A Hands-On Quantum Cryptography Workshop For Pre-University Students

Adrian Nugraha Utama,* Lee Jianwei, and Mathias Alexander Seidler

Centre for Quantum Technologies, National University of Singapore, Block S15, 3 Science Drive 2, 117543

(Dated: June 7, 2020)

Abstract

We developed a modified version of a conventional (BB84) quantum key distribution protocol that can be understood and implemented by students at a pre-university level. We intentionally introduce a subtle but critical simplification to the original protocol, allowing the experiment to be assembled at the skill level appropriate for the students, at the cost of creating a security loophole. The security vulnerability is then exploited by student hackers, allowing the participants to think deeper about the underlying physics that makes the protocol secure in its original form.

I. INTRODUCTION

Quantum cryptography promises information-theoretic security based on quantum mechanics.[1] Secure communication between two parties typically consists of two steps. First, a key is distributed between two parties. Second, parties use the key to encrypt their messages. Classical cryptography [2] relies on either "safely" distributed keys (e.g. via smart cards), or computational complexity (e.g. factorization), and requires trusting the courier, or can be compromised with either increased computational power or with quantum algorithms.[3] However, quantum cryptography does not have these shortcomings, as it provides a framework, based on quantum mechanics, to determine if a would-be adversary has gained enough information to compromise the secrecy of distributed keys.[4,5]

In 1984, Bennett and Brassard invented a scheme[6] to distribute symmetric keys securely between two parties. The security is based on the no-cloning theorem[7]: an eavesdropper cannot create identical copies of an arbitrary unknown quantum state. This quantum key distribution (QKD) protocol is known as the BB84 protocol. In 1991, Artur Ekert introduced another important QKD scheme[8] which utilizes entanglement[9] and a Bell violation measurement[10] to guarantee privacy. These inventions spurred considerable interest from both the computer science and physics communities, and QKD is now an emerging technology.[11–13] Based on counter-intuitive, quantum-physical concepts, QKD also intrigues the general public. There now exist several QKD demonstrations for non-experts which focus on the key distribution step of the protocol.[14–19] However, since QKD and encryption are both non-trivial concepts, a session that implements both steps facilitates learners to arrive at a holistic understanding of how quantum cryptography is used to transmit a private message.

In this work, we present a quantum cryptography workshop developed for pre-university students (age 15 to 19). Our aim is to go from abstract to concrete; from understanding quantum-mechanical concepts, to implementing a working setup in the laboratory. First, the students set up two communication channels: a classical channel based on infrared pulses, and a quantum channel based on polarization encoded photons. Second, they distribute a symmetric key between two parties with the BB84 protocol.[6] Finally, they use the key to encrypt secret messages and send them over to the other party via the classical channel.

The security of BB84 relies on the fact that a single quantum bit (qubit) cannot be copied.[4] When multiple qubits of the same state are distributed, security is compromised since a fraction of the qubits can be intercepted and measured by an adversary. To illustrate this concept, we use macroscopic laser beams instead of single photons, creating a security loophole. We task another group of students to retrieve the key and decode secret messages. This security breach allows the students to revisit the ‘no-cloning’ theorem and its role in quantum cryptography.

In subsequent sections, we describe the BB84 protocol and our modification which enables it to be implemented with commercial, off-the-shelf components. To facilitate educators in conducting the workshop, we have provided detailed schematics and programs used to control the experiment. Our approach received good feedback from the students, who appreciated the hands-on nature of the learning experience, and the unexpected security loophole present in what they presumed to be a secure quantum key distribution system.

II. BBB84 PROTOCOL WITH A TWIST

img/protocol_summ2.pdf

FIG. 1. Summary of the BB84 protocol.

In this section, we describe the BB84 protocol and its security assumption. The protocol generates a symmetric encryption key between two parties, usually called Alice and Bob.

1. Alice’s part of the protocol

Alice generates a sequence of random bits $A = \{a_1, a_2, ..., a_n\}$, where $a_i \in [0, 1]$ is the i-th bit. Next, she generates another random sequence $X = \{x_1, x_2, ..., x_n\}$, where $x_i \in [0, 1]$ denotes the polarization basis used to encode a_i. According to A and X, Alice transmits a sequence of polarized photon pulses to Bob. The encoding scheme is tabulated in Figure 1.

2. Bob's part of the protocol

   Similar to Alice, Bob generates a random sequence of polarization measurement bases $Y = \{y_1, y_2, ..., y_n\}$. He projects the incoming polarization according to Y, and measures the corresponding light intensity. The intensity values are categorized into low and high values and recorded as $B = \{b_1, b_2, ..., b_n\}$, with $b_i \in [0, 1]$.

