Quantum Algorithms 4/20/2023

Lecture 24: Cryptography

Instructor: Dieter van Melkebeek

This lecture is an introduction to quantum cryptography. We start with some background on classical cryptography and present two widely used classical systems whose security is compromised by quantum computing: the RSA system by the quantum algorithm for factoring integers and the Diffie-Hellman system by the quantum algorithm for the discrete log problem. Most of the lecture deals with a quantum system for secret key exchange, which achieves an information-theoretic level of security that is provably impossible to achieve in the classical setting.

1 Crypto background

Cryptography is the study of secure interaction between two or more parties through channels that may be accessible to others. There are several common interpretations for "secure."

- Confidentiality: ensuring that only the intended parties can extract the content of the messages.
- Integrity: ensuring that tampering with the content of the messages is detected.
- Authenticity: ensuring that the interaction is between the intended parties.

Providing confidentiality between two parties can be modeled as follows. Suppose that Alice wants to send a message M to Bob. An eavesdropper, Eve, tries to figure out what Alice has sent to Bob by intercepting the communication between them. Therefore, to secure the message against Eve, Alice has to massage M into something else before sending it to Bob. After receiving the message, Bob recovers the original content of M by some operation. In order for this scheme to work, either Alice or Bob (or both) must know something more than Eve does; otherwise, Eve could simulate Bob's actions to recover M.

Formally, Alice runs an encryption algorithm E with some encryption key K_E on the plaintext M. The algorithm outputs a ciphertext C, which is sent to Bob over the channel. At Bob's side, a decryption algorithm D with decryption key K_D is run on the ciphertext C. We expect the decryption algorithm to output M. We need to pick E, D, K_E and K_D such that Eve is unable to recover M by just looking at C.

There are two different settings. The first one is *private key systems*, also known as symmetric systems, where $K_E = K_D$. In such systems the encryption and decryption keys need to be kept private. The second one is *public key systems*, also known as asymmetric systems), where K_E can be easily computed from K_D but not the other way around. In such systems the encryption key is made public and only the decryption key needs to be kept private.

Private key systems. We need to specify how to encrypt and decrypt a message $M \in \{0,1\}^n$ using a single key K. The encryption algorithm outputs $C = E_K(M) = M \oplus K$. The decryption algorithm outputs $D_K(C) = C \oplus K = (M \oplus K) \oplus K = M$. In this system, not only are the encryption and decryption keys the same (as is the case for all private key systems), but the encryption and

decryption algorithms also happen to be the same. We pick K uniform at random from $\{0,1\}^n$. No matter how much computational power the eavesdropper has, she cannot correctly determine the plaintext M with a probability higher than 2^{-n}. Information theoretically, this is the best we can hope for. In other words, the one-time pad system has perfect security.

One drawback to the one-pad time system is that the key needs to have the same length as the message. If we want to send a message of size n, we have to generate n random bits. We can relax the notion of security from information-theoretic to complexity-theoretic. Instead of requiring the system to be secure against eavesdroppers with unlimited computational resources, we only require it to be secure against computationally bounded adversaries, e.g., polynomial-time machines. We achieve this by using pseudorandom bit generators (PRG) with short seed length that fool computationally bounded adversaries. With a such PRG, we can first pick a short random string and then apply the PRG to get a long pseudorandom string K which is used as the key. The existence of such PRGs is known to be equivalent to the existence of one-way functions, which are (length-preserving) functions that are efficiently computable but not efficiently invertible. Such functions can only exist if P differs from NP. It remains an open question whether P $\neq$ NP is equivalent to their existence.

Other examples of private key systems include the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES). These systems have been shown to be secure under some specific complexity-theoretic hypotheses.

Public key systems. In this setting, Bob picks K_D at random and out of it computes K_E. Then he makes K_E public to everyone he expects to receive messages from, while keeping K_D private. Now, if Alice wants to send Bob a message, she has to encrypt the message with the *public key* K_E and send the ciphertext over the channel. After receiving the ciphertext, Bob uses his *private key* K_D to decrypt it back into plaintext message. A widely used public key system is the RSA encryption system, which we will discuss later in this lecture.

