# Quantum Cyber Security
# Lecture 3: Quantum Key Distribution I

Petros Wallden

University of Edinburgh

20th January 2026

# Outline of Quantum Key Distribution Lectures

- Lecture 3: Motivation and idea of QKD; The first protocol (BB84) and intuition of security
- Lecture 8: Proper Security proof of BB84
- Lecture 9: Other QKD protocols (and quantum money)
- Lecture 10: Device-independent QKD and quantum non-locality

# Outline of Quantum Key Distribution Lectures

- Lecture 3: Motivation and idea of QKD; The first protocol (BB84) and intuition of security
- Lecture 8: Proper Security proof of BB84
- Lecture 9: Other QKD protocols (and quantum money)
- Lecture 10: Device-independent QKD and quantum non-locality

**Reference:** Advances in Quantum Cryptography, Pirandola et al 2019, `https://arxiv.org/abs/1906.01645`

# Cyber Security & Privacy: General

In modern communications there are many essential tasks requiring privacy and security properties guaranteed.

# Cyber Security & Privacy: General

In modern communications there are many essential tasks requiring privacy and security properties guaranteed.

Examples of tasks:

1. **Encryption:** Two parties communicate where no third party can learn anything about the content of the communication
2. **Authentication:** Parties communicate knowing that messages received come from the legitimate party (public messages)
3. **Digital Signatures:** A message with the guarantee of authenticity, integrity and non-repudiation

# Types of Security & the "Quantum Threat"

1. Computational Security: Security guaranteed when adversaries do not have the computational power/time to "break" it

# Types of Security & the "Quantum Threat"

1. Computational Security: Security guaranteed when adversaries do not have the computational power/time to "break" it
   - Frequently relies on assuming that certain **problems** are **hard to solve** (need exponential time)
   - **Security may break** if better (classical) algorithms are found, or new devices (quantum computers), or much faster (classical) computers, or given sufficient time.
   - Security could break **retrospectively** (revealing past secrets)

# Types of Security & the "Quantum Threat"

1. Computational Security: Security guaranteed when adversaries do not have the computational power/time to "break" it
   - Frequently relies on assuming that certain **problems** are **hard to solve** (need exponential time)
   - **Security may break** if better (classical) algorithms are found, or new devices (quantum computers), or much faster (classical) computers, or given sufficient time.
   - Security could break **retrospectively** (revealing past secrets)
2. Information Theoretic Security (ITS): **Cannot be broken irrespective of the computational power** of the adversary ("Perfect Security")

# Types of Security & the "Quantum Threat"

1. Computational Security: Security guaranteed when adversaries do not have the computational power/time to "break" it
   - Frequently relies on assuming that certain **problems** are **hard to solve** (need exponential time)
   - **Security may break** if better (classical) algorithms are found, or new devices (quantum computers), or much faster (classical) computers, or given sufficient time.
   - Security could break **retrospectively** (revealing past secrets)
2. Information Theoretic Security (ITS): **Cannot be broken irrespective of the computational power** of the adversary ("Perfect Security")

Quantum Computers (when scalable) can break computationally secure cryptosystems (RSA, DSA, ECDSA)

# Information Theoretic Secure Encryption: One-Time-Pad

- Message to be sent $x = x_1 x_2 \cdots x_n$ called **plaintext**
- Encrypted message $c = c_1 c_2 \cdots c_n$ called **ciphertext**
- Adversaries learn nothing about $x$ from accessing $c$

# Information Theoretic Secure Encryption: One-Time-Pad

- Message to be sent $x = x_1 x_2 \cdots x_n$ called **plaintext**
- Encrypted message $c = c_1 c_2 \cdots c_n$ called **ciphertext**
- Adversaries learn nothing about $x$ from accessing $c$
- The only (essentially) ITS encryption is the One-Time-Pad:
  1. A secret key $k$ of same size with the plaintext $|x| = |k| = n$
  2. The secret key is known to sender and receiver and no other party has *any* information about it