3. Establishing the final key

   Thereafter, Alice and Bob communicate their basis choices over a classical channel. When $x_i \neq y_i$, b_i does not provide any information about a_i, since the measurement bases are mutually unbiased - these results are discarded. Alice and Bob keep only outcomes when their basis choice agree to generate a shared secret key. In the literature, this procedure is known as "key sifting".

In the original BB84, each bit is encoded by a single photon. Security is ensured by two quantum mechanical concepts. First, a single photon is a quantum state which cannot be cloned. Second, to learn about the state transmitted to Bob, Eve would have to perform a measurement which inadvertently disturbs the state.

We guide students to build a working implementation of the protocol – with a twist, which breaks the security of the protocol. We use laser pulses containing many copies of the same quantum state, which allows for a side-channel attack: An eavesdropper (Eve) can analyse a fraction of the transmitted signal and get full information about the state.

III. EXPERIMENTAL IMPLEMENTATION

We design a workshop for students to gain insight to the inner workings of the protocol by tasking them to construct and operate the modified version of the protocol, described in the previous section. Students establish the quantum channel using polarization-encoded img/schematic-eps-converted-to.pdf

FIG. 2. BB84 setup – quantum channel: Alice encodes a string of bits using different polarization choices set by her rotatable polarizer (RP). The quarter-wave plate (QWP) transforms the linearly polarized light from the laser diode into circular polarization such that the output from Alice's polarizer has equal intensity for all linear polarization choices. Bob projects Alice's photons into different polarization bases, and measures the corresponding intensity with a photodetector (PD). Classical channel: Using infrared transceivers (IR-TR), Alice and Bob communicate the matched bases and the encrypted message.
Side channel attack (SCA): Using a beam splitter (BS), Eve intercepts and measures some of Alice's photons in two different bases simultaneously. As Eve's basis choice is *a priori* not aligned to Alice and Bob's, she may not be able to distinguish between polarization states optimally. However, by measuring in more than one basis choice simultaneously, she improves her ability to identify distinct polarization states even in the presence of laser intensity noise (see main text). She also intercepts the matched bases and encrypted message using her IR receiver from the classical channel.

qubits by controlling laser diode systems, motorized polarization analyzers, and photodetection circuits. They also establish a public classical channel using infrared (IR) pulses by assembling IR transmitting and receiving circuits. A separate group of students eavesdrop on these channels using a beamsplitter and additional IR devices. Figure 2 shows the schematic for the entire setup.

img/setup_stu.png

FIG. 3. BB84 setup as implemented in the workshop. Optical components for the quantum channel were placed and aligned on small optical breadboards on top of short wooden tables – the beam height was below eye level to ensure safety. Students are seated in a diamond configuration enclosing the setup, preventing accidental misalignment. The dashed line indicates the laser beam path.

In the following, we describe the procedure for establishing the quantum and classical channels. The experimental sequence was largely automated as it requires synchronization between different elements, as well as for maintaining consistency in the generation of every bit. The sequence was executed using Arduino and Python programs. These are fully documented and available on Github[20].

A. Quantum Channel

Figure 4 illustrates the quantum channel. Alice prepares different polarization states using laser diode pulses and a motorized polarizer: a quarter-wave plate (QWP) converts linear-polarized light from the laser diode to circular-polarized light, which is subsequently projected into one of the four polarization states (H, V, D, A) using a motorized polarizer. To perform a measurement on the incoming polarization state, Bob chooses one of two measurement bases (H/V or D/A) at random, implements his choice with a motorized img/quant_circuits-eps-converted-to.pdf

FIG. 4. Schematic of the quantum channel. Alice transmits a series of polarized light pulses to Bob – created using a laser diode and a motorized polarizer. Bob projects the incoming states with his motorized polarizer and measures its intensity with a photodetector. The motors and laser diode are operated via the digital output on the microcontroller (DIG pins), while the photodiode readout is recorded by the analog-to-digital converter (ADC) on the microcontroller (A pins). QWP: Quarter-wave plate. GND: Ground connection.

polarizer, and measures the corresponding light intensity.

