Notes for Lecture 20

1 Authenticating Quantum States

Today we're going to talk about authenticating quantum states. Suppose Alice and Bob have a shared classical key k, and Alice wants to send a (mixed) state ρ to Bob, but there's an adversary who can modify ρ to be ρ'. In the classical setting we solved this with a message authentication code, but in the quantum setting it turns out that just appending a tag doesn't work here.

1.1 Authentication Implies Encryption

But first we'll show that quantum authentication implies encryption. Suppose we have some procedure $\mathsf{Auth}(k, \rho) \to \hat{\rho}$. Suppose it's possible to distinguish $\mathsf{Auth}(k, |0\rangle)$ from $\mathsf{Auth}(k, |1\rangle)$, and that I can distinguish perfectly (but it turns out we can make this work without assuming perfect distinguishing). Note that I'm using orthogonal states, but we can actually make this work even without orthogonal states.

I'll consider $\mathsf{Auth}(k, \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle)$. We'll say the authentication scheme is a unitary, so we'll write

$$\begin{aligned} \mathsf{Auth}(k, |0\rangle) &= U_k|0\rangle \\ \mathsf{Auth}(k, |1\rangle) &= U_k|1\rangle. \\ \mathsf{Auth}(k, \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle) &= \frac{1}{\sqrt{2}}U_k|0\rangle + \frac{1}{\sqrt{2}}U_k|1\rangle \end{aligned}$$

I'm going to use the adversary that distinguishes the first two authentications to maul the third authentication to

$$\frac{1}{\sqrt{2}}(U_k|0\rangle)|1\rangle + \frac{1}{\sqrt{2}}(U_k|1\rangle) \otimes |1\rangle$$

In other words, I'm using my adversary to write whether I get 0 or 1 in some new qubit, and then I flip the one corresponding to 1, and then undo the adversary to get

$$\frac{1}{\sqrt{2}}U_k|0\rangle - \frac{1}{\sqrt{2}}U_k|1\rangle = \mathsf{Auth}(k, \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle)$$

In other words, using the distinguisher we were able to form an authentication of one state to an authentication of another state.

So what does the fact that authentication implies encryption mean? It actually causes us a lot of problems. In particular it means that I can't sign a state by just appending a header. So any quantum authentication that I want to authenticate has to be secret.

So how should I define security? One issue is that states are a continuous quantity instead of a discrete quantity as they are in the classical setting, so how do we detect an extremely small perturbation to the quantum state? And how do we formalize what it means for the adversary to produce a different authenticated state?

1.2 Security Definition

There have been a few attempts in the literature. What seems to be the best one is based on the following. Let's first ask: What can an adversary trivially do?

The adversary can do nothing, they can drop the message, they can do an extremely small perturbation. The adversary can measure the first qubit. Since this only has two outcomes, the probability Bob accepts is still at least 1/2, but the state that's being sent over has clearly been tampered with. This is actually a more general principal that if you do a measurement that only has 1 of 2 outcomes, you can only decrease the probability of something happening later by a 1/2 factor. There are ways to protect against these measuring attacks, but there's nothing we can do to prevent the adversary from deciding whether or not they want to forward the message along.

The intuition for the security definition is that the only adversaries that exist are these "ideal" adversaries considered above.

Let's try to make this a bit more rigorous. I'm going to allow mixed quantum states to have trace ≤ 1. This corresponds to the adversary sending nothing. A trace of 0 corresponds to the adversary sending nothing, and a trace of ρ is just the probability ρ exists at all.

So the syntax for quantum authentication is

- $\mathsf{Auth}(k, \rho) \to \hat{\rho}$. We'll have $Tr[\rho] = Tr[\hat{\rho}]$.
- $\mathsf{Ver}(k, \hat{\rho}) \to \rho$. Here we might have an abort, so we mean that $Tr[\rho] \leq Tr[\hat{\rho}]$.

So correctness says that $\mathsf{Ver}(k, \mathsf{Auth}(k, \rho)) \approx \rho$.

For security, we'll first define$\mathcal{I}$ to be a set of ideal adversaries. In particular,

$$\mathcal{I} = \{\rho \to c\rho \text{ for } 0 \leq c \leq 1\}.$$

So what we mean is that the adversary can only scale the whole density matrix down, but it can't vary any components individually.

We'll say a scheme is ϵ-secure if for any real adversary $A, \exists I \in \mathcal{I}$ such that $\forall \rho$,

$$\frac{1}{|\mathcal{K}|}\sum_k \mathsf{Ver}(k, A(\mathsf{Auth}(k,\rho))) \approx_\epsilon \frac{1}{|\mathcal{K}|}\sum_k \mathsf{Ver}(k, I(\mathsf{Auth}(k,\rho)))$$

So in particular I pick a random key k, and then the result of the adversary is ϵ-close to the result of the ideal adversary.

Consider, for example, the adversary A that measures the first qubit. Sometimes this adversary will still pass verification and sometimes it will not. And we can have an ideal adversary that simply forwards along the original message so that it gets accepted with the same probability A gets accepted. What this definition means is that the adversary that measures first qubit might as well have not measured the first qubit, but could have aborted with some probability and just forwarded the state. So we can think of the verification that accepts as "correcting" the state.

So why can't we insist on equality? I can just try to guess a random key k, which will allow me to perform a successful mauling (and clearly this can't be simulated by the ideal adversary).

So what does it mean for $\rho \approx_\epsilon \rho'$? We mean that $|\rho - \rho_1| \leq \epsilon$ where we're taking the trace norm, which corresponds to $\sum_i |\lambda_i|$. A fact is that $\rho \approx_\epsilon \rho'$ iff $\exists$ an adversary that can distinguish with probability ϵ. Note that this corresponds to the classical notion of statistical distance.

1.3 Constructions

There's actually a few constructions out there. Here's one.

- The key is a random unitary U.
- $\mathsf{Auth}(U, \rho)$: apply U to $\rho \otimes |0^n\rangle\langle 0^n|$. The classical analogue of this would be to append your message with some 0's and then apply a random permutation. This authenticates because the adversary doesn't know the random permutation, so if they change to anything else, they won't get something with a bunch of 0's at the end. It turns out this works in the quantum setting as well.
- $\mathsf{Ver}(U, r\hat{h}o)$: Apply $U^\dagger$ to $\hat{\rho}$ and measure the last n bits. If the last n bits are not 0^n, reject, and otherwise output ρ (i.e. first few bits).

Efficient: Unitary 2-design, which is the quantum analogue of pairwise independent functions.

Here's another construction

- First apply classical authentication in superposition.

- Then apply the QFT.
- Then apply classical authentication in superposition.

The classical authentication scheme needs to satisfy certain strong security properties for this transformation to work. Essentially, authenticating in the computational basis and the fourier basis turns out to be enough, despite the fact that a classical authentication scheme is not enough.

The idea for this scheme comes from the impossibility result, which said that if I can authenticate $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ and $\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$, this implies encryption of $|0\rangle, |1\rangle$. So this is really authenticating $H|0\rangle$ and $H|1\rangle$. So authenticating in the Hadamard basis is encrypting in the computational (and vice versa), so to encrypt the entire state I just need to authenticate in both bases.

Next week we'll look at some ways to strengthen the security definition. In the classical setting if we want to authenticate or encrypt an unbounded number of messages, we always need computational assumptions. Remarkably, in the quantum setting it turns out you can start recycling your key information theoretically.