One advantage of public key systems over private key systems is that when m parties want to communicate with each other, only m pairs of keys are needed using a public key system, whereas $O(m^2)$ are needed in a private key system. Nevertheless, many private key systems are still being used. One reason is that most public key systems are more computationally intensive than private key systems. Moreover, known classical algorithms for breaking public key systems are more efficient than those for private key systems. As a result, the key size of a public key system needs to be larger than that of a private key system, which amplifies the higher computational complexity. A combination used in practice is to run a public key system to set up a private key, which is subsequently used for the actual communication.

2 Quantum vs. crypto

Quantum computing offers both challenges and opportunities for cryptography.

Challenges. The quantum algorithms for integer factorization and the hidden subgroup problem (HSP) on abelian groups threaten all cryptosystems that rely on the hardness of integer factorization and discrete log. Algorithms relying on elliptic curves are also compromised as the underlying group formed by points on an elliptic curve is an abelian group.

An important challenge is constructing a new classical cryptosystem that is resilient against quantum computers. One option is lattice-based cryptosystems. These systems presume the hardness of the shortest vector problem: given a basis of integer vectors in $\mathbb{R}^n$, find an integer linear combination of that basis that results in the shortest vector. This problem can be reduced to solving HSP on dihedral groups, which are non-abelian. Currently, no efficient quantum algorithms for this instance of HSP are known and these systems are conjectured to be safe against quantum computers.

As a side note, we mention that none of the problems that have been used to build cryptosystems are known to be NP-hard. This includes the lattice-based systems; whereas the shortest vector problem is NP-hard in general, the subproblem of instances used is not known to be NP-hard. It is a major open question to build classical cryptographic systems based on NP-hard problems, for which breaking the system classically would imply P = NP. Such a system would arguably provide a significantly higher degree of confidence in their (classical) security. This is because of the pervasiveness of NP-completeness, many people from very different backgrounds have (implicitly) tried to break such a system. Since NP-hard problems are conjectured to be infeasible on quantum computers, such systems are expected to remain secure against quantum adversaries, as well.

Opportunities. One can try to exploit the special properties of quantum information to build efficient cryptosystems that offer information-theoretical security where this is classically provably impossible. An early success was achieved for secure key exchange, which we discuss in the remainder of this lecture.

There are also classical cryptosystems that have been shown to remain secure against quantum computers. An example are the zero knowledge proofs discussed in the next lecture. Such results are of interest even in an era where quantum computers are not (widely) available.

3 RSA cryptosystem

RSA is a public-key cryptosystem for exchanging confidential messages. The security of the protocol hinges on the hardness of factoring integers. The acronym is composed of the first letter of each of the three computer scientists who developed the protocol, Rivest, Shamir, and Adleman.

The private key is composed of two randomly chosen distinct primes p and q, and an integer $d \in \mathbb{N}$ relatively prime to $(p-1)$ and $(q-1)$. By the prime number theorem, picking an n-bit number at random yields a prime with probability $\Omega(1/n)$. Combined with known polynomial-time algorithms for primality test and for testing relative primality (the Euclidean algorithm), this allows us to generate the private key efficiently.

The public key is the product $\mu = p \cdot q$, and $e \in \mathbb{N}$ such that $de \equiv 1 \bmod (p-1)(q-1)$. In other words, e is the modular inverse of d, which exists since d is relatively prime to $(p-1)(q-1)$ and can be found efficiently using the extended Euclidean algorithm.

Encryption Let $M \in \mathbb{Z}_\mu$ be the confidential message. Alice computes the encrypted message $E = M^e \bmod \mu$, using Bob's public key e, and sends E to Bob.

Decryption Bob receives the encrypted message E and calculates $D = E^d \bmod \mu$.

Proposition 1. *The decryption step recovers the original message, i.e., $D = M$.*

Proof. We need to show that for all $M \in \mathbb{Z}_\mu$ and an integer n, $M^{ed} \equiv M^{n(p-1)(q-1)+1} \equiv M \bmod \mu$. To prove this, recall Fermat's little theorem, which states that $a^{p-1} \equiv 1 \bmod p$ for prime p and a satisfying $\gcd(a, p) = 1$. Therefore, for any integer k, $a^{k(p-1)} \equiv 1 \bmod p$ when $\gcd(a, p) = 1$. Going further,

$$a^{k(p-1)+1} \equiv a \pmod{p},$$

and this is even true when $a \equiv 0 \bmod p$.

