1 Probability of Eve measuring the key

Given an ε-secure QKD protocol, we want to bound the probability that Eve measures the key in her register. After the QKD protocol we have a ρ_{real} which is ε-close to some $\rho_{ideal} \in S_{ideal}$. As ρ_{real} is difficult to work with, we find the probability on ρ_{ideal}.

After QKD, Eve can run some post-processing $\mathcal{E}$ in order to try to arrive at the key. This is an operation that only applies to Eve's register, so we can the operation on the whole state as $I \otimes I \otimes \mathcal{E}$. This operation just maps ρ_E to an unknown ρ'_E.

$$\begin{aligned}
\rho_{ideal} &= \sum_{k\in\{0,1\}^n} 2^{-n}|k\rangle\langle k| \otimes |k\rangle\langle k| \otimes \rho_E \\
\rho'_{ideal} &= \sum_{k\in\{0,1\}^n} 2^{-n}|k\rangle\langle k| \otimes |k\rangle\langle k| \otimes \mathcal{E}(\rho_E) = \\
&= \sum_{k\in\{0,1\}^n} 2^{-n}|k\rangle\langle k| \otimes |k\rangle\langle k| \otimes \rho'_E
\end{aligned}$$

We want to bound the probability $\Pr[k_A = k_E]$ where k_A, k_E refer to Alice and Eve's key registers. We define P_k as a projector which measures a certain k on both Alice's and Eve's registers. Then $\Pr[k_A = k_E]$ is the sum of applying these P_k's on the density op.

$$\begin{aligned}
P_k &= |k\rangle\langle k| \otimes I \otimes |k\rangle\langle k| \\
\Pr[k_A = k_E] &= \sum_{k\in\{0,1\}^n} tr\ P_k\rho_{real} \leq \sum_{k\in\{0,1\}^n} tr\ P_k\rho_{ideal} = \\
&= \sum_{k\in\{0,1\}^n} tr\ P_k(\sum_{k'\in\{0,1\}^n} 2^{-n}|k'\rangle\langle k'| \otimes |k'\rangle\langle k'| \otimes \rho'_E) \\
&= \sum_{k\in\{0,1\}^n} tr\ 2^{-n}(|k\rangle\langle k| \otimes I \otimes |k\rangle\langle k|)(\sum_{k'\in\{0,1\}^n} |k'\rangle\langle k'| \otimes |k'\rangle\langle k'| \otimes \rho'_E) \\
&= \sum_{k\in\{0,1\}^n, k'\in\{0,1\}^n} tr\ 2^{-n}|k\rangle\langle k||k'\rangle\langle k'| \otimes |k'\rangle\langle k'| \otimes |k\rangle\langle k|\rho'_E
\end{aligned}$$

Now $|k\rangle\langle k||k'\rangle\langle k'|$ is multiplied with 0 when $k \neq k'$. This is because $|k\rangle$ are all basis states, and $\langle k||k'\rangle = 0$ if they're not equal, 1 otherwise. Thus the double sum simplifies.

$$\sum_{k\in\{0,1\}^n, k'\in\{0,1\}^n} tr\ 2^{-n}|k\rangle\langle k||k'\rangle\langle k'| \otimes |k'\rangle\langle k'| \otimes |k\rangle\langle k|\rho'_E =$$
$$\sum_{k\in\{0,1\}^n} tr\ 2^{-n}|k\rangle\langle k| \otimes |k\rangle\langle k| \otimes |k\rangle\langle k|\rho'_E =$$
$$\sum_{k\in\{0,1\}^n} tr\ 2^{-n}|k\rangle\langle k|\rho'_E = 2^{-n}$$

These last steps used a few properties that we now explain hold for computational basis vectors i and any density operator σ.

First we use the fact that for a computational basis vector i, $tr\ |i\rangle\langle i| \otimes \sigma = tr\ \sigma$. This is because $|i\rangle\langle i|$ has only a 1 somewhere on the main diagonal. And thus $|i\rangle\langle i| \otimes \sigma$ will put σ along the main diagonal.

$$|i\rangle\langle i| = \begin{pmatrix} 0 & \dots & 0 & \dots & 0 \\ 0 & \ddots & \vdots & \dots & 0 \\ 0 & \dots & 1 & \dots & 0 \\ 0 & \dots & \vdots & \ddots & 0 \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix},\ |i\rangle\langle i| \otimes \sigma = \begin{pmatrix} 0 & \dots & 0 & \dots & 0 \\ 0 & \ddots & \vdots & \dots & 0 \\ 0 & \dots & \sigma & \dots & 0 \\ 0 & \dots & \vdots & \ddots & 0 \\ 0 & \dots & \dots & \dots & 0 \end{pmatrix}$$

Thus it is easy to see that $tr\ |i\rangle\langle i| \otimes \sigma = tr\ \sigma$.