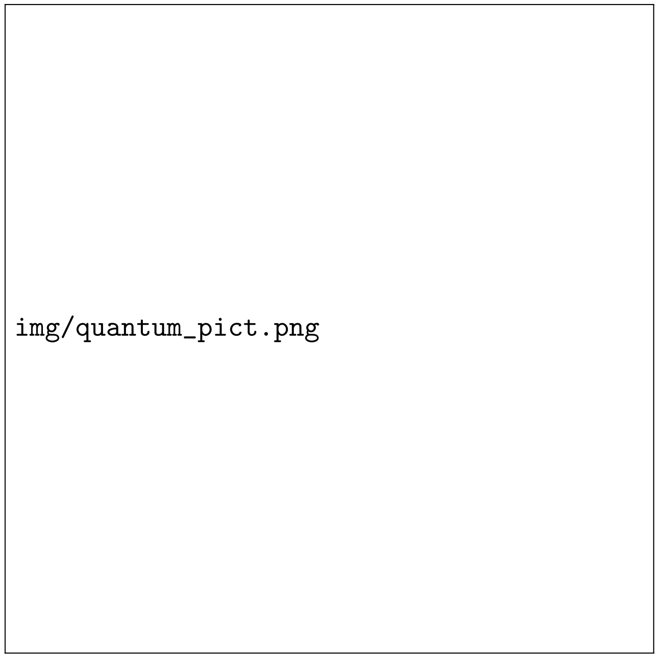

FIG. 5. The quantum setup of Alice (top) and Bob (bottom). The laser beam path is provided as a guidance.

Before using the quantum channel, both parties need to agree on a common coordinate system for their polarization states. This is performed using a visibility measurement. First, Alice chooses an arbitrary rotation angle for her polarizer to fix the orientation of H-polarization. Then, she transmits a macroscopic beam with this polarization to Bob as a reference. To arrive on a common coordinate system with Alice, Bob rotates his polarizer and identifies the position that maximises the transmission of the reference beam with the H-polarization.

Alice begins her portion of the BB84 protocol by using an Arduino microcontroller to produce random binary strings A and X – the strings determine the sequence of polarization states to be transmitted.[21] To generate each polarization qubit, Alice's microcontroller rotates her polarizer to the corresponding angle, and transmits through it a 200 ms-long laser diode pulse. The overall duration required to generate each qubit (1.5 s) is limited by our stepper motors (see Appendix B), which have a maximum rotation speed of about 10 rpm.

In our implementation, Alice transmits a sequence of 16 qubits. She precedes each transmission with a prearranged laser pulse sequence – a header – marking the beginning of each qubit sequence. Due to the clock-stability of the microcontroller ($\sim 10^{-4}$), the uncertainty of the transmission time for each qubit is less than 2.4 ms. The uncertainty is small compared to the duration of each qubit, allowing Bob to predict the arrival times of the qubits after receiving the header pulse sequence. A random binary string Y generated by Bob beforehand determines the bases used for projecting the transmitted qubits. The predictability of the arrival times allows him to rotate his polarizer to the corresponding position between each qubit pulse. For each qubit, the light intensity measured after the polarizer is compared with a predetermined threshold value. This produces two intensity classes: high and low, allowing the measurement to be encoded as a binary number. Measuring the entire sequence generates a binary string B.

B. Classical Channel

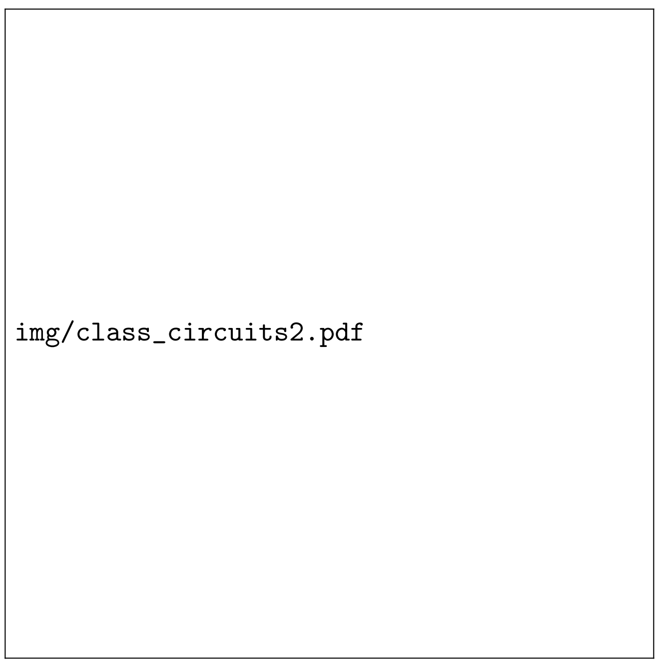

FIG. 6. Schematic of the classical channel. A series of IR pulses enables Alice and Bob to communicate with each other. This enables them to exchange classical information e.g. basis choices, encrypted messages. Information, encoded as binary strings, is translated into pulse sequences. A microcontroller uses the pulse sequences to switch the state of an IR LED to transmit the message. An IR receiver detects the pulses and decodes it with a microcontroller.

