Advanced Cryptography — Midterm Exam
Solution

Serge Vaudenay

3.5.2018

- duration: 1h45
- any document allowed
- a pocket calculator is allowed
- communication devices are not allowed
- the exam invigilators will **not** answer any technical question during the exam
- readability and style of writing will be part of the grade

The exam grade follows a linear scale in which each question has the same weight.

1 Threshold Implementation to Mitigate Power Cryptanalysis

This exercise is inspired from Vaudenay, Side-Channel Attacks on Threshold Implementations using a Glitch Algebra*, CANS 2016, LNCS vol. 10052, Springer.*

We consider a hardware circuit to implement a cryptographic function F mapping k secret key bits and p input bits to q output bits:

$$\begin{array}{rcccl} F: \{0,1\}^k & \times & \{0,1\}^p & \longrightarrow & \{0,1\}^q \\ (K & , & x) & \longmapsto & y \end{array}$$

We assume that the circuit is composed of AND gates (denoted by $\wedge$), XOR gates (denoted by $\oplus$), and wires. The circuit works following a clock signal. During each time period, the wires have constant signals and the gates propagate the computations (with a small latency). Gates normally dissipate no power. So, the total power consumption of a circuit is normally null during each time period. However, a wire could have a *glitch* which makes gates compute more during the time period, trying to follow the glitch in the signal. In that case, gates dissipate power and may reproduce the glitch with a small latency to their output. To simplify the analysis, we assume that during each time period, a wire w represents a bit $v_w \in \{0,1\}$ and has a number of glitches equal to n_w. Concretely, we assume the following behaviors for a gate $g : (a,b) \to c$ with input wires a and b and output wire c:

- for a $\wedge$ gate: $v_c = v_a v_b$ and $n_c = v_a n_b + v_b n_a$;
- for a $\oplus$ gate: $v_c = (v_a + v_b) \bmod 2$ and $n_c = n_a + n_b$;

- the gate dissipates an energy equal to $H_g = n_c h_g$, where h_g is a constant depending on the gate g (i.e., $h_g = h_\wedge$ for an AND gate and $h_g = h_\oplus$ for a XOR gate).

We consider a hardware implementation with a built-in secret $K \in \{0,1\}^k$ which is randomly set up at the beginning, and unknown to the adversary. The goal of the adversary is to recover K. The adversary can arbitrarily select the input x, get $y = F(K,x)$, and see the total amount of energy $H = \sum_g H_g$ which is dissipated during each time period. We assume that the adversary knows the structure of the hardware circuit. We further assume that $n_w = 0$ for all the input gates except one special wire w_0 for which $n_{w_0} = 1$. The adversary knows w_0 and n_{w_0} as well.

Q.1 To start with a simple example, we assume that w_0 is such that $v_{w_0} = x_i$, the ith input bit in x, and that w_0 is an input wire to an AND gate g where the second input wire is w_1 such that $v_{w_1} = K_j$, the jth bit of K. Show how the adversary can obtain K_j in this attack model.

Let w_2 be the output wire of g. We have $n_{w_2} = v_{w_0} n_{w_1} + v_{w_1} n_{w_0} = x_i \times 0 + K_j \times 1 = K_j$. If $n_{w_2} = 0$, no glitch propagate to the rest of the circuit so $H = 0$. Otherwise, $H \geq n_{w_2} h_g = K_j h_\wedge$. Hence, the adversary deduces $K_j = 1_{H \geq h_\wedge}$.

Q.2 We now consider a special way to compute an AND. Assume we want to compute the AND between a bit A and a bit B. We first represent A and B by two random pairs of bits (v_{a_1}, v_{a_2}) and (v_{b_1}, v_{b_2}) such that $A = v_{a_1} \oplus v_{a_2}$ and $B = v_{b_1} \oplus v_{b_2}$. Then, we evaluate the following formula in a circuit:

$$c_1 = \mathsf{random} \qquad c_2 = (((c_1 \oplus (a_1 \wedge b_1)) \oplus (a_1 \wedge b_2)) \oplus (b_1 \wedge a_2)) \oplus (a_2 \wedge b_2)$$

We thus have a circuit with input wires a_1, a_2, b_1, b_2 and output wires c_1, c_2 and gates as defined by the above formula.