Second, to see that for computational basis vectors k, $\sum_{k\in\{0,1\}^n} tr\ 2^{-n}|k\rangle\langle k|\rho'_E = 2^{-n}$, we have that

$$\begin{aligned} \sigma &= \sum_{ij} \alpha_{ij}|i\rangle\langle j| \\ tr\ |k\rangle\langle k|\sigma &= tr\ |k\rangle\langle k||k\rangle\langle k|\sigma = tr\ |k\rangle\langle k|\sigma|k\rangle\langle k| \\ &= tr\ |k\rangle\langle k|(\sum_{ij} \alpha_{ij}|i\rangle\langle j|)|k\rangle\langle k| = \\ &= tr\ \sum_{ij} \alpha_{ij}|k\rangle\langle k||i\rangle\langle j||k\rangle\langle k| = \\ &= tr\ \alpha_{kk}|k\rangle\langle k| = \alpha_{kk} \end{aligned}$$

And α_{kk} is the k-th element on the diagonal, applying this for all k gives us all the elements of the diagonal, which is the full trace of σ. The third property we used was that the trace of a density matrix is 1 by definition.

And thus we have shown that for a ρ_{ideal}, the probability of Eve measuring the key is $\frac{1}{2^n}$. However, as ρ_{real} is ε-close to ρ_{ideal} from QKD, we know that the probability of measuring the key is $\varepsilon + \frac{1}{2^n}$.

2 SMT from QKD

Now that we've gotten familiar with the security definition of QKD, let's try to use the keys from a secure QKD protocol in other situations. First we'll look at secure message transfer, and then we'll look at how the keys can be used to log in.

2.1 Setup and security def

The protocol works as follows:

- Alice and Bob perform the ε-secure QKD protocol to get keys K_A and K_B. If QKD aborts, we cancel.
- To secure a message m, Alice sends Bob the state $|c\rangle := |m \oplus K_A\rangle$[1]. Since we assume the channel is public, she also sends this to Eve.
- Bob retrieves the ciphertext and retrieves the original message by retrieving $|m'\rangle = |c \oplus K_B\rangle$. Since $K_A = K_B$ after the error-correction of QKD, this just returns the original m.

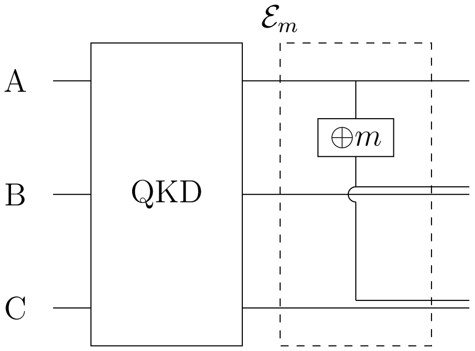

Figure 1: The quantum operator $\mathcal{E}_m$ for sending message m.

We denote sending the message m (after QKD has already been run) as a quantum operator $\mathcal{E}_m$. This is just a natural application of the one-time pad in our setting. But is this protocol secure? As usual, the answer to this is "What do you mean by secure?" as there are many different notions of security, and choosing the right definition depends on what the protocol is used for.

[1] While these are really classical values, I'll keep them as basis vectors of a quantum system since it allows me to use density operators naturally. Also note that the XOR operation can be easily done with a unitary as it just shuffles the computational basis states around and is self-inverse.

In our case, let's say we only want that Eve learns nothing about the message. The most natural way to go about formalizing this is to say that Eve can't distinguish between the ciphertexts of any two messages. Since Eve had the small probability ε of succeeding in learning something from QKD, we must also limit ourselves to saying that Eve can distinguish with some small probability δ. We've already seen how trace distance captures the idea of distinguishability between two density operators.

$$\forall E, m_0, m_1 \; TD(tr_{AB} \; \rho_{m_0}, tr_{AB} \; \rho_{m_1}) \leq \delta$$

The above definition looks at the view of Eve, which is why the registers of Alice and Bob are traced out (remember that limiting our view is also what partial trace models).