The next step of the protocol is for Alice and Bob to communicate their basis choice over a classical channel. In our implementation, the channel is established with IR transmitters

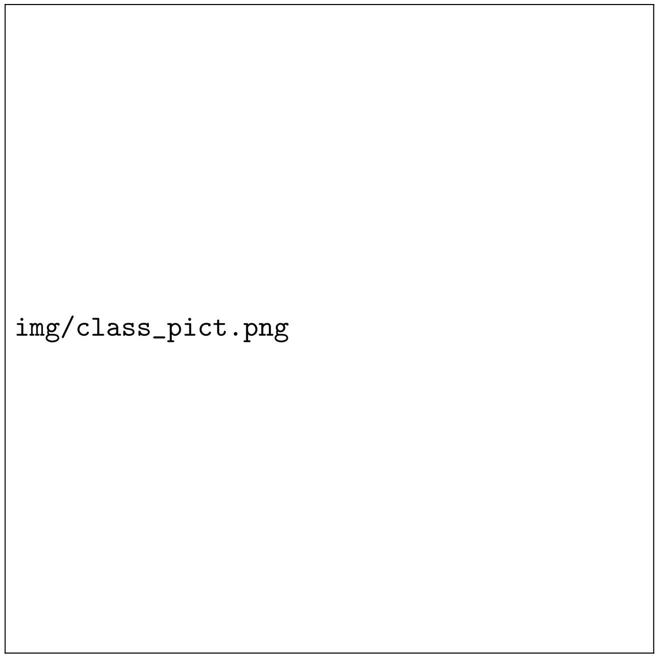

FIG. 7. Top: IR transmitter and receiver circuits are assembled on a solderless electrical breadboard. Bottom: The circles indicate Alice and Bob's IR transceiver circuits, while the arrow indicates the location of Eve's IR receiver.

and receivers (Figure 6). Messages, encoded in a binary string sequence, are translated into IR pulsed sequences using the microcontroller[22].

To generate a key common to both parties using A and B, Alice sends her basis choices X to Bob, who compares it with his basis choices Y. He then transmits the mask $X = Y$ back to Alice. The mask corresponds to them transmitting and measuring in the same basis. Applying the mask on the strings A and B generates the key K. This step concludes the key distribution protocol.

C. Building and applying the encryption key

The length of K is not fixed since the number of events where $X = Y$ is random. For a transmission sequence 16 qubits long, the average length of K is 8-bits.

The key generation procedure can be repeated until the desired length is obtained. For our implementation, we repeated the procedure about four times to obtain a key 32-bit long. This process takes about 2.5 minutes. The key can be used as a one-time pad to encode a message 32-bit long. To encrypt a message M, Alice (or Bob) applies the XOR operation $M \oplus K =: C$, obtaining the encrypted "cipher" C of the original message that is to be transmitted through the classical channel. To decode C and recover M, the operation $C \oplus K$ is applied by the remote party.

Messages longer than 32 bit can still be encoded with a 32 bit key. In our workshop, we used a protocol based on the XOR operation above (see Appendix A). Although this results in insecure encryption, it has the advantage of allowing students to encode longer messages without having to understand the workings of more sophisticated encryption techniques, which might detract from the main point of the workshop.

D. Hacking the key distribution protocol

To eavesdrop on the quantum channel, a third party (Eve) exploits the fact that macroscopic light pulses were used to represent qubits, and directs part of the transmission in the channel into her own polarization analyzing setup (Figure 2) using a beam splitter.

Various factors influence the effectiveness of the eavesdropping attempt. First, the laser intensity fluctuates ($\sim 20\%$) with the position of the polarizer sheet, which is sensitive to mechanical disturbances (e.g. wind, students' movements). Moreover, the current to the laser diode was also not stabilized.

Second, as Eve did not take part in the polarization calibration of Alice's quantum channel, her polarizers are not optimally aligned to distinguish between the various polarization states transmitted in the quantum channel (e.g. Photodiode 1 histogram in Figure 8). To circumvent this issue, Eve projects each state in two measurement basis, each separated by about $45°$ from each other, and subsequently measure the corresponding intensity.

In this way, each polarization state is associated with two intensity values (see Figure 8). In this 2-dimensional space, we use a K-means clustering algorithm to identify four distinct groups that correspond to the polarization states used. Polarization states were arbitrarily assigned to each group and used to generate possible permutations of the key. To proceed, Eve intercepts part of the light emitted from Alice's IR transmitter over the classical channel; the spatial mode of Alice's LED has a relatively large solid angle (see Figure 7). For her first intercept, she obtains the bit positions corresponding to the sifted key (step 3 in Section II) from the classical channel – she uses this information to sift the permuted keys. Next, she intercepts cipher texts over the classical channel. She uses them to identify the correct key permutation that results in a legible decoded message.

img/cluster.pdf

FIG. 8. Four identified clusters of signal voltages measured by Eve's dual photodiodes. Each cluster represents a polarization state intercepted in the quantum channel and are arbitrarily assigned A, B, C, D. Each polarization state represents H, V, D or A sent from Alice to Bob. By assigning the correct polarization state through trial and error, Eve is able to derive the transmitted key. This is done via obtaining intelligible messages by decoding the ciphers transmitted through the classical channel.