Q.2a Prove that $v_{c_1} \oplus v_{c_2} = A \wedge B$.

We simplify

$$\begin{aligned} v_{c_1} \oplus v_{c_2} &= v_{c_1} \oplus (((v_{c_1} \oplus (v_{a_1} \wedge v_{b_1})) \oplus (v_{a_1} \wedge v_{b_2})) \oplus (v_{b_1} \wedge v_{a_2})) \oplus (v_{a_2} \wedge v_{b_2}) \\ &= (v_{a_1} v_{b_1} + v_{a_1} v_{b_2} + v_{b_1} v_{a_2} + v_{a_2} v_{b_2}) \bmod 2 \\ &= (v_{a_1} + v_{a_2})(v_{b_1} + v_{b_2}) \bmod 2 \\ &= AB \end{aligned}$$

Q.2b Assume that $w_0 = a_1$ in the above circuit. Compute H and prove that the adversary can recover B from H.

Following the above circuit, we split into the following gates:

$$\begin{array}{llll} g_1 = a_1 \wedge b_1 & g_2 = a_1 \wedge b_2 & g_3 = b_1 \wedge a_2 & g_4 = a_2 \wedge b_2 \\ g_5 = c_1 \oplus g_1 & g_6 = g_5 \oplus g_2 & g_7 = g_6 \oplus g_3 & c_2 = g_7 \oplus g_4 \end{array}$$

and we compute $n_{g_1} = v_{b_1}$, $n_{g_2} = v_{b_2}$, $n_{g_3} = n_{g_4} = 0$, $n_{g_5} = v_{b_1}$, *and* $n_{g_6} = n_{g_7} = n_{c_2} = v_{b_1} + v_{b_2}$. *So,*

$$H = (v_{b_1} + v_{b_2})h_\wedge + (4v_{b_1} + 3v_{b_2})h_\oplus$$

If $B = 0$, *we have* $v_{b_1} = v_{b_2}$ *which are random, so* $H = v_{b_1}(2h_\wedge + 7h_\oplus)$. *If* $B = 1$, *we have* $v_{b_2} = 1 - v_{b_1}$ *and* v_{b_1} *random, so* $H = h_\wedge + (3 + v_{b_1})h_\oplus$. *So,* H *can only take 4 different values. The two extreme ones indicate* $B = 0$ *and the two others indicate* $B = 1$.

Q.3 We now represent $A = v_{a_1} \oplus v_{a_2} \oplus v_{a_3}$ and $B = v_{b_1} \oplus v_{b_1} \oplus v_{b_3}$, and take the following circuit

$$\begin{aligned} c_1 &= (a_2 \wedge b_2) \oplus ((a_2 \wedge b_3) \oplus (a_3 \wedge b_2)) \\ c_2 &= (a_3 \wedge b_3) \oplus ((a_1 \wedge b_3) \oplus (a_3 \wedge b_1)) \\ c_3 &= (a_1 \wedge b_1) \oplus ((a_1 \wedge b_2) \oplus (a_2 \wedge b_1)) \end{aligned}$$

Q.3a Prove that $v_{c_1} \oplus v_{c_2} \oplus v_{c_3} = A \wedge B$.

Clearly,

$$\begin{aligned} & v_{c_1} + v_{c_2} + v_{c_3} \\ \equiv\; & v_{a_2}v_{b_2} + v_{a_2}v_{b_3} + v_{a_3}v_{b_2} + v_{a_3}v_{b_3} + v_{a_1}v_{b_3} + v_{a_3}v_{b_1} + v_{a_1}v_{b_1} + v_{a_1}v_{b_2} + v_{a_2}v_{b_1} \\ =\; & (v_{a_1} + v_{a_2} + v_{a_3})(v_{b_1} + v_{b_2} + v_{b_3}) \pmod 2 \\ =\; & AB \end{aligned}$$

Q.3b Assume that $w_0 = a_1$ in the above circuit. Prove that $H = (v_{b_1} + v_{b_2} + v_{b_3})h_\wedge + (v_{b_1} + 2v_{b_2} + 2v_{b_3})h_\oplus$.