# Information Theoretic Secure Encryption: One-Time-Pad

- Message to be sent $x = x_1 x_2 \cdots x_n$ called **plaintext**
- Encrypted message $c = c_1 c_2 \cdots c_n$ called **ciphertext**
- Adversaries learn nothing about $x$ from accessing $c$
- The only (essentially) ITS encryption is the One-Time-Pad:
  1. A secret key $k$ of same size with the plaintext $|x| = |k| = n$
  2. The secret key is known to sender and receiver and no other party has *any* information about it
  3. Encryption: Bitwise addition modulo 2 of the plaintext and the secret key: $c = c_1 c_2 \cdots c_n := (x_1 \oplus k_1)(x_2 \oplus k_2) \cdots (x_n \oplus k_n)$
  4. Decryption: Bitwise addition modulo 2 of the ciphertext and the secret key: $(c_1 \oplus k_1)(c_2 \oplus k_2) \cdots (c_n \oplus k_n) =$
     $= (x_1 \oplus k_1 \oplus k_1)(x_2 \oplus k_2 \oplus k_2) \cdots (x_n \oplus k_n \oplus k_n) = x_1 x_2 \cdots x_n = x$

# Information Theoretic Secure Encryption: One-Time-Pad

- Message to be sent $x = x_1 x_2 \cdots x_n$ called **plaintext**
- Encrypted message $c = c_1 c_2 \cdots c_n$ called **ciphertext**
- Adversaries learn nothing about $x$ from accessing $c$
- The only (essentially) ITS encryption is the One-Time-Pad:
  1. A secret key $k$ of same size with the plaintext $|x| = |k| = n$
  2. The secret key is known to sender and receiver and no other party has *any* information about it
  3. Encryption: Bitwise addition modulo 2 of the plaintext and the secret key: $c = c_1 c_2 \cdots c_n := (x_1 \oplus k_1)(x_2 \oplus k_2) \cdots (x_n \oplus k_n)$
  4. Decryption: Bitwise addition modulo 2 of the ciphertext and the secret key: $(c_1 \oplus k_1)(c_2 \oplus k_2) \cdots (c_n \oplus k_n) =$
     $= (x_1 \oplus k_1 \oplus k_1)(x_2 \oplus k_2 \oplus k_2) \cdots (x_n \oplus k_n \oplus k_n) = x_1 x_2 \cdots x_n = x$

     **Example:** $x = 1011$, $k = 0110$

     Encryption: $c = (1 \oplus 0)(0 \oplus 1)(1 \oplus 1)(1 \oplus 0) = 1101$

     Decryption: $(1 \oplus 0)(1 \oplus 1)(0 \oplus 1)(1 \oplus 0) = 1011 = x$

Inf Theor Sec **Encryption**: Large Secret Key (One-Time-Pad)

# The Task: Key Distribution Background

Inf Theor Sec **Encryption**: Large Secret Key (One-Time-Pad)

Shannon's Thm: $|s| \geq |m|$ (key larger than message)

# The Task: Key Distribution Background

Inf Theor Sec **Encryption**: Large Secret Key (One-Time-Pad)

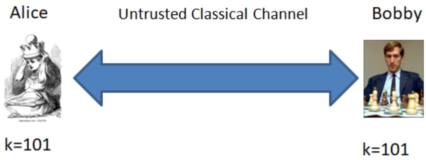
Shannon's Thm: $|s| \geq |m|$ (key larger than message)

Inf Theor Sec **Authentication**: Short Secret Key (Wegman-Carter)

Inf Theor Sec **Encryption**: Large Secret Key (One-Time-Pad)

Shannon's Thm: $|s| \geq |m|$ (key larger than message)

Inf Theor Sec **Authentication**: Short Secret Key (Wegman-Carter)

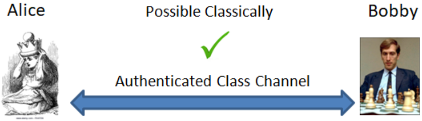


# The Task: Key Distribution Background

Inf Theor Sec **Encryption**: Large Secret Key (One-Time-Pad)

Shannon's Thm: $|s| \geq |m|$ (key larger than message)

Inf Theor Sec **Authentication**: Short Secret Key (Wegman-Carter)