IV. THE LEARNING EXPERIENCE

We aspire to create a learning experience where students are exposed to the inner-workings of current technologies with activities resembling hands-on lab experiences. This extension to a typical lab visit utilizes ideas and components already familiar to the students – a platform to introduce the complex systems found in the typical research laboratories.

We trialed this idea at a week-long outreach camp for pre-university students (aged 15-19) in Singapore, where they learn about quantum technologies in a series of lectures and lab visits (QCamp).[23] In the first iteration of the workshop, we tasked the students to custom-build an IR remote controller using the same know-how required for the classical channel (Section III B). As we were encouraged by positive responses from the students and organisers, we decided to extend the workshop into a full BB84 experiment.

A. Planning considerations

We designed the implementation of the workshop to be simple and cost-effective. This is addressed primarily by the use of macroscopic light intensities in the quantum channel with centimeter-sized, commercially available, Si-PIN photodiodes. This removes the requirement to focus light to millimeter-sized single-photon detectors (eg. avalanche photodiodes) usually required for BB84. To further reduce cost, we use standard off-the-shelf optical and mechanical components (Appendix B). Whereas the vulnerability created from using macroscopic light is undesirable in a security context, we found that the failure to realize a qubit as a single quanta provides a powerful learning opportunity for students who successfully hack the system (Section III D) despite following the key distribution procedure faithfully (Section IV D).

Another powerful motivation for simplifying the setup is to allow students the opportunity to build a working QKD system from the ground up, given the time-constraints that could be expected for an extra-curricular activity.

Experimental realization of QKD makes abstract concepts concrete such as the use of polarization encoding in the quantum channel, and basis matching via the classical channel. Furthermore, the hands-on aspect of this approach allows students to demonstrate competence, which is an essential intrinsic motivating factor in any learning task.[24]

Our approach complements previous quantum cryptography demonstrations realized by Lemelle et.al.[14] and Camargo et.al.[15], who also used macroscopic light pulses as a proxy for single photons to encode qubits. However, by incorporating Eve's hacking attempt, our approach provides students with the opportunity to investigate the consequences of using classical resources instead of quantum resources for key distribution.

B. Learning objectives

Before the workshop, the students underwent a series of QCamp lectures covering a spectrum of basic quantum physics and technology discourses. A few lectures serve as the prerequisites to our workshop: photon polarisation, old school cryptography, and quantum key distribution (particularly BB84). The total time for these lectures are about 3 hours.

Although the workshop might appear to require a relatively advanced syllabus, we were able to keep the students focused on the main tasks to build up the cryptographic system. We leave it for the students to explore the more technical side of the protocol implementation on their own[20].

We limited the learning objectives for the students in order to keep the workshop to a time limit of 3 hours:

1. Students should be able to assemble and operate experimental apparatus to distribute encrypted messages with keys obtained from BB84 – refer to Section IV C for task list. By providing a hands-on experience in both creating the secret key and using it to encrypt messages, students will benefit from a more concrete understanding of the cryptography protocol[25].

2. Students should be able to understand the central operating principle of QKD after the hacking attempt – the no-cloning theorem. The apparent failure of the QKD system creates a cognitive conflict which has been shown to result in greater learning gains[26,27].

The workshop ends when Eve successfully decodes the secret message. The debriefing session usually starts with Eve reading out loud the secret message that Alice sends to Bob. This unexpected revelation sparks the discussion of how the setup differs from an ideal BB84 setup – our setup does not use light at the single photon level. We further review the assumptions necessary for BB84, the importance of the no-cloning theorem in quantum cryptography, and discuss how device-independent QKD protocols can alleviate some of the assumptions.[28,29]

C. Student's roles

The students are divided into four teams corresponding to Alice, Bob and Eve (two teams). More students were assigned to Eve as the duration of our workshop limited the tasks that could be accomplished by each team – Eve had to acquire the same knowledge

FIG. 9. Eve is revealed – students are discussing in the debriefing session.

as Alice and Bob, in addition to hacking the protocol. Eve comprised of two teams Eve-classical and Eve-quantum, each focusing on their respective channels. Each team consists of 3-5 members. We assigned a facilitator to each group to help and guide the students throughout the tasks.