We compute $n_{a_i \wedge b_j} = v_{b_j} n_{a_i}$ for all i, j.

$$n_{c_1} = v_{b_2} n_{a_2} + v_{b_3} n_{a_2} + v_{b_2} n_{a_3}$$
$$n_{c_2} = v_{b_3} n_{a_3} + v_{b_3} n_{a_1} + v_{b_1} n_{a_3}$$
$$n_{c_3} = v_{b_1} n_{a_1} + v_{b_2} n_{a_1} + v_{b_1} n_{a_2}$$

We have also three internal XOR. Overall, with $n_{a_1} = 1$ and $n_{a_2} = n_{a_3} = 0$, we obtain

$$\begin{aligned} n_{(a_2 \wedge b_3) \oplus (a_3 \wedge b_2)} &= 0 \\ n_{(a_1 \wedge b_3) \oplus (a_3 \wedge b_1)} &= v_{b_3} \\ n_{(a_1 \wedge b_2) \oplus (a_2 \wedge b_1)} &= v_{b_2} \\ n_{c_1} &= 0 \\ n_{c_2} &= v_{b_3} \\ n_{c_3} &= v_{b_1} + v_{b_2} \end{aligned}$$

hence

$$H = (v_{b_1} + v_{b_2} + v_{b_3}) h_\wedge + (v_{b_1} + 2v_{b_2} + 2v_{b_3}) h_\oplus$$

Q.3c Show that $E(H|B=0) = E(H|B=1)$ so, the expected value of H does not depend on B.

Since $H = (v_{b_1} + v_{b_2} + v_{b_3}) h_\wedge + (v_{b_1} + 2v_{b_2} + 2v_{b_3}) h_\oplus$, by linearity, $E(H|B) = (E(v_{b_1}|B) + E(v_{b_2}|b) + E(v_{b_3}|B)) h_\wedge + (E(v_{b_1}|B) + 2E(v_{b_2}|B) + 2E(v_{b_3}|B)) h_\oplus$. But $E(v_{b_i}|B) = \frac{1}{2}$ for every i and B. So, $E(H|B) = \frac{3}{2} h_\wedge + \frac{5}{2} h_\oplus$. This does not depend on B.

Q.3d We assume that $h_\oplus = 4h_\wedge$. Study the probability distribution of H when $B = 0$ and when $B = 1$ and prove that the adversary can recover B from H.

For $B = 0$, we have the following equiprobable cases:

v_{b_1}	v_{b_2}	v_{b_3}	H	$H/h_\wedge$ with $\mathsf{h}_\oplus = 4\mathsf{h}_\wedge$
0	0	0	0	0
0	1	1	$2h_\wedge + 4h_\oplus$	18
1	0	1	$2h_\wedge + 3h_\oplus$	14
1	1	0	$2h_\wedge + 3h_\oplus$	14

For $B = 1$, we have the following equiprobable cases:

v_{b_1}	v_{b_2}	v_{b_3}	H	$H/h_\wedge$ with $\mathsf{h}_\oplus = 4\mathsf{h}_\wedge$
0	0	1	$1h_\wedge + 2h_\oplus$	5
0	1	0	$1h_\wedge + 2h_\oplus$	5
1	0	0	$1h_\wedge + 1h_\oplus$	4
1	1	1	$3h_\wedge + 5h_\oplus$	13

Clearly, from the value of H, we can see if we are in one case or the other.

2 The Gap Diffie-Hellman Problem

We define three problems: CDH, DDH, and GDH. They are all relative to a public parameters setup scheme $\mathsf{Gen}(1^\lambda) \to \mathsf{pp}$. We assume that pp defines a cyclic group G_{pp} with generator g_{pp} (we assume multiplicative notations) of prime order p_{pp}, and an algorithm to multiply in G_{pp}.