2.2 Security proof overview

Now let's look at two runs of the SMT protocol for some m_0,m_1 and try to bound the distance between them. **The main idea is to use ideal "intermediate" steps to get a bound on the maximum distance from one state to the other.**

First, both of them run the QKD protocol, giving them each a $\tilde{\rho}_{real_m}$ which is ε-close to $\tilde{\rho}_{ideal}$. Note that the $\tilde{\rho}_{ideal}$ is the same for both runs, because the $\tilde{\rho}_{ideal}$ is fixed for a given adversary and we're using the same Eve in both runs. In the schematics one run is represented on the right, and the other on the left.

$$\begin{tikzcd}
\tilde{\rho}_{real_{m_0}} \arrow[r, dash, "\varepsilon"] & \tilde{\rho}_{ideal} \arrow[r, dash, "\varepsilon"] & \tilde{\rho}_{real_{m_1}}
\end{tikzcd}$$

Figure 2: Bound on distance when only QKD has been run.

And since they're both ε-close to the ideal, we know that $TD(\tilde{\rho}_{real_{m_0}}, \tilde{\rho}_{real_{m_1}}) \leq 2 \cdot \varepsilon$.

We then run the rest of the SMT protocol, applying $\mathcal{E}_{m_1}$ and $\mathcal{E}_{m_0}$ in both the real and ideal cases. Note that since applying some operation can only lose information, $TD(\mathcal{E}(\rho), \mathcal{E}(\tau)) \leq TD(\rho, \tau)$ which is something we already saw in the trace distance lecture. Thus the distance between the reals and the ideals will not grow larger than ε. But the distance between the corresponding ideal cases can grow. $TD(\mathcal{E}_{m_0}(\tilde{\rho}_{ideal}), \mathcal{E}_{m_1}(\tilde{\rho}_{ideal})) \nleq TD(\tilde{\rho}_{ideal}, \tilde{\rho}_{ideal})$ as we're applying different operators to each side.

However, the thing we care about is the distance between the two message states from Eve's perspective, and to get this limited view of the system we used the partial trace. Since we know from one of the homework tasks that the partial

$$\begin{tikzcd}
\tilde{\rho}_{real_{m_0}} \arrow[rrr, dash, "\varepsilon"] \arrow[d, "\mathcal{E}_{m_0}"'] & & & \tilde{\rho}_{ideal} \arrow[rrr, dash, "\varepsilon"] \arrow[dl, "\mathcal{E}_{m_0}"'] \arrow[dr, "\mathcal{E}_{m_1}"] & & & \tilde{\rho}_{real_{m_1}} \arrow[d, "\mathcal{E}_{m_1}"] \\
\tilde{\rho}'_{real_{m_0}} \arrow[rr, dash, "\varepsilon"] & & \tilde{\rho}'_{ideal_{m_0}} \arrow[rr, dash, "??"] & & \tilde{\rho}'_{ideal_{m_1}} \arrow[rr, dash, "\varepsilon"] & & \tilde{\rho}'_{real_{m_1}}
\end{tikzcd}$$

Figure 3: Bound on distance when SMT has been run.

$$\begin{tikzcd}
\tilde{\rho}_{real_{m_0}} \arrow[rrr, dash, "\varepsilon"] \arrow[d, "\mathcal{E}_{m_0}"'] & & & \tilde{\rho}_{ideal} \arrow[rrr, dash, "\varepsilon"] \arrow[dl, "\mathcal{E}_{m_0}"'] \arrow[dr, "\mathcal{E}_{m_1}"] & & & \tilde{\rho}_{real_{m_1}} \arrow[d, "\mathcal{E}_{m_1}"] \\
\tilde{\rho}'_{real_{m_0}} \arrow[rr, dash, "\varepsilon"] \arrow[d, "tr_{A,B}"'] & & \tilde{\rho}'_{ideal_{m_0}} \arrow[rr, dash, "??"] \arrow[d, "tr_{A,B}"] & & \tilde{\rho}'_{ideal_{m_1}} \arrow[rr, dash, "\varepsilon"] \arrow[d, "tr_{A,B}"] & & \tilde{\rho}'_{real_{m_1}} \arrow[d, "tr_{A,B}"] \\
\tilde{\rho}''_{real_{m_0}} \arrow[rr, dash, "\varepsilon"] & & \tilde{\rho}''_{ideal_{m_0}} \arrow[rr, dash, "\sigma"] & & \tilde{\rho}''_{ideal_{m_1}} \arrow[rr, dash, "\varepsilon"] & & \tilde{\rho}''_{real_{m_1}}
\end{tikzcd}$$

Figure 4: Bound on distance when SMT has been run, Eve's view only.

trace is also a quantum operator, it also preserves the trace distance bounds. This is pictured on Figure 4.