Alice

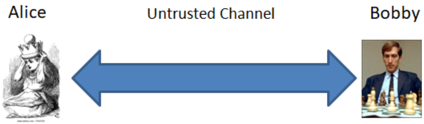
s=110010010

Bobby

s=110010010

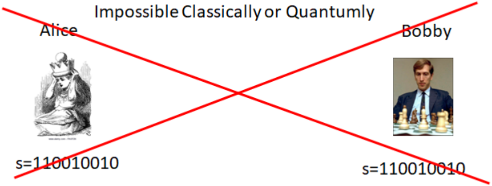
Two spatially separated parties want to share a Large Secret Key

# The Task: Key Distribution Background



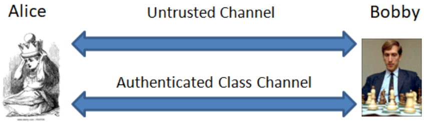

Two spatially separated parties want to share a Large Secret Key

# The Task: Key Distribution Background



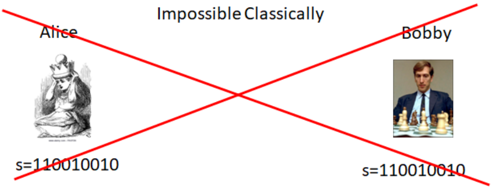

Two spatially separated parties want to share a Large Secret Key

# The Task: Key Distribution Background




Two spatially separated parties want to share a Large Secret Key

# The Task: Key Distribution Background




Two spatially separated parties want to share a Large Secret Key

# The Task: Key Distribution Background

Possible Quantumly

Alice

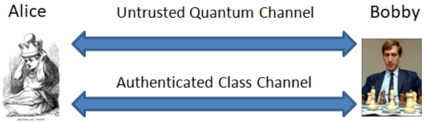
s=110010010

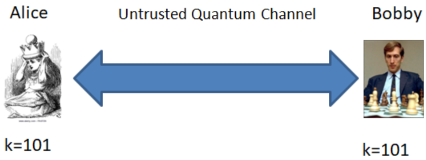
Bobby

s=110010010

Two spatially separated parties want to share a Large Secret Key

# What Quantum Key Distribution Offers






Replace Auth Class Channel with Short Key $k$

# What Quantum Key Distribution Offers


QKD uses untrusted quantum communication and achieves:

Information Theoretic Secure **Secret Key Expansion**

# What Quantum Key Distribution Offers

Possible with QKD

Alice

s=110010010

Bobby

s=110010010

From **Short-Key** sufficient for Inf Theor Sec **Authentication**

Obtain **Long-Key** sufficient for Inf Theor Sec **Encryption**

# Is Happening Now!

QKD is commercially available **currently**

# Is Happening Now!

QKD is commercially available **currently**

Does **not** require a quantum computer

# Is Happening Now!

QKD is commercially available **currently**

Does **not** require a quantum computer

Satellite QKD

# The BB84 Protocol

Bennett and Brassard 1984 first QKD protocol

Followed "quantum money" of Wiesner

# The BB84 Protocol

Bennett and Brassard 1984 first QKD protocol
Followed "quantum money" of Wiesner

Alice

- Sends a string of qubits each from the set $\{|h\rangle, |v\rangle, |+\rangle, |-\rangle\}$
- For each position $(i)$ chooses randomly pair of bits $(a^{(i)}, x^{(i)})$
- $x^{(i)}$ selects the basis: $x^{(i)} = 0 \rightarrow \{|h\rangle, |v\rangle\}$ ; $x^{(i)} = 1 \rightarrow \{|+\rangle, |-\rangle\}$
- $a^{(i)}$ selects state: $a^{(i)} = 0 \rightarrow \{|h\rangle \text{ or } |+\rangle\}$ ; $a^{(i)} = 1 \rightarrow \{|v\rangle \text{ or } |-\rangle\}$
- Stores string of pairs: $(a^{(1)}, x^{(1)}), (a^{(2)}, x^{(2)}), \cdots, (a^{(n)}, x^{(n)})$