Using this fact, we can deduce that $M^{n(p-1)(q-1)+1} \equiv M \bmod p$ and $\bmod q$, by using $k = n(q-1)$ and $n(p-1)$ respectively. Therefore,

$$M^{n(p-1)(q-1)+1} \equiv M \bmod \mu$$

by the Chinese remainder theorem. □

The security of this protocol depends on preventing the attacker from gaining d from e given μ. If the attacker can factor an integer efficiently, then she can find p and q from μ, which allows her to know $(p-1)(q-1)$. Because calculating a modular inverse is efficient, the attacker can produce d from e just like Bob can.

Efficient integer factorization would allow an attacker to execute the steps described above. Classically, it is an open question as to whether RSA is secure, as no efficient classical integer factorization algorithm is known but we cannot rule out its existence either. In addition, it is unknown if breaking RSA and integer factorization are equivalent, so there is the possibility that RSA can be broken without requiring efficient integer factorization. A system for confidential communication whose security is known to be equivalent to integer factorization is the Rabin cryptosystem.

4 Diffie-Hellman key exchange

The Diffie-Hellman system (DH) allows Alice and Bob to agree on a key chosen uniformly at random that only they know. The key can subsequently be used for a symmetric cryptosystem, for example. DH is a public-key system that hinges on the difficulty of the discrete log problem.

In DH, there are two public parameters: a prime p and a generator g for $\mathbb{Z}_p^\times, \cdot$. We already explained how p can be generated efficiently. There are classical efficient ways to generate g. If we are willing to use the power of quantum computing, in particular the efficient quantum for factoring integers, the following elementary way suffices. The fact that $\mathbb{Z}_p^\times, \cdot$ is cyclic, that any fixed generator g raised to a power relatively prime to $p-1$ is also a generator, and the prime number theorem imply that picking g at random in $\mathbb{Z}_p$ yields a generator with probability $\Omega(1/n)$, where n is the bit-length of p. We can check whether g is a generator by making sure it is relatively prime to $p-1$, factoring $p-1$, and verifying that for every prime divisor q of $p-1$, $g^{(p-1)/q} \neq 1 \bmod p-1$ (using repeated squaring). The expected time to find a generator is polynomial in the bit-length of p.

When Alice and Bob want to create a shared secret key, they use the following protocol.

1. Alice picks $a \in \mathbb{Z}_p^\times$ uniformly at random, and sends $A \equiv g^a \bmod p$ to Bob. Bob picks $b \in \mathbb{Z}_p^\times$ uniformly at random, and sends $B \equiv g^b \bmod p$ to Alice.

2. Alice computes $K_A \equiv B^a \bmod p$ and Bob computes $K_B \equiv A^b \bmod p$. Since $K_A \equiv K_B \equiv g^{ab} \bmod p$, Alice and Bob can use K_A and K_B as the shared key.

Note that if x is uniformly distributed in $\mathbb{Z}_p^\times$, then g^x is also uniformly distributed in $\mathbb{Z}_p^\times$, because the set $\{g, g^2, \ldots, g^{p-1}\}$ is a permutation of $\{1, 2, \ldots, p-1\}$. This prevents any easy brute force attack.

DH is vulnerable to any attacker who can efficiently calculate discrete logs. First, a and b can be extracted using discrete log from the intercepted messages A and B. From this, the attacker can compute ab and then the shared key $g^{ab} \bmod p$. In fact, only one of a or b is necessary, as the attacker can choose either A^b or $B^a \bmod p$ to compute the secret key.

The hardness of DH relies on a problem that is possibly easier than discrete log: given g^a and $g^b \bmod p$, compute $g^{ab} \bmod p$. It is unknown if breaking DH is equivalent to solving the full discrete log problem efficiently. Either way, due to the efficient quantum algorithm for the discrete log problem, DH is not secure in the quantum setting.

5 Quantum key exchange

We now switch to an exciting opportunity that quantum computing offers in cryptography, namely information-theoretically secure secure key generation. It allows Alice and Bob to agree on a random key that is essentially uncorrelated to Eve's state. To do so, Alice and Bob use a quantum channel that Eve can tamper with. They cannot prevent Eve from tampering with the channel, but they will detect if the distribution deviates in a non-negligibly from the uniform one and/or when Eve's state is non-negligibly entangled with the key. One caveat: In addition to the quantum channel that Eve can tamper with, all known protocols require a classical channel that Eve can watch but cannot tamper with. Even with this caveat, the classical counterpart of such protocols provably do not exist.