The states $\tilde{\rho}''_{real_{m_0}}, \tilde{\rho}''_{real_{m_1}}$ correspond to the states in the security definition for the SMT protocol - they're density operators after we've ran the protocol for two different messages and traced away Alice and Bob's part. So we know that our scheme is $\delta = \varepsilon + \sigma + \varepsilon$-secure. However, we currently don't know how large σ can be - if it is very large, the scheme is still insecure.

2.3 Upper bound of σ

We have shown that the security of the scheme comes down to finding a bound on the size of σ. This is much easier to do, because we're no longer dealing with the "real" states, which can be difficult to express and work with as they can have errors. The ideal case $\tilde{\rho}_{ideal}$ (the state after only QKD has been run) is simply

$$\tilde{\rho}_{ideal} = p(2^{-n} \sum_{x \in \{0,1\}^n} |x\rangle\langle x|_A \otimes |x\rangle\langle x|_B \otimes \rho_E) + (1-p)\rho_{abort}$$

The A,B subscripts distinguish between what parts belong to Alice and Bob.

Then for m_0, the state after running SMT is

$$\begin{aligned}\mathcal{E}_{m_0}(\tilde{\rho}_{ideal}) =& p(2^{-n} \sum_{x\in\{0,1\}^n} |x\rangle\langle x|_A \otimes |x\rangle\langle x|_B \otimes |x \oplus m_0\rangle\langle x \oplus m_0|_B \otimes |x \oplus m_0\rangle\langle x \oplus m_0|_E \otimes \rho_E) \\ &+ (1-p)\rho_{abort} = \tilde{\rho}'_{ideal_{m_0}}\end{aligned}$$

It may be helpful to look at Figure 1 again to orient yourself in this formula. Now to get $\tilde{\rho}''_{ideal_{m_0}}$, we only need to trace away Alice and Bob's systems and leave only Eve's part. Because $|x\rangle\langle x|$ has trace 1 for basis states x, this is quite simple.

$$\begin{aligned}tr_{A,B}\ \tilde{\rho}'_{ideal_{m_0}} &= p(2^{-n} \sum_{x\in\{0,1\}^n} |x \oplus m_0\rangle\langle x \oplus m_0|_E \otimes \rho_E) + (1-p)\rho'_{abort} \\ &= \tilde{\rho}''_{ideal_{m_0}}\end{aligned}$$

You may have noticed we haven't dealt with the abort state at all - this is because it is always the same, as we don't run SMT if we abort. However, here we still need to trace the abort state as well, as we're still limiting our view (and the matrices need to be the same size to add together).

Now you may remember from Homework 3 task 1.b that applying a unitary to an equal superposition of basis states does nothing.

$$\begin{aligned}2^{-n} \sum_{x\in\{0,1\}^n} U|x\rangle\langle x|U^\dagger &= 2^{-n}U \sum_{x\in\{0,1\}^n} U^\dagger = 2^{-n}UIU^\dagger \\ &= 2^{-n}UU^\dagger = 2^{-n}I = 2^{-n} \sum_{x\in\{0,1\}^n} |x\rangle\langle x|\end{aligned}$$

In our case, we're applying an XOR function where this is especially easy to see as it just maps basis states to other basis states. And since each basis state has the same probability[2], this shuffling around does nothing. And since ρ_E is independent from x, we can now reindex the basis states. This is the exact same reason why a one-time pad works.

$$\begin{aligned}\tilde{\rho}''_{ideal_{m_0}} &= p(2^{-n} \sum_{x\in\{0,1\}^n} |x \oplus m_0\rangle\langle x \oplus m_0|_E \otimes \rho_E) + (1-p)\rho'_{abort} \\ &= p(2^{-n} \sum_{x\in\{0,1\}^n} |x\rangle\langle x|_E \otimes \rho_E) + (1-p)\rho'_{abort}\end{aligned}$$

Now notice that the above state is completely independent from the choice of m_0. If we were to compute the same process using e.g. m_1, we would get the exact same result. And since they're the exact same state, $\sigma = TD(\tilde{\rho}''_{ideal_{m_0}}, \tilde{\rho}''_{ideal_{m_1}}) = 0$. And thus our SMT protocol is $\delta = \varepsilon + \sigma + \varepsilon = 2 \cdot \varepsilon$-secure.

[2] Probability rather than amplitude because we have a distribution of classical states when we have $\sum_x p_x |x\rangle\langle x|$, not a superposition