The tasks assigned to each team are described below. Teams Alice and Bob were assigned tasks 1-4, team Eve-classical were assigned tasks 1, 5 and 7, and team Eve-quantum were assigned tasks 2, 6, and 7. Students were given three hours to accomplish them.

1. Classical channel. The students build the IR transceiver on the electrical breadboard, and run pre-written programs which allows them to exchange messages back and forth.

2. Quantum channel. The students align the polariser to establish a common polarisation basis, and perform a basic 16-bit BB84 protocol by exchanging a paper (simulating a public channel) back and forth. We have assembled and aligned the optical components prior to the workshop.

3. Encryption key generation. The students run an automated BB84 protocol which utilises both the classical and quantum channel to generate a 32-bit key.

4. Secret message communication. The students use the 32-bit key to encrypt/decrypt

the secret messages with the XOR operation (for 32-bit messages or shorter). Alternatively, they run a program which encrypt/decrypt long messages with a key expansion procedure (see Appendix A). Alice sends the encrypted message through the classical channel.

5. Classical channel hacking. The students position the IR transceiver at a convenient location to eavesdrop on Alice's messages – in particular the matched basis and the encrypted secret messages.

6. Quantum channel hacking. The students record the light pulses split off from Alice's beam using the polarization analyzer, and cluster the polarizations accordingly. A beam-splitter has been placed on Alice's beam path prior to the workshop.

7. Cracking the secret message. The students work together – Eve-classical gives the matched basis to Eve-quantum to generate all the possible key combinations. They use the key combinations on the intercepted cipher text to decode a legible message.

At the end of the workshop, we allocate time for the students to exchange learning experiences with each other, in particular the set of tasks performed by the other teams. The workshop facilitators also explained to the students various optical components and guided them through the whole setup – most importantly, they reveal the location of the beam-splitter and the polarisation analyzer, which was not clandestinely placed yet often overlooked (Figure 3).

D. Students feedback

We conducted the workshops four times during the participants' school holiday period (Jun 2018 and 2019). After each workshop, we surveyed the students. Table I shows the aggregated results.

Overall, students liked the workshop, and were able to follow and apply the concepts thought. Students also highlighted aspects of the workshop that they liked: the successful eavesdropping attempt (31%), the hands-on experience to operate and assemble a functioning QKD system involving lasers and microcontrollers (29%), and the opportunity to apply what they have learnt in lectures on quantum mechanics prior to the workshop (19%).

Students also enjoyed the novelty of the instructional method, and appreciated the worksheets designed to facilitate their understanding of concepts such as key-sifting and K-means clustering (8%).

Around half of the students (54%) highlighted areas for improvement: the absence of time allocated to experience the tasks of adjacent teams (12%), the waiting time for facilitators to troubleshoot the experimental apparatus and debug computer programs (10%), and the level of difficulty of the worksheets and the steep learning curve required to run the programs (10%). A few students thought that the clarity of the instruction could be improved, while some obviously did not enjoy the experience of being hacked.

Questions	Average
How did you like the workshop? 0 (Didn't like it) to 3 (It is great!)	2.67 ± 0.57
How difficult did you find it? -1 (Too easy!) to 1 (Too hard!)	0.04 ± 0.33
Favourite aspects?	Percentage
1. The system being hacked	31%
2. Hands-on experience	29%
3. Learning to apply new concepts	19%
4. Instructions and worksheets	8%
5. Other comments	13%
Areas for improvement?	Percentage
1. Not experiencing other tasks	12%
2. Waiting time when things fail	10%
3. Worksheets and programs	10%
4. Other comments	22%
5. No comment	46%

TABLE I. Feedback from 59 students.

V. CONCLUSION

We have presented a modified BB84 QKD setup aimed at pre-university students. The setup was intentionally modified for ease of implementation while simultaneously providing a deeper insight into how the protocol critically relies on the underlying quantum physics - the no-cloning theorem.

The experiment is implemented with off-the-shelf, commercially available apparatus and can be operated with minimal training in optics, computing and electronics. With the codes and documentations made available online, we believe that this simplified implementation of BB84 is an easily deployable and engaging QKD demonstration that can be realized by a wide range of educational institutions.

A. ENCRYPTION OF LONG MESSAGES

In the main text, we encrypted 32 bit messages with an XOR operation using the key generated from BB84 with the same length. However, for longer messages, we do not repeat the key generation procedure, but instead expand the 32 bit key to the same length as the message. Although this procedure allows the students to encode the message with the XOR operation outlined above,[30] the resulting ciphertext is vulnerable to at least two different attacks.