# The BB84 Protocol

Bennett and Brassard 1984 first QKD protocol
Followed "quantum money" of Wiesner

Alice

- Sends a string of qubits each from the set $\{|h\rangle, |v\rangle, |+\rangle, |-\rangle\}$
- For each position $(i)$ chooses randomly pair of bits $(a^{(i)}, x^{(i)})$
- $x^{(i)}$ selects the basis: $x^{(i)} = 0 \rightarrow \{|h\rangle, |v\rangle\}$ ; $x^{(i)} = 1 \rightarrow \{|+\rangle, |-\rangle\}$
- $a^{(i)}$ selects state: $a^{(i)} = 0 \rightarrow \{|h\rangle \text{ or } |+\rangle\}$ ; $a^{(i)} = 1 \rightarrow \{|v\rangle \text{ or } |-\rangle\}$
- Stores string of pairs: $(a^{(1)}, x^{(1)}), (a^{(2)}, x^{(2)}), \cdots, (a^{(n)}, x^{(n)})$

Bob

- For each qubit $(i)$ chooses randomly basis $y^{(i)}$ and measures
- Obtains result $b^{(i)}$: $(b^{(1)}, y^{(1)}), (b^{(2)}, y^{(2)}), \cdots, (b^{(n)}, y^{(n)})$

# The BB84 Protocol

Only part that quantum was required!

The correlations between $a^{(i)}$'s and $b^{(i)}$'s and the bound on correlations these bit-strings have with **any** bit-string Eve can produce are **impossible to achieve classically** (see next)

# The BB84 Protocol

Only part that quantum was required!

The correlations between $a^{(i)}$'s and $b^{(i)}$'s and the bound on correlations these bit-strings have with **any** bit-string Eve can produce are **impossible to achieve classically** (see next)

Subsequent Public Communication

- Alice/Bob announce the bases $x^{(i)}, y^{(i)}$ ONLY
  They keep the positions where $x^{(i)} = y^{(i)}$ **raw key**

# The BB84 Protocol

Only part that quantum was required!

The correlations between $a^{(i)}$'s and $b^{(i)}$'s and the bound on correlations these bit-strings have with **any** bit-string Eve can produce are **impossible to achieve classically** (see next)

Subsequent Public Communication

- Alice/Bob announce the bases $x^{(i)}, y^{(i)}$ ONLY
  They keep the positions where $x^{(i)} = y^{(i)}$ **raw key**
- If there is no eavesdropping $a^{(i)} = b^{(i)}$ $\forall$ $i$ of the raw key

# The BB84 Protocol

Only part that quantum was required!

The correlations between $a^{(i)}$'s and $b^{(i)}$'s and the bound on correlations these bit-strings have with **any** bit-string Eve can produce are **impossible to achieve classically** (see next)

Subsequent Public Communication

- Alice/Bob announce the bases $x^{(i)}, y^{(i)}$ ONLY
  They keep the positions where $x^{(i)} = y^{(i)}$ **raw key**
- If there is no eavesdropping $a^{(i)} = b^{(i)}$ $\forall$ $i$ of the raw key
- **Parameter Estimation Phase**
  They choose fraction $f$ of the raw key **randomly** and announce $a^{(i)}, b^{(i)}$ to estimate the correlation of their strings:
  **QBER – Quantum-Bit Error Rate**
  Also can bound the correlation third parties have

Example:

Obtaining the Raw Key

| | | | | | |
|---|---|---|---|---|---|
| Key value $a$ | 0 | 0 | 1 | 1 | 0 |
| Encoding $x$ | 0 | 1 | 1 | 0 | 1 |
| BB84 state sent by Alice | $\|h\rangle$ | $\|+\rangle$ | $\|-\rangle$ | $\|v\rangle$ | $\|+\rangle$ |
| | | | | | |
| Measurement basis $y$ by Bob | 0 | 0 | 1 | 1 | 0 |
| Measurement outcome $b$ | 0 | 1 | 1 | 1 | 1 |
| | | | | | |
| Raw Key | | | | | |

# The BB84 Protocol

Example:

Obtaining the Raw Key

| | | | | | |
|---|---|---|---|---|---|
| Key value $a$ | 0 | 0 | 1 | 1 | 0 |
| Encoding $x$ | 0 | 1 | 1 | 0 | 1 |
| BB84 state sent by Alice | $\|h\rangle$ | $\|+\rangle$ | $\|-\rangle$ | $\|v\rangle$ | $\|+\rangle$ |
| | | | | | |
| Measurement basis $y$ by Bob | 0 | 0 | 1 | 1 | 0 |
| Measurement outcome $b$ | 0 | 1 | 1 | 1 | 1 |
| | | | | | |
| Raw Key | | | | | |

