Prof. Claude Crépeau
Classical Cryptography
Will you marry me?
Will you marry me?
Divorce your wife first!
Will you marry me?

Divorce your wife first!

The papers are in the mail...
Will you marry me?

Divorce your wife first!

The papers are in the mail...

OK, I will!
Information

Theoretical

Cryptography
Information Theoretical Cryptography

- Key Distribution
- Encryption
- Authentication
Key Distribution
Encryption
Will you marry me? Yes! T9@

8RdewtU5qkLa$es!T9@

Decryption

Encryption
Divorce your wife first!
Symmetric Encryption

Encryption

P  K  C

Decryption

Information Theoretical Security
Symmetric Encryption

Ceasar’s Cipher
VERNAM’s Cipher
VERNAM’s Cipher

\[ m \oplus k \]

1 0 1 0 0 1 0 1 0 0 1 0 1 1 1 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1
\( m \oplus k = c \)

\[
\begin{array}{ccc}
1 & 1 & 0 \\
0 & 1 & 1 \\
1 & 1 & 0 \\
0 & 0 & 0 \\
1 & 1 & 0 \\
0 & 1 & 1 \\
0 & 0 & 0 \\
1 & 1 & 0 \\
1 & 1 & 0 \\
1 & 1 & 0 \\
1 & 1 & 0 \\
1 & 0 & 1 \\
1 & 1 & 0 \\
1 & 0 & 1 \\
0 & 1 & 1 \\
0 & 1 & 1 \\
1 & 1 & 0 \\
\end{array}
\]

\[
\oplus_1 =
\]

\[
\begin{array}{ccc}
0 & 1 & 0 \\
0 & 1 & 0 \\
0 & 1 & 0 \\
0 & 0 & 0 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array}
\]
VERNAM’s Cipher

$m ⊕ k = c$

\[
\begin{array}{cccc}
1 & 1 & 0 & \\
0 & 1 & 1 & \\
1 & 1 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
1 & 1 & 0 & \\
0 & 1 & 1 & \\
0 & 0 & 0 & \\
1 & 1 & 0 & \\
1 & 1 & 0 & \\
1 & 0 & 1 & \\
1 & 1 & 0 & \\
1 & 1 & 0 & \\
1 & 1 & 0 & \\
0 & 1 & 1 & \\
0 & 1 & 1 & \\
1 & 1 & 0 & \\
1 & 1 & 0 & \\
1 & 1 & 0 & \\
1 & 1 & 0 & \\
\end{array}
\]

\[=\]

\[
\begin{array}{cccc}
0 & 1 & 0 & \\
0 & 1 & 0 & \\
0 & 0 & 0 & \\
1 & 0 & 0 & \\
0 & 1 & 0 & \\
1 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
0 & 0 & 0 & \\
\end{array}
\]
VERNAM’s Cipher

\[ m \oplus k = c \]

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ c \oplus k \]

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ \oplus = \]

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

23
VERNAM’s Cipher

\[ m \oplus k = c \]

\[ \begin{array}{cccc}
1 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 \\
\end{array} \]

\[ \oplus = \]

\[ \begin{array}{cccc}
1 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 0 & 1 \\
0 & 1 & 1 & 1 \\
\end{array} \]

\[ c \oplus k = m \]

\[ \begin{array}{cccc}
0 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 \\
1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 \\
\end{array} \]
M VERNAM ⊕ K = C

K ⊕ K = M'
M \text{ GILBERT } \oplus K = C \oplus K = M' \text{ GILBERT }
VERNAM’s One-Time Pad

\[ m_1 \oplus k = c_1 \]
\[ m_2 \oplus k = c_2 \]

\[ c_1 \oplus k = m_1 \]
\[ c_2 \oplus k = m_2 \]

\[ c_1 \oplus c_2 = m_1 \oplus m_2 \]
\[ M_0 \text{VERNAM} \oplus C_0 \]

\[ M_1 \text{GILBERT} \]

\[ = \]

\[ X \text{VERNAM} \times X' \text{VERNAM} \]

\[ C_0 \]

\[ = \]

\[ C_1 \]
Authentication
Authentication

Will you marry me?
Divorce your wife first!
The papers are in the mail...
OK, I will!
Authentication

Will you marry me?

Verification

INVALID

merry me?

merry me?

Authentication

Will you marry me?

merry me?
Authentication

Will you marry me?

No, I never will!

Authentication

Verification

No, I never will!

No, I never

INVALID
Symmetric Authentication

Information Theoretical Security
Symmetric Authentication

\[
(m, t) \nonumber
\]

\[
\text{Authentication} \nonumber \quad t := A_t(m) \nonumber
\]

\[
\text{Verification} \nonumber \quad t = A_t(m) \nonumber
\]
Impersonation

(m, t)

Substitution

(m, t) → (m', t')

Information Theoretical Security
Wegman-Carter
One-Time Authentication
Wegman-Carter
One-Time Authentication
Complexity

Theoretical

Cryptography
Complexity Theory
Complexity Theory

NP
Complexity Theory
Complexity Theory

P ⊆ NP ⊆ BPP
Complexity Theory

Bounded-Probability Polynomial-time

\(\forall x \in L \ Prob[M(x) = \text{accept}] \approx 1\)