Quantum key exchange represents one of the early applications of quantum computing, and a number of protocols have been developed over time. The historically first such protocol is known as BB84, referring to its inventors Bennett and Brassard, and the year 1984 of the publication [1]. It makes use of EPR pairs that are split between Alice and Bob, where Alice generates them, keeps her halves, and sends the remaining halves over the quantum channel to Bob. Other than that, the protocol only requires very simple qubit operations, namely individual rotations, and has been implemented in practice. However, the proof of security is involved. In fact, it took more than 15 years until the first correct proof was found [3]. For that reason, we will describe the protocol but not argue its security.

We will cover another protocol in full, i.e., including its proof of security, namely to one due to Lo and Chau [2]. The proof of security is simpler, but the protocol itself is less practical in the current state of quantum computing due to the use of many CNOTs. We first present Lo and Chau, and the Bennett and Brassard.

6 Lo & Chau

The starting point is the following quantum version of the one-time pad: Alice and Bob jointly prepare n EPR pairs, take half of each pair, and separate. After they arrive at their remote locations and want to use a secret common key, they measure their halves of the EPR pairs. They are guaranteed to measure the same random n-bit string. Just like for the one-time pad, the problem is that this requires Alice and Bob to have been at the same place at some point in the past and have generated all the EPR pairs they would need in the future. In addition, the use of

EPR pairs seems like an overkill; Alice and Bob could just as well have picked a common random bit string, as happens in the one-time pad.

The protocol by Lo and Chau obviates the need for Alice and Bob to have jointly generated the EPR pairs. Instead, Alice generates the pairs on her own, keeps her halves, and sends the other halves to Bob over a quantum channel that Eve can tamper with. The protocol then runs some tests that exploit the quantum entanglement and, by sacrificing some of the pairs, ensures with high probability that the key obtained by measuring the remaining pairs is chosen close to uniformly at random and has negligible entanglement with Eve's state.

In order to develop and analyze the tests, it helps to express the state of the EPR pairs in the Bell basis.

6.1 Bell basis

The Bell basis is an orthonormal basis for 2-qubit systems that includes the EPR pair. It consists of:

$$
\begin{aligned}
|\Phi^+\rangle &= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \\
|\Phi^-\rangle &= \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle), \\
|\Psi^+\rangle &= \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), \\
|\Psi^-\rangle &= \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).
\end{aligned}
$$

Note that all of the basis elements are invariant under swapping the two components, modulo a phase flip for $|\Psi^-\rangle$.

One way to obtain the basis is by applying the following unitary operator to the standard basis: First apply the Hadamard gate to the first qubit, and then a CNOT with the first qubit as the control and the second qubit as the target. This leads to the following alternate notation, where $x, y \in \{0, 1\}$:

$$|\Psi_{xy}\rangle \doteq \mathrm{CNOT} \circ (H \otimes I)\,|xy\rangle\,.$$

The correlation between the two notations is given in the following table:

$\lvert\Psi_{xy}\rangle$	$x=0$	$x=1$
$y=0$	$\lvert\Phi^+\rangle$	$\lvert\Phi^-\rangle$
$y=1$	$\lvert\Psi^+\rangle$	$\lvert\Psi^-\rangle$

Observe the double use of the letter Ψ. In the $|\Psi_{xy}\rangle$ notation, the bit x describes the sign, where 0 denotes that the two contributing standard basis vectors have the same sign, and 1 that they have different signs. The bit y denotes the type, where 0 indicates that the two contributing standard basis vectors each consist of identical bits (Φ type), and 1 that they consist of complementary bits (Ψ type).

This way of generating the Bell basis is captured by the leftmost circuit in Figure 1. For future reference, it is useful to start from the fixed basis element $|00\rangle$ and perform the operations that depend on x and y at the end. The first property can be realized by introducing bit flips depending on x and y as in the second circuit in Figure 1, where X^x denotes the identity when $x = 0$, and X

when $x = 1$. Note that the operator X^y on the second qubit can be postponed till after the CNOT without affecting the result. This is because powers of X commute and because operations on the second qubit have no effect on the state of the controlling qubit of the CNOT. As for the operator X^x on the first qubit, recall that X and Z are conjugates through the Hadamard operator H. In particular, $HX^x = Z^xH$. Combined, this gives us the third circuit in Figure 1. Finally, note that the operator Z^x in the third circuit can be postponed till after the CNOT without affecting the result. This is because the controlling qubit is not affected by the CNOT. We end up with the rightmost circuit in Figure 1 for generating $|\Psi_{xy}\rangle$.

