates cryptologists.
- The National Cryptologic Museum has exhibits on ancient and modern cryptography.