Example:

Obtaining the Raw Key

| | | | | | |
|---|---|---|---|---|---|
| Key value $a$ | 0 | 0 | 1 | 1 | 0 |
| Encoding $x$ | 0 | 1 | 1 | 0 | 1 |
| BB84 state sent by Alice | $\|h\rangle$ | $\|+\rangle$ | $\|-\rangle$ | $\|v\rangle$ | $\|+\rangle$ |
| | | | | | |
| Measurement basis $y$ by Bob | 0 | 0 | 1 | 1 | 0 |
| Measurement outcome $b$ | 0 | 1 | 1 | 1 | 1 |
| | | | | | |
| Raw Key | 0 | × | 1 | × | × |

## Intuition for Security:

- Measurements affect the quantum state – can **detect** amount of **eavesdropping** and abort if high (more than 11% QBER)
- Copying unknown qubits is impossible (No-Cloning Thm)

# Security: Intuition and Attempted Attack

Intuition for Security:

- Measurements affect the quantum state – can **detect** amount of **eavesdropping** and abort if high (more than 11% QBER)

- Copying unknown qubits is impossible (No-Cloning Thm)

  Cannot intercept, copy and resend! **Ideas for attacks?**

# Security: Intuition and Attempted Attack

## Intuition for Security:

- Measurements affect the quantum state – can **detect** amount of **eavesdropping** and abort if high (more than 11% QBER)
- Copying unknown qubits is impossible (No-Cloning Thm)

  Cannot intercept, copy and resend! **Ideas for attacks?**

**Question**

What about intercept, measure and resend?

- We assume that Alice and Bob used same basis $x^{(i)} = y^{(i)}$ (otherwise $(i)$ is not in the raw key)

# Forging attempts: Intercept, measure and resend

- We assume that Alice and Bob used same basis $x^{(i)} = y^{(i)}$ (otherwise $(i)$ is not in the raw key)
- Eve measures in basis $z^{(i)}$
- With probability $p_1 = 1/2$ the basis $x^{(i)} \neq z^{(i)}$ (otherwise no eavesdropping is detected)
- After the measurement, Eve sends the output which is a state from the basis $z^{(i)}$

# Forging attempts: Intercept, measure and resend

- We assume that Alice and Bob used same basis $x^{(i)} = y^{(i)}$ (otherwise $(i)$ is not in the raw key)
- Eve measures in basis $z^{(i)}$
- With probability $p_1 = 1/2$ the basis $x^{(i)} \neq z^{(i)}$ (otherwise no eavesdropping is detected)
- After the measurement, Eve sends the output which is a state from the basis $z^{(i)}$
- Bob measures in the $x^{(i)} \neq z^{(i)}$ basis
- With probability $p_2 = 1/2 = |\langle +| h\rangle|^2$ Bob obtains each of the two outcomes $b^{(i)}$, i.e. with $p_2 = 1/2$ Bob obtains the different outcome from what Alice sent

# Forging attempts: Intercept, measure and resend

- We assume that Alice and Bob used same basis $x^{(i)} = y^{(i)}$ (otherwise $(i)$ is not in the raw key)
- Eve measures in basis $z^{(i)}$
- With probability $p_1 = 1/2$ the basis $x^{(i)} \neq z^{(i)}$ (otherwise no eavesdropping is detected)
- After the measurement, Eve sends the output which is a state from the basis $z^{(i)}$
- Bob measures in the $x^{(i)} \neq z^{(i)}$ basis
- With probability $p_2 = 1/2 = |\langle + | h\rangle|^2$ Bob obtains each of the two outcomes $b^{(i)}$, i.e. with $p_2 = 1/2$ Bob obtains the different outcome from what Alice sent
- Alice and Bob detect 25% **QBER**, i.e. $p_1 \times p_2 = 1/4$

# Full proof and final steps

Full security proof $\Rightarrow$ all possible attacks of Eve

# Full proof and final steps

Full security proof $\Rightarrow$ all possible attacks of Eve

Alice: bit-string $A$; Bob: bit-string $B$
Eve: bit-string $E$ the best guess she can make