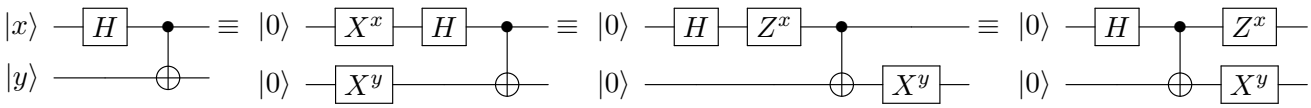

Figure 1: Generating circuits for Bell state $|\Psi_{xy}\rangle$

Apart from including the EPR pair, other useful properties of the Bell basis are closure up to a global phases under some elementary operations, namely the Hadamard transform $H^{\otimes 2}$ and "owner-wise" CNOT on two pairs. Whereas, the result of a unitary operator on a basis vector can generally be written as a linear combination of all basis vectors, in this case only one basis vector is needed, which simplifies further analysis.

Hadamard transform. What happens if both Alice and Bob perform an Hadamard transform on the qubit that they own? Applying $H^{\otimes 2}$ to $|00\rangle$ yields the uniform superposition. Applying $H^{\otimes 2}$ to $|11\rangle$ also yields an equal-weight real superposition of all standard basis vectors, but now the sign of the non-homogeneous ones, $|01\rangle$ and $|10\rangle$, is flipped. Thus, by linearity, $H^{\otimes 2}|\Phi^+\rangle = |\Phi^+\rangle$ and $H^{\otimes 2}|\Phi^-\rangle = |\Psi^+\rangle$. That is, $|\Phi^+\rangle$ is invariant, and $|\Phi + -\rangle$ changes type and sign.

More generally, Figure 2 shows that $H^{\otimes 2}|\Psi_{xy}\rangle = (-1)^{xy}|\Psi_{yx}\rangle$, i.e., each of Bell states is mapped the Bell state with the sign and type swapped, and there is a global phase change for $|\Psi^-\rangle = |\Psi_{11}\rangle$. The first step in Figure 2 follows from the conjugation of X and Z by H. The second step follows because conjugation of CNOT by H yields a CNOT with the control and the target swapped, and the fact that $H^2 = I$. Note that the right-hand side of Figure 2 is the same as the right-hand side of Figure 1 after swapping the two qubits and swapping x and y. The conclusion holds because the Bell states are invariant under swapping the qubits except that $|\Psi^-\rangle = |\Psi_{11}\rangle$ receives a global phase change, which is captured by the factor $(-1)^{xy}$.

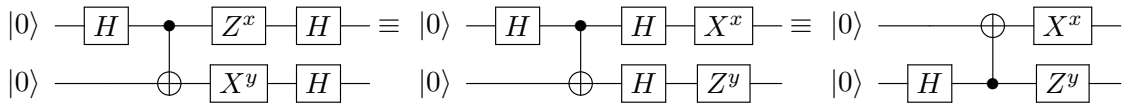

Figure 2: Effect of $H^{\otimes 2}$ on Bell state $|\Psi_{xy}\rangle$

Owner-wise CNOT. Consider a situation with two pairs, say in state $|\Psi_{x_1y_1}\rangle|\Psi_{x_2y_2}\rangle$, and both Alice and Bob apply a CNOT to their halves, where the first pair acts as the control and the second

one as the target. If the controlling qubits are in state $|00\rangle$, there is no effect on the controlled pair. If the controlling qubits are in state $|11\rangle$, then both qubits of the controlled pair are complemented. By linearity, this means that owner-wise CNOT has no effect on $|\Phi^+\rangle|\Phi^+\rangle$. When considering $|\Phi^+\rangle|\Phi^-\rangle$, the component $|11\rangle$ of the controlling pair induces a global phase of -1 on $|\Phi^-\rangle$, which can be kicked back to the $|11\rangle$ component of the controlling pair while leaving the controlled pair unaffected. The net result is a transformation of $|\Phi^+\rangle|\Phi^-\rangle$ into to $|\Phi^-\rangle|\Phi^-\rangle$.