We define three games below. We say the CDH problem is hard if for every PPT algorithm $\mathcal{A}$, $\Pr[\mathsf{CDH}_{\mathcal{A}}(1^\lambda)\ \mathsf{wins}]$ is negligible in the CDH game. We say the DDH problem is hard if for every PPT algorithm $\mathcal{A}$, the advantage

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{DDH}}(\lambda) = \Pr[\mathsf{DDH}_{\mathcal{A}}(1^\lambda, 1) \to 1] - \Pr[\mathsf{DDH}_{\mathcal{A}}(1^\lambda, 0) \to 1]$$

is negligible in the DDH game. We say the $\mathsf{GDH}_{\mathcal{A}}$ problem is hard if for every PPT algorithm $\mathcal{A}$, $\Pr[\mathsf{GDH}_{\mathcal{A}}(1^\lambda)\ \mathsf{wins}]$ is negligible the GDH game. Essentially, the GDH problem is the CDH problem with access to an oracle $\mathcal{O}$ who can tell if a triplet (g^x, g^y, g^z) satisfies $z \equiv xy \pmod{p_{\mathsf{pp}}}$. Namely, $\mathcal{O}(\mathsf{pp}, g^x, g^y, g^z) = 1_{z \equiv xy \pmod{p_{\mathsf{pp}}}}$. We call such $\mathcal{O}$ a *perfect* DDH *oracle*.

$\mathsf{CDH}_{\mathcal{A}}(1^\lambda)$:	$\mathsf{DDH}_{\mathcal{A}}(1^\lambda, b)$:	$\mathsf{GDH}_{\mathcal{A}}(1^\lambda)$:
$\mathsf{Gen}(1^\lambda) \to \mathsf{pp}$ pick $x, y \in \mathbf{Z}_{p_{\mathsf{pp}}}$ uniformly $X \leftarrow g_{\mathsf{pp}}^x$ $Y \leftarrow g_{\mathsf{pp}}^y$ $Z \leftarrow \mathcal{A}(\mathsf{pp}, X, Y)$ win if and only if $Z = g_{\mathsf{pp}}^{xy}$	$\mathsf{Gen}(1^\lambda) \to \mathsf{pp}$ pick $x, y, z \in \mathbf{Z}_{p_{\mathsf{pp}}}$ uniformly if $b = 1$, overwrite $z \leftarrow xy$ $X \leftarrow g_{\mathsf{pp}}^x$ $Y \leftarrow g_{\mathsf{pp}}^y$ $Z \leftarrow g_{\mathsf{pp}}^z$ $b' \leftarrow \mathcal{A}(\mathsf{pp}, X, Y, Z)$ output b'	$\mathsf{Gen}(1^\lambda) \to \mathsf{pp}$ pick $x, y \in \mathbf{Z}_{p_{\mathsf{pp}}}$ uniformly $X \leftarrow g_{\mathsf{pp}}^x$ $Y \leftarrow g_{\mathsf{pp}}^y$ $Z \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{pp}, X, Y)$ win if and only if $Z = g_{\mathsf{pp}}^{xy}$ oracle $\mathcal{O}(\mathsf{pp}, A, B, C)$: compute the discrete logarithm $a \in \mathbf{Z}_{p_{\mathsf{pp}}}$ such that $A = g_{\mathsf{pp}}^a$ $C' \leftarrow B^a$ return $1_{C=C'}$

Q.1 Give an example of a generator Gen with which the DDH problem is easy but the CDH problem is believed to be hard.

