11 Perfect security from physical assumptions

11.1 The noisy storage model

As we saw in the previous chapter, security for two-party cryptography is hard to achieve. Even when given the ability to use quantum communication, we still cannot hope to achieve the security conditions. Yet, you maybe see that these are important problems that we need to solve on an everyday basis! One is therefore willing to make some reasonable assumptions on how powerful the dishonest party (adversary) can be, in order to obtain security guarantees. Security will thus hold as long as these assumptions are satisfied.

Classically, one typically makes so-called computational assumptions. This means that we assume that a particular problem, like factoring a large integer, requires a large amount of computational resources, and furthermore that the adversary has a relatively limited amount of computational resources - namely, insufficient to solve the difficult problem within a reasonable time frame. Once we build a quantum computer, however, many of these assumptions will no longer be valid! For example, we know that the Shor algorithm provides a much more efficient way to factor large numbers on a quantum computer! What's more, however, security can be broken retroactively: that is, if we build a quantum computer tomorrow, most two-party protocols that have been executed to date will lose their security/secrecy, since the adversary can now use the quantum computer to break the protocol. Clearly, this is very undesirable.

One way out of this dilemma is to make physical rather than computational assumptions. These assumptions only need to be valid during the execution of the protocol, but become irrelevant afterwards. Here, we will take a very simple assumption, namely we assume that the adversary can only store q qubits during one particular point during the protocol. Otherwise, the adversary remains all powerful: he/she may perform arbitrary quantum operations/computations, arbitrary encoding and decoding procedures, and store an infinite amount of classical information. This assumption is referred to as the bounded storage model. More generally, one can also invoke the noisy storage model, where the storage is not only bounded, but also noisy in general. In such cases, upper bounds quantities such as the entanglement assisted quantum capacity of the memory [**?**] leads to security. Given that even the most sophisticated experimental realization of quantum memories to date can reliably store no more than a few qubits for a few milliseconds [**?**, **?**, **?**], this is a technologically well motivated assumption. Before, or after the execution of the protocol, however, the attacker is allowed to have an arbitrary quantum memory, and even a quantum computer. In particular, this means that if tomorrow we can build better quantum memories, security can nevertheless never be broken retroactively.

In the next section, we have a look at a very simple protocol for this task.

11.2 1-2 Oblivious Transfer in the noisy-storage model

We have encountered the oblivious transferm in Week 8, and have seen that this is an important protocol, in the sense that any two-party protocol may be achieved by a combination of 1-2 oblivious transfer protocols, making it a fundamental building block of interest. We have also seen that despite the attempt to construct OT protocols that make use of quantum communication, no secure protocols exist – a malicious Bob could easily break the protocol by using for example a large quantum memory. Fortunately, this is extremely difficult to do in current-day technology, and therefore one may, by using the noisy storage model assumption, prove security for any Bob that has a limited quantum memory.

Protocol 1 *Protocol for 1-2 OT in the noisy-storage model. Alice has inputs $s_0, s_1 \in \{0,1\}^\ell$, Bob has input $y \in \{0,1\}$.*

1 *Alice chooses a random strings $x = x_1, \ldots, x_n \in \{0,1\}^n$ and random basis $\theta = \theta_1, \ldots, \theta_n \in \{0,1\}^n$. She sends the bits encoded in the randomly chosen BB84 bases $H^{\theta_1} |x_1\rangle \otimes \ldots \otimes H^{\theta_n} |x_n\rangle$ to Bob.*
2 *If $y = 0$, then Bob measures all of the qubits in the standard basis. If $y = 1$, he measures in the Hadamard basis. He records the resulting outcome string $\tilde{x} = \tilde{x}_1, \ldots, \tilde{x}_n$.*
3 *Both parties wait time Δt. (Storage assumption is applied!)*
4 *Alice sends to Bob the string $\theta_1, \ldots, \theta_n$. Bob computes the set of indicdes $\mathcal{I} = \{j \mid \theta_j = y\}$ where the measured in the same basis than Alice.*
5 *Alice chooses two random extractor functions as specified by random seeds r_0 and r_1. She computes $k_0 = Ext(x_+, r_0)$ and $k_1 = Ext(x_\times, r_1)$. Where x_+ is the substring of x where Alice encoded in the standard basis, and $x_\times$ is the substring where she used the Hadamard basis. She sends r_0 and r_1 to Bob.*
6 *Alice sends to Bob $m_0 = s_0 \oplus k_0$ and $m_1 = s_1 \oplus k_1$, where $\oplus$ denotes the bit wise xor.*
7 *Bob computes $k = Ext(x_\mathcal{I}, r_y)$ and $s_y = k_y \oplus r_y$.*

Why does this protocol work? Let us first check that the protocol is correct, that is, Bob actually obtains s_y! Note that if there is no noise, then whenever $\theta_j = y$, we have $x_j = \tilde{x}_j$. That is, whenever Alice had encoded in the basis in which Bob measures, then Bob learns the corresponding element of Alice's bit string. This means that if Alice's now applies an extractor to hash down the elements of the strings corresponding to the standard and Hadamard basis respectively, then Bob knows one of them. Since Alice also sends him r_0 and r_1, he hence learns k_y, which acts like a key that encrypts s_y using one-time pad encryption, allowing him to recover s_y. Similar to the case of QKD, when a small amount of errors occur on the channel, information reconciliation can be performed in order to ensure that Bob is able to correct for the errors in $\tilde{x}$.