First, we note that our key expansion procedure uses Mersenne Twister (MT) pseudo-random number generator (PRNG)[31,32]: the 32 bit key is used as an input (seed) to initialize the PRNG, deterministically producing a longer key. Although it has been shown that an adversary is able to guess the rest of expanded key given partial knowledge of the key ("leaks"), we chose to proceed with this PNRG as it provides another learning opportunity pertaining to encryption security – students exploit linear relations between bits to reconstruct the key with high probability[33]. The leaks to the expanded key can, for example, be obtained by examining the structure of the message. For the UTF-8 scheme which is widely used to represent characters in a message, the two leading bits of an English alphabet is 01. This could be used to deduce the leading bits of every byte of the expanded key by sampling the ciphertext. Alternatively, a secure key expansion protocol could be implemented using highly non-linear PRNG protocols.[34,35]

Second, an adversary can try all possible combinations of the 32 bit key, expand them, and decode the ciphertext with the XOR operation. For any message length m, this operation results in only 2^{32} possible combinations of decoded messages. If the message is much longer than the key ($m \gg 32$), it is likely that only a few decoded messages are coherent. This contrasts with the scenario where the message is as long as the key as in the one-time pad scenario ($m = 32$), as this results in as many coherently decoded messages as possible. It is possible for students to perform this brute-force attack with modern computers – if each combination requires $\sim 1\,\mu$s to be computed and checked for coherence, it would only take ~ 1.2 hours to exhaust all possibilities. It is important to note that every encryption protocols are prone to this type of attack, if the key is short enough[36]. The solution to this attack is to create a key long enough that requires an absurd amount of resources to be brute-forced (128 bits, in our current technology).

Although these vulnerabilities were not the main features of our workshop, we used them to highlight potential weak links that could have been overlooked in a presumably secure cryptographic system. We observed that this was an interesting point for a few students, who attempted the brute-force attack by writing their own codes.

B. COST AND COMPONENTS

The total cost of optomechanical, optical, and electronic components for the workshop was approximately USD 2.5k. However, as we were able to borrow most of the standard components from the optics lab and the electronics workshop, the out-of-pocket expenses were barely USD 450, most of which comes from purchasing Arduinos. To reduce the cost, one can consider 3D-printing.

The electrical components for the classical channel consist of a small solderless breadboard (MCBB4000), infrared LED (TSAL6200), infrared receiver (TSOP38238), bipolar NPN transistor (BC547), and various sizes of jumper wires and resistor.

The light source for the quantum channel was built with a reasonably-priced laser diode (Thorlabs L785P5) in a simple transistor switch circuit (see Figure 4). The light output was collimated with a lens (Thorlabs LT220P-B). Temperature and current stabilization were not necessary. The quarter wave plate (Dayoptics WPA215Q) was the most expensive optical component in our list, which cost around USD 100. The motorized polarizers were built from a stepper motor (28BYJ-48-5V) driven with Darlington transistors (ULN2003) and a polarizer sheet (3DLens P50) glued to the motor shaft. As the stepper motor requires higher peak current than what an Arduino supplies, we require an external power supply. The photodetection circuit used a PIN silicon photodiode (OP906) in a reverse-biased configuration. A variable resistor was used to modify the conversion gain of the photodiode. The beam was aligned with high-reflective mirrors on kinematic mirror mounts (KM100).

ACKNOWLEDGMENTS

We would like to thank Prof. Christian Kurtsiefer for his support and guidance for this project, and for his valuable inputs on the manuscript. We appreciate the help and support from Jenny Hogan and the steering committee of QCamp 2018 and 2019. We would like to extend our gratitude to all the students that took part in this experiment. Last but not least, we gratefully acknowledge the help, effort, and perseverance from the following facilitators: Chow Chang Hoong, Tan Ting You, Yeo Xi Jie, Chen Jia Pern Neville, Florentin Thierry Adam and Ng Boon Loong.

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore and the Ministry of Education, Singapore under the Research Centres of Excellence programme.

---

* adrian_utama@u.nus.edu

[1] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, "Quantum cryptography," Rev. Mod. Phys. **74**, 145 (2002).

[2] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Communications of the ACM, **21**(2), 120-126 (1978).

[3] M. A. Nielsen, and I. Chuang, Quantum computation and quantum information (2002).

[4] P. W. Shor, and J. Preskill, "Simple Proof of Security of the BB84 Quantum Key Distribution Protocol," Phys. Rev. Lett. **85**, 441 (2000).

[5] R. Renner, N. Gisin, and B. Kraus, "Information-theoretic security proof for quantum-key-distribution protocols," Phys. Rev. A **72**, 012332 (2005).

[6] C. H. Bennett, and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," Theor. Comput. Sci. **560**, 7–11 (2014).

[7] W. K. Wootters, and W. H. Zurek. "A single quantum cannot be cloned," Nature **299** (5886), 802-803 (1982).