More generally, Figure 3 shows how owner-wise CNOT transforms $|\Psi_{x_1y_1}\rangle|\Psi_{x_2y_2}\rangle$ into $|\Psi_{x'_1y'_1}\rangle|\Psi_{x'_2y'_2}\rangle$, where the new signs and types are given by the following table:

	$i=1$	$i=2$
x'_i	x_1+x_2	x_2
y'_i	y_1	y_1+y_2

Note that the type of the controlling pair does not change, and the type of the controlled pair becomes the XOR of the types of the two pairs. The relationships for the sign is the same but with the roles of the controlling and controlled pairs swapped.

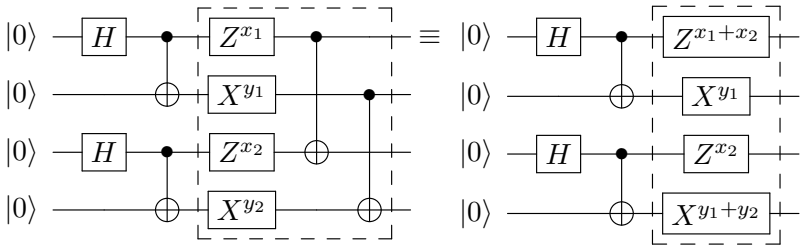

Figure 3: Effect of owner-wise CNOT on Bell product states $|\Psi_{x_1y_1}\rangle|\Psi_{x_2y_2}\rangle$

To justify the transformation in Figure 3, we argue the equivalence of the boxed circuits on the left and the right. First consider the effect on Bob's qubit. The CNOT has no effect on the controlling qubit. The effect of the CNOT on the controlled qubit can be written as X^{y_1}: there is a bit flip if $y_1 = 1$, and no change, otherwise. Thus, the overall effect is equivalent to $X^{y_1}X_{y_2} = X^{y_1+y_2}$. The effect on Alice's qubits follow from a similar reasoning combined with the conjugation properties under H. Conjugating the boxed circuit on the left yields the leftmost circuit in Figure4. The property that $HZ = XH$ yields the circuit in the middle of Figure4, and the conjugation property of CNOT the right-most circuit of Figure 4. Conjugation the right-most circuit of Figure 4 and once more the property that $HXH = Z$ then yields Alice's part in the boxed circuit on the right of Figure 3.

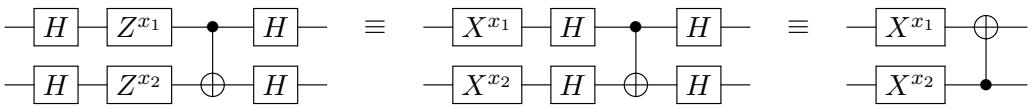

Figure 4: Detail of Alice's part in Figure 3

6.2 Tests

We now develop tests that, at the expense of some of the pairs, ensure that the remaining pairs are essentially a tensor product of EPR pairs, unentangled with the rest of the universe. If the starting state is of the desired form, the test should always accept. If the starting state is not close to one of the desired form, the test should either reject or purify the starting state so that the final state is close to one of the desired form.

EPR test. How are we going to perform the checking process? Alice and Bob could measure some of the qubits directly, which would actually destroy them but would leave enough of the joint state to be used in obtaining the random key. To see how direct measurement helps, we first consider a single two qubit state in the Bell basis. To detect whether there is a high weight on the $|\Psi\rangle$ terms (type 1), Alice and Bob both measure their qubit and compare their measurement outcomes using the classical channel. They reject if the outcomes differ, and continue the protocol otherwise, knowing that the state only has $|\Phi\rangle$ components (type 0).