# Full proof and final steps

Full security proof $\Rightarrow$ all possible attacks of Eve

Alice: bit-string $A$; Bob: bit-string $B$
Eve: bit-string $E$ the best guess she can make

Can bound correlations of $E$ with $A, B$ given estimated correlation (QBER) of $A, B$ from Parameter Estimation

# Full proof and final steps

Full security proof $\Rightarrow$ all possible attacks of Eve

Alice: bit-string $A$; Bob: bit-string $B$
Eve: bit-string $E$ the best guess she can make

If **QBER** low then $A, B$ more correlated than $A, E$ or $B, E$.

$$H(A:B) > H(A:E)$$

Alice/Bob advantage in the final post-processing:

Final Classical Post-Processing

# Full proof and final steps

Full security proof $\Rightarrow$ all possible attacks of Eve

Alice: bit-string $A$; Bob: bit-string $B$
Eve: bit-string $E$ the best guess she can make

If **QBER** low then $A, B$ more correlated than $A, E$ or $B, E$.

$$H(A:B) > H(A:E)$$

Alice/Bob advantage in the final post-processing:

Final Classical Post-Processing

**Information Reconciliation (IR)**: Exchange information (error-correcting codes) to make $A' = B'$ (extra info leaked to Eve)

# Full proof and final steps

Full security proof $\Rightarrow$ all possible attacks of Eve

Alice: bit-string $A$; Bob: bit-string $B$
Eve: bit-string $E$ the best guess she can make

If **QBER** low then $A, B$ more correlated than $A, E$ or $B, E$.

$$H(A:B) > H(A:E)$$

Alice/Bob advantage in the final post-processing:

Final Classical Post-Processing

**Information Reconciliation (IR)**: Exchange information (error-correcting codes) to make $A' = B'$ (extra info leaked to Eve)

**Privacy Amplification (PA)**: Distil shorter key completely secret from Eve (use universal hash functions to amplify privacy)

# Realistic QKD and post-processing

- Realistic systems have noise: **QBER** $\neq 0$ even if honest
- Cannot tell errors from noise Vs errors from eavesdropping

# Realistic QKD and post-processing

- Realistic systems have noise: **QBER** $\neq 0$ even if honest
- Cannot tell errors from noise Vs errors from eavesdropping
- QBER is used for:
  1. Estimate correlation of Alice's raw bit-string $A$ with Bob's $B$
  2. Bound the max correlation that any adversary's bit string $E$ can have with $A$ (using QM and **specific details of protocol**)

# Realistic QKD and post-processing

- Realistic systems have noise: **QBER** $\neq 0$ even if honest
- Cannot tell errors from noise Vs errors from eavesdropping
- QBER is used for:
  1. Estimate correlation of Alice's raw bit-string $A$ with Bob's $B$
  2. Bound the max correlation that any adversary's bit string $E$ can have with $A$ (using QM and **specific details of protocol**)
- If $(A, B)$ "correlation" is higher than $(A, E)$ then it **is possible** for Alice and Bob to distil an (identical) bit-string $A''$ totally secret from Eve (using IR & PA)
- The key-rate $R$, highest possible noise-tolerance and maximum distance possible all depend on the advantage $H(A:B) - H(A:E)$

## Insights to Remember

- QKD achieves ITS **secret key expansion**
- QKD uses classical authenticated channel
- BB84 requires sending/measuring **single qubits** in **two bases**
- Eavesdropping is detected in **Parameter Estimation Phase**
- If eavesdropping is high (QBER above threshold) we **abort**
- If eavesdropping is low, there is classical algorithm (IR, PA) to generate a **perfectly secret shared key**

# Summary and Demo

## Insights to Remember

- QKD achieves ITS **secret key expansion**
- QKD uses classical authenticated channel
- BB84 requires sending/measuring **single qubits** in **two bases**
- Eavesdropping is detected in **Parameter Estimation Phase**
- If eavesdropping is high (QBER above threshold) we **abort**
- If eavesdropping is low, there is classical algorithm (IR, PA) to generate a **perfectly secret shared key**

## Satellite QKD is real!

https://www.youtube.com/watch?v=YYbp-v4W_yg