[8] Artur K. Ekert, "Quantum cryptography based on Bell's theorem," Phys. Rev. Lett. **67**, 661 (1991).

[9] R. Horodecki, P. Horodecki, M. Horodecki, and K. Horodecki, "Quantum entanglement," Rev. Mod. Phys. **81**, 865 (2009).

[10] John S. Bell, "On the einstein podolsky rosen paradox," Physics Physique Fizika **1**(3): 195 (1964).

[11] T. Schmitt-Manderbach et.al., "Experimental Demonstration of Free-Space Decoy-State Quantum Key Distribution over 144 km," Phys. Rev. Lett. **98**, 010504 (2007).

[12] C. Z. Peng et. al., "Experimental Long-Distance Decoy-State Quantum Key Distribution Based on Polarization Encoding," Phys. Rev. Lett. **98**, 010505 (2007).

[13] J. Yin et. al., "Satellite-based entanglement distribution over 1200 kilometers," Science **356**(6343), 1140-1144 (2017).

[14] D. S. Lemelle, M. P. Almeida, P. H. Souto Ribeiro, and S. P. Walborn, "A simple optical demonstration of quantum cryptography using transverse position and momentum variables," Am. J. Phys. **74**, 542–546 (2006).

[15] A. L. P. Camargo, L. O. Pereira, W. F. Balthazar, and J. A. O. Huguenin, "Simulation of the BB84 protocol of quantum cryptography by using an intense laser beam," Revista Brasileira de Ensino de Física **39**, 2305 (2017).

[16] Karl Svozil, "Staging quantum cryptography with chocolate balls," Am. J. Phys. **74**, 800–803 (2006).

[17] A. López-Incera, and W. Dür, " Entangle me! A game to demonstrate the principles of quantum mechanics," Am. J. Phys. **87**(2), 95–101 (2019).

[18] A. Kohnle, and A. Rizzoli, "Interactive simulations for quantum key distribution," Eur. J. Phys. **38**, 035403 (2017).

[19] S. DeVore, and S. Chandralekha, "An Interactive Learning Tutorial on Quantum Key Distribution," arXiv:1601.01301 (2016).

[20] The codes and technical documentation for this project is available at ⟨`https://github.com/`

HelpMeFinishPhD/Qcamp2019⟩

21 We generate random numbers using the Entropy library, available at ⟨https://sites.google.com/site/astudyofentropy⟩.

22 Arduino IR Remote Library, available at ⟨https://z3t0.github.io/Arduino-IRremote/⟩

23 QCamp, hosted by Centre for Quantum Technologies. Visit us at ⟨https://qcamp.quantumlah.org⟩

24 R. M. Ryan, and E. L. Deci, "Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions," Contemporary Educational Psychology **25**, 54–67 (2000).

25 D. L. Haury, and P. Rillero, "Perspectives of Hands-On Science Teaching," (1994).

26 B. J. Guzzetti, T. E. Snyder, G. V. Glass, and W. S. Gamas, "Promoting conceptual change in science: A comparative meta-analysis of instructional interventions from reading education and science education," Reading Research Quarterly, 117-159 (1993).

27 M. Limón, "On the cognitive conflict as an instructional strategy for conceptual change: A critical appraisal," Learning and instruction, **11**(4-5), 357-380 (2001).

28 A. Acin, N. Gisin, and L. Masanes, "'From Bell's theorem to secure quantum key distribution," Phys. Rev. Lett. **97**, 120405 (2006).

29 U. Vazirani, and T. Vidick, "Fully Device-Independent Quantum Key Distribution," Phys. Rev. Lett. **113**, 140501 (2016).

30 M. J. Robshaw, "Stream ciphers," RSA Laboratories Technical Report (1995).

31 M. Matsumoto, and T. Nishimura, "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator," ACM Transactions on Modeling and Computer Simulation (TOMACS), 8(1), 3-30 (1998).

32 Python implementation of Mersenne Twister PRNG, available at ⟨https://docs.python.org/3/library/random.html⟩

33 G. Argyros and A. Kiayias, "I Forgot Your Password: Randomness Attacks Against PHP Applications," Presented as part of the 21st USENIX Security Symposium (USENIX Security 12) (2012).

34 Makoto Matsumoto, et al, "CryptMT stream cipher version 3." eSTREAM, ECRYPT Stream Cipher Project, Report 28 (2007).

35 T, Good, and M. Benaissa, "Hardware results for selected stream cipher candidates," State of the art of stream ciphers **7**, 191-204 (2007).

[36] C. Paar, and J. Pelzl, "Understanding cryptography: a textbook for students and practitioners," Springer Science & Business Media (2009).