11.3 Security from quantum uncertainty

Is Protocol 1 secure? Let us first check security against dishonest Alice. Here, we want to show that Alice cannot learn y. If you look at the protocol above, it is clear that this is definitely the case: Bob never sends any information at all to Alice, from which we could learn anything about y!

The only difficulty is thus to show security against dishonest Bob. Here, we want to show that while Bob might learn one of the two strings, there is always a string $s_{\bar{c}}$ about which he knows (almost) nothing. As you can see from a protocol above, we have used our favorite trick encountered already in QKD, namely privacy amplification/randomness extraction. This means that if we could only ensure somehow that Bob's min-entropy about either x_+ or $x_\times$ is high, then by the properties of privacy amplification we could be sure that he knows nothing about the extracted key.

Before we do this, let us first consider whether we can say anything at all about Bob's min-entropy about *the entire* string x. To reason about this, let us first observe that something magic needs to happen in the waiting time Δt. Clearly, if Bob could store all of Alice's qubits, then he can just measure the entire string in the correct basis and learn the whole string x. This means that security against Bob can never be achieved if the number q of qubits that Bob can store is $q \geq n$.

Let us thus assume that $q < n$. Note that since we allow Bob to have an arbitrary quantum memory and computer *before* the waiting time, he can first store all of Alice's qubits, and perform an arbitrary quantum operation on all of them. For example, he might measure some of them, resulting in some classical information K. However, he can then keep only q qubits in his quantum register which we denote by Q. If we are interested in Bob's min-entropy about the entire string, we thus want to bound

$$H_{\min}(X_+X_\times|\Theta, K, Q) \tag{11.1}$$

where we have written $X_+X_\times$ for the string X to remind ourselves that we are ultimately interested in the two portions corresponding to the two different bases. To make his guess, Bob can use the classical information K, the quantum register Q, as well as the basis information Θ that Alice sends to him after the waiting time.

In QKD, we saw that is it often much easier to show security against an adversary who is purely classical, so let us try and get rid of Q. As we have done so before in previous weeks, we can apply the chain rule for the min-entropy. Recall that the chain rule says

$$\begin{aligned} H_{\min}(X_+X_\times|\Theta, K, Q) &\geq H_{\min}(X_+X_\times|\Theta, K) - \log|Q| \;, & (11.2)\\ &= H_{\min}(X_+X_\times|\Theta, K) - q \;. & (11.3) \end{aligned}$$

Hence, we could worry about a Bob who has only Θ and K. How could we possibly analyze this?

Again, let us think back to the tricks learned in QKD! By the guessing game, we could again think of Bob preparing qubits and send them to Alice. Alice chooses one of two random basis, after which she announces the basis choice to Bob. That is, we can use precisely the same guessing game that we had used in QKD to analzye the case of an eavesdropper Eve who has only classical information K! This gives the familiar

$$H_{\min}(X_+X_\times|\Theta, K) = n\left[-\log\left(\frac{1}{2} + \frac{1}{2\sqrt{2}}\right)\right] \approx 0.22n \;. \tag{11.4}$$

Of course, what we really want is to make a statement about the different parts X_+ and $X_\times$. That is, we want that there exists a $\bar{c} \in \{+, \times\}$ such that Bob's entropy about $X_{\bar{c}}$ is high. Fortunately, there is a beautiful lemma known as the *min-entropy splitting lemma* proven by Wullschleger which says that there exists some register $\bar{C}$ such that

$$H_{\min}(X_{\bar{C}}|\Theta, K, \bar{C}) \geq \frac{H_{\min}(X_+X_\times|\Theta, K)}{2} - 1 \;. \tag{11.5}$$

It is noteworthy that min-entropy splitting only holds if K really is classical which is why we first have to get rid of q. We thus know that Bob's min-entropy about at least one of the strings is large! Putting the two ideas together we thus have

$$\begin{aligned} H_{\min}(X_{\bar{C}}|\Theta, K, \bar{C}Q) &\geq H_{\min}(X_{\bar{C}}|\Theta, K, \bar{C}) - q & (11.6)\\ &\geq \frac{H_{\min}(X_+X_\times|\Theta, K)}{2} - 1 - q \;. & (11.7) \end{aligned}$$

As usual we can now employ privacy amplifcation to say that Bob is ϵ-close to being ignorant, whenever

$$\begin{aligned} \ell &< H_{\min}(X_{\bar{C}}|\Theta, K, \bar{C}, Q) - O(\log 1/\varepsilon) - 1 & (11.8)\\ &\approx 0.11n - q - O(\log 1/\varepsilon) - 2 \;. & (11.9) \end{aligned}$$

This means that whenever $q \lesssim 0.11n$ we can have security for some $\ell > 0$! Or, reading it the other way around, assuming a maximum q for the adversary tells us that we need to send at least $n \approx 1/0.11q$ qubits in order to achieve security. By much more sophisticated analysis, it is now possible to show that security can be achieved as long as $q \leq n - O(\log^2 n)$ which is essentially optimal. We thus see that security is possible by sending just a few more qubits than Bob can store.