Consider $\mathsf{Gen}(1^\lambda)$ *which generates a random prime number p of λ bits then finds a generator g of $\mathbf{Z}_p^*$. We define $G_{\mathsf{pp}} = \mathbf{Z}_p^*$, $g_{\mathsf{pp}} = g$, and $p_{\mathsf{pp}} = p$. We know that the* DDH *problem is easy in this case using the following distinguisher:*

$\mathcal{A}(\mathsf{pp}, X, Y, Z)$:

set $x \in \{0,1\}$ such that $(-1)^x = \left(\frac{X}{p}\right)$

set $y \in \{0,1\}$ such that $(-1)^y = \left(\frac{Y}{p}\right)$

set $z \in \{0,1\}$ such that $(-1)^z = \left(\frac{Z}{p}\right)$

return $1_{z=xy}$

However, the CDH *problem is believed to be hard.*

Q.2 Prove that the GDH problem reduces to the CDH problem (i.e., solving CDH implies solving GDH).

Assuming an oracle $\mathcal{S}$ which solves the CDH *problem, solving the* GDH *problem is trivial: we just define*

$$\mathcal{A}^{\mathcal{O},\mathcal{S}}(\mathsf{pp}, X, Y) = \mathcal{S}(\mathsf{pp}, X, Y)$$

without using $\mathcal{O}$.

Q.3 We let $\mathcal{O}$ be a perfect DDH oracle. We now assume there exists a PPT distinguisher $\mathcal{D}$ such that for any PPT algorithm $\mathcal{G}(\mathsf{pp}) \to (X, Y, Z)$, if we generate $\mathsf{Gen}(1^\lambda) \to \mathsf{pp}$ then $\mathcal{G}(\mathsf{pp}) \to (X, Y, Z)$, then $\mathcal{D}(\mathsf{pp}, X, Y, Z) \to b$, then $b = \mathcal{O}(\mathsf{pp}, X, Y, Z)$ except with negligible probability.

Q.3a Prove that for any PPT algorithm $\mathcal{A}$ with access to an oracle, then running $\mathcal{A}$ with oracle $\mathcal{D}$ or $\mathcal{O}$ and the same random coins produces the same result, except with negligible probability.

Conditioned to that $\mathcal{D}$ returned the same result as $\mathcal{O}$ for the first $i-1$ queries, the ith query is defined by a PPT algorithm $\mathcal{G}$. Due to the previous question, the answer will match the one of $\mathcal{O}$ except with negligible probability. We have a polynomially bounded number of queries. So, by induction, the probability that any query does not match the one of $\mathcal{O}$ is negligible.

Q.3b Under the same assumption that $\mathcal{D}$ exists, prove that the CDH problem is as hard as the GDH problem.

We have proven one reduction in the previous question. We then prove that the CDH *problem reduces to the* GDH *problem.*
Assume that $\mathcal{S}$ is a GDH *oracle, i.e., if $\mathcal{O}$ is a perfect* DDH *oracle, then $\mathcal{S}^{\mathcal{O}}$ is a perfect* CDH *solver. We define $\mathcal{A}(\mathsf{pp}, X, Y) = \mathcal{S}^{\mathcal{D}}(\mathsf{pp}, X, Y)$. Due to the previous question, $\mathcal{A}$ returns the same as $\mathcal{S}^{\mathcal{O}}$, except with negligible probability. So, $1 - \Pr[\mathsf{CDH}_{\mathcal{A}}(1^\lambda)\ \mathsf{wins}]$ is negligible.*

3 Number of Samples to Distinguish Distributions

A *distribution* is a function P from a set $\mathcal{Z}$ to $\mathbf{R}$ such that for all $z \in \mathcal{Z}$, we have $P(z) \geq 0$ and $\sum_{z\in\mathcal{Z}} P(z) = 1$. (We implicitly focus on discrete distributions on finite sets $\mathcal{Z}$.)