The above procedure allows us to eliminate the unwanted $|\Psi\rangle$ components. What about $|\Phi^-\rangle$? As we analyzed, applying the Hadamard gate to both qubits of our base states preserves $|\Phi^+\rangle$ but switches $|\Phi^-\rangle$ and $|\Psi^+\rangle$. Applying the above test then eliminate the original $|\Phi^-\rangle$ component as well as the $|\Psi^-\rangle$ component. So, depending on whether we apply the Hadamard gate or not, we have two separate tests that combined allow us to eliminate the unwanted states. However, as the measurement collapses the state, we can only perform a single test on a given pair. Alice picks one of the two tests uniformly at random and sends a bit over the classical channel to Bob to coordinate the check. The state $|\Phi^+\rangle$ always passes the combined test. For a generic state $|\psi\rangle = \sum_{xy\in\{0,1\}^n} \alpha_{xy} |\Psi_{xy}\rangle$ we have that

$$\Pr[|\psi\rangle \text{ passes }] = |\alpha_{00}|^2 + \frac{1}{2}|\alpha_{01}|^2 + \frac{1}{2}|\alpha_{10}|^2 \leq \frac{1 - |\alpha_{00}|^2}{2}.$$

EPR* test. We now consider a pure state $|\psi\rangle$ for the whole of the n pairs. This can be seen as a superposition over all combinations of n tensors of the Bell basis. The direct approach would be to perform the check on some of the pairs of qubits. However, passing all of those tests is insufficient to guarantee that the remaining pairs are close to a tensor product of EPR pairs. In particular, if Eve tampers with a small number of pairs, the probability of detection is low. We need to use a better approach to broaden the detection range.

View the situation as follows. There are n possible combined tests Alice and Bob can perform, namely one for each of their n pairs. Each of those tests involves comparing two classical bits, one from Alice and one from Bob, which are obtained by measuring the corresponding qubits. Each individual test passes iff the bits are the same. We'd like to design a new test that rejects with high probability if at least one of those individual tests rejects, and accepts otherwise. We can do so by taking a random subset of the individual tests and reject iff the parity of the number of those that reject is odd. If all individual tests accept, the parity is always even; otherwise, it is odd with probability 50%. Moreover, Alice and Bob only need to sacrifice one qubit pair in order to execute this new test. They use the classical channel to select the subset, and then XOR into one of the selected positions (say the last one) all the other selected qubits. They then measure the one selected position, compare the measurement outcomes using the classical channel, and reject if they differ.

Recall that we need to choose between one of two tests for each pair: either apply H or not. We could accommodate different choices for individual pairs but it turns out that a single choice for all does the job. We go for the latter, simpler approach: Alice picks a random bit $b \in \{0,1\}$ and sends it to Bob over the classical channel: If $b = 1$ both Alice and Bob apply $H^{\otimes n}$ to their qubits before performing the XORs and measurement, otherwise they don't.

The fact that the Bell basis is invariant under owner-wise CNOTs facilitates the analysis of the effects of the test. We can write a pure state $|\psi\rangle$ of the n pairs as a superposition over all possible tensors of n Bell states

$$|\psi\rangle = \sum_{z \in \{\Phi^+,\Phi^-,\Psi^+,\Psi^-\}^n} \alpha_z |z\rangle ,$$

and know that the CNOT operations map each $|z\rangle$ into some $|z'\rangle$, where $z' = z$ for $z = (\Phi^+)^n$. As the component $|z\rangle$ for $z = (\Phi^+)^n$ passes all tests, it is maintained modulo a collapse of the measured pair to either $|00\rangle$ or $|11\rangle$. In case $b = 0$, components $|z\rangle$ for $z \notin \{\Phi^+,\Phi^-\}^n$ get annihilated for half of the choices of I. In case $b = 1$, components $|z\rangle$ for $z \notin \{\Phi^+,\Psi^+\}^n$ get annihilated for half of the choices of I. If we pick b uniformly at random, a component $|z\rangle$ for $z \neq (\Phi^+)^n$ gets annihilated with probability either $1/4$ or $1/2$.

6.3 Protocol and analysis

There are three steps in the protocol. First, Alice generates $|\Phi^+\rangle^{\otimes n}$ (the tensor product of n EPR pairs) and sends Bob his halves of each pair. Then Alice and Bob perform $n/2$ EPR* tests. If one or more of the tests fail, then Alice and Bob discard the qubits and restart the protocol. Otherwise, in the third and final step, they both measure their qubits of the remaining $n/2$ pairs, and retain the resulting string as their shared secret key.