\(\forall x \notin L \ Prob[M(x) = \text{accept}] \approx 0\)
Complexity Theory

Bounded-Probability Polynomial-time

∀x ∈ L Prob[M(x) = accept] ≈ 1
∀x ∉ L Prob[M(x) = accept] ≈ 0

Deciding if a number is prime
Deciding if a number is prime

\[\forall x \in L \text{ Prob}[M(x) = \text{accept}] \approx 1\]
\[\forall x \notin L \text{ Prob}[M(x) = \text{accept}] \approx 0\]

Complexity Theory

Bounded-Probability Polynomial-time
Complexity Theory

Decomposing a number into primes

Deciding if a number is prime

Bounded-Probability Polynomial-time

\[ \forall x \in L \ Prob[M(x) = \text{accept}] \approx 1 \]

\[ \forall x \notin L \ Prob[M(x) = \text{accept}] \approx 0 \]
Complexity Theoretical Symmetric Cryptography

Encryption

Authentication
Pseudo-random Bit Generator

RANDOM \ x \ g \ g(x) \ SEEMS RANDOM
Truely Random Bits
Pseudo-random Bits
Stream Cipher from Pseudo-random Bits

pseudo-key \oplus cleartext = ciphertext

pseudo-key \oplus ciphertext = cleartext
The Enigma Machine

German Army Military Enigma: This model was the most widely used version of the German War-Time Enigmas.
Data Encryption Standard

Plaintext

Ciphertext

IP^{-1}

L_0 \quad R_0

F

K_1

L_1 = R_0 \quad R_1 = L_0 \oplus F(R_0, K_1)

F

K_2

L_2 = R_1 \quad R_2 = L_1 \oplus F(R_1, K_2)

\ldots

L_{15} = R_{14} \quad R_5 = L_{14} \oplus F(R_4, K_{15})

F

K_{16}

R_6 = L_{15} \oplus F(R_5, K_{16}) \quad L_{16} = R_{15}

Ciphertext
Figure 7: Propagation of activity pattern (in grey) through a single round
authentication
Authentication from Pseudo-random Bits
Complexity Theoretical Asymmetric Cryptography

- public key distribution
- asymmetric encryption
- asymmetric authentication
Public Key Distribution
Public-Key Distribution

\[ x := f(p, a) \]
\[ y := f(p, b) \]
\[ k := f(y, a) \]
\[ k := f(x, b) \]

\[ f(f(p, a), b) = k = f(f(p, b), a) \]
Diffie-Hellman Key Exchange

**FIGURE 9.2:** The Diffie-Hellman key-exchange protocol.
The Discrete Logarithm and Diffie-Hellman Assumptions

Fix a cyclic group $G$ and a generator $g \in G$.

Given two group elements $h_1, h_2$, define

$$ DH_g(h_1, h_2) \stackrel{\text{def}}{=} g^{\log_g h_1 \cdot \log_g h_2}. $$

That is, if $h_1 = g^x$ and $h_2 = g^y$ then

$$ DH_g(h_1, h_2) = g^{x \cdot y} = h_1^y = h_2^x. $$

• The CDH problem is to compute $DH_g(h_1, h_2)$ given randomly-chosen $h_1$ and $h_2$. 
Public Key Encryption
Asymmetric Encryption
(Public-Key Cryptography)

Complexity Theoretical Security
Public-Key Cryptography

Decryption
Will you marry me?
Yes! T9@

Encryption
8RdewtU5qkLa$es! T9@
RSA Encryption

Ron Rivest, Adi Shamir and Len Adleman
CONSTRUCTION 10.15

Let GenRSA be as in the text. Define a public-key encryption scheme as follows:

- **Gen**: on input $1^n$ run GenRSA($1^n$) to obtain $N, e,$ and $d$. The public key is $\langle N, e \rangle$ and the private key is $\langle N, d \rangle$.

- **Enc**: on input a public key $pk = \langle N, e \rangle$ and a message $m \in \mathbb{Z}_N^*$, compute the ciphertext

$$c := [m^e \mod N].$$

- **Dec**: on input a private key $sk = \langle N, d \rangle$ and a ciphertext $c \in \mathbb{Z}_N^*$, compute the message

$$m := [c^d \mod N].$$

The “textbook RSA” encryption scheme.
The RSA Assumption

(Informal) Given a modulus $N$, an exponent $e > 0$ that is relatively prime to $\varphi(N)$, and an element $y \in \mathbb{Z}^*_N$, compute $e\sqrt{y} \mod N$;

Given $N, e, y$, finding $x$ such that $x^e = y \mod N$ is hard; the success probability of any polynomial-time algorithm is negligible.

However, finding such an $x$ is easy given $p$ and $q$ such that $N = pq$; an exponent $d$ can be easily computed so that $x = e\sqrt{y} \mod N = y^d \mod N$. 
Digital Signatures
Asymmetric Authentication
(Digital Signature Scheme)

Authentication

Verification

Complexity Theoretical Security
Digital Signature

Will you marry me?

Verification

VALID

esIT9@

Authentication

8RdewtU5qkLa$esIT9@

Will you marry me?
Prof. Claude Crépeau

School of Computer Science
McGill University