Without Eve, step 2 starts with the n pairs in the state $|\Phi^+\rangle^{\otimes n}$, unentangled with the rest of the universe, each of the EPR* tests passes, so the state at the start of step 3 of the $n/2$ remaining pairs is $|\Phi^+\rangle^{\otimes n/2}$, after the measurement resulting in a uniformly at random chosen binary string of length $n/2$ that is unentangled with the rest of the universe.

We now argue the security of the protocol in the presence of Eve. During the transmission of the n qubits from Alice to Bob, they may have been tampered with and entangled with the rest of the universe. In general, the state of the universe at the start of the second step can be written as a pure state $\sum_j \beta_j |\psi_j\rangle |j\rangle$, where $|\psi_j\rangle$ denotes a pure state of the n pairs and $|j\rangle$ a basis state of the remainder of the universe. As before, we decompose the pure states $|\psi_j\rangle$ in the Bell basis:

$$\sum_j \beta_j |\psi_j\rangle = \sum_j \beta_j \left(\sum_{z \in \{\Phi^+,\Phi^-,\Psi^+,\Psi^-\}^n} \alpha_{z,j} |z\rangle \right) |j\rangle .$$

We analyze the evolution without renormalization, i.e., a test annihilates the components that are inconsistent with the test outcome but we do not renormalize the remaining superposition. The key observation is that the weight of the components $|(\Phi^+)^*\rangle$ does not change, and the expected weight of all components other than $|(\Phi^+)^*\rangle$ decreases by a factor of at least $3/4$ with each EPR* test. We distinguish between two cases.

- If the initial weight of $|\Phi^+\rangle^{\otimes n}$ is very small, then chances are at least one of the EPR* tests rejects.

- Otherwise, the sequence of EPR* tests makes the expected weight of the components other than $|\Phi^+\rangle^{\otimes n/2}$ negligibly small.

In the second case, the normalized state at the end of step 2 is very close to $|\Phi^+\rangle^{\otimes n/2}$ tensored with some state of the rest of the universe. Somewhat more quantitatively, if we haven't rejected after $n/2$ EPR* tests, the remaining $n/2$ qubits are exponentially close to the ideal state with exponentially high confidence. Note that "the rest of the universe" includes the measured qubits, as well as the other bits that were exchanged over the classical channel.

7 Bennett & Brassard

For comparison, we present the BB84 protocol without proof of security. For the initial communication over the quantum channel:

1. Alice randomly generates two binary strings $a = a_1 a_2 \dots a_n$ and $b = b_1 b_2 \dots b_n$ of length n. String a forms the basis of the secret key, and string b is used to decide how we encode a as qubits. If b_i is 0 she encodes a_i in the standard $\{|0\rangle, |1\rangle\}$ basis, otherwise she encodes a_i in the Hadamard basis $\{|+\rangle, |-\rangle\}$.

2. Alice sends the encoded qubits to Bob, who has no way of knowing which basis was used for each qubit. Instead, Bob generates a random string b' and use the i-th bit of b' to decide which basis he is going to measure the i-th qubit in. This gives Bob another string a'. We know for sure that the i-th bit of a' is the same as a_i if b' and b are the same on the corresponding bit.

Since the choice of b' is random, Bob is expected to get at least half of b correct. To decide on the secret key, Alice and Bob share their b and b' over the classical channel. The bits of a and a' where b and b' do not coincide are discarded.

If Eve has not observed or tampered with the encoded qubits, the resulting a and a' will be the same; otherwise they may not be in total agreement. Alice and Bob check if tampering has occurred as follows: Alice and Bob disclose a portion of their strings a and a' over the classical channel; these bits are removed from the secret key. If there are any differences, then we know that Eve has been tampering or observing the state, and the protocol is aborted. Otherwise, the remaining bits in a and a' form (two copies of) the secret key.

The intuition for the security of the protocol is that, if Eve does not know b_i, then the probability of her predicting a_i correctly is only 3/4.

References

[1] C. Bennett and G. Brassard Quantum Cryptography: Public key distribution and coin tossing. *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing (Bangalore, India, Dec.)*, pages 175–179, 1984.

[2] H. Lo and H. Chau. Unconditional Security of Quantum Key Distribution over Arbitrarily Long Distances. *arXiv:quant-ph/9803006v5*, 1999.

[3] D. Mayers. Unconditional security in quantum cryptography. *Journal of the ACM, vol. 48, no. 3*, pages 351–406. 2001.