Given two distributions P and Q, we define

$$d(P,Q) = \frac{1}{2}\sum_{z\in\mathcal{Z}} |P(z) - Q(z)|$$

as the *statistical distance* between P and Q. We recall that d is a distance, which means that for all distributions P, Q, and R, we have $d(P,Q) \geq 0$, $d(P,Q) = 0$ is equivalent to $P = Q$, $d(P,Q) = d(Q,P)$, and $d(P,R) \leq d(P,Q) + d(Q,R)$. We also define

$$F(P,Q) = \sum_{z\in\mathcal{Z}} \sqrt{P(z)Q(z)}$$

as the *fidelity* between P and Q. The fidelity F is not a distance but $H = \sqrt{1-F}$ is. (This is the *Hellinger distance*.) The statistical distance and the fidelity satisfy the *Fuchs – van de Graaf inequality*

$$1 - F(P,Q) \leq d(P,Q) \leq \sqrt{1 - F(P,Q)^2}$$

Given two distributions P and Q on sets $\mathcal{A}$ and $\mathcal{B}$ respectively, we define a distribution $R = P \otimes Q$ on the set $\mathcal{A} \times \mathcal{B}$ by $R(a,b) = P(a)Q(b)$. We define $P^{\otimes n} = \overbrace{P \otimes \cdots \otimes P}^{n \text{ times}}$.

Q.1 For any distributions P, P', Q, Q', prove that $F(P\otimes P', Q\otimes Q') = F(P,Q)\times F(P',Q')$.

We have

$$\begin{aligned}
F(P\otimes P', Q\otimes Q') &= \sum_{z,z'} \sqrt{P(z)P'(z')Q(z)Q'(z')} \\
&= \sum_{z,z'} \sqrt{P(z)Q(z)}\sqrt{P'(z')Q'(z')} \\
&= \left(\sum_{z} \sqrt{P(z)Q(z)}\right) \times \left(\sum_{z'} \sqrt{P'(z')Q'(z')}\right) \\
&= F(P,Q) \times F(P',Q')
\end{aligned}$$

Q.2 Given a real number $t \in [0,1]$, we let n_t be the minimal number of samples n such that there exists a distinguisher $\mathcal{A}$ using n independent and identically distributed samples to distinguish P from Q such that $\mathsf{Adv}(\mathcal{A}) \geq t$. Prove that for any t, we have

$$\frac{\log(1-t^2)}{2\log F(P,Q)} \leq n_t < 1 + \frac{\log(1-t)}{\log F(P,Q)}$$

By definition of n_t, we have $d(P^{\otimes n_t}, Q^{\otimes n_t}) \geq t$ and $d(P^{\otimes(n_t-1)}, Q^{\otimes(n_t-1)}) < t$. We have

$$t \leq d(P^{\otimes n_t}, Q^{\otimes n_t}) \leq \sqrt{1 - F(P^{\otimes n_t}, Q^{\otimes n_t})^2} = \sqrt{1 - F(P, Q)^{2n_t}}$$

so

$$F(P, Q)^{2n_t} \leq 1 - t^2$$

This shows the lower bound on n_t. Similarly, we have

$$t > d(P^{\otimes(n_t-1)}, Q^{\otimes(n_t-1)}) \geq 1 - F(P^{\otimes(n_t-1)}, Q^{\otimes(n_t-1)}) = 1 - F(P, Q)^{n_t-1}$$

so

$$F(P, Q)^{n_t-1} > 1 - t$$

This shows the upper bound on n_t.

Q.3 Let T be a random process mapping an input $x \in \mathcal{X}$ and some random coins $\rho \in \{0,1\}^*$ to an output $T(x;\rho) \in \mathcal{Y}$. If X follows a distribution P on $\mathcal{X}$, and the random coins ρ are independent and following the uniform distribution, we say that $T(X;\rho)$ follows a distribution P^T on $\mathcal{Y}$. Similarly, a distribution Q on $\mathcal{X}$ induces a distribution Q^T on $\mathcal{Y}$.

Prove that $d(P^T, Q^T) \leq d(P, Q)$.

Given $y \in \mathcal{Y}$, let $I_y = \{(x,\rho) \in \mathcal{X} \times \{0,1\}^*; T(x;\rho) = y\}$. We have

$$\begin{aligned}
d(P^T, Q^T) &= \frac{1}{2} \sum_{y \in \mathcal{Y}} \left| \sum_{(x,\rho) \in I_y} (P(x) \Pr[\rho] - Q(x) \Pr[\rho]) \right| \\
&= \frac{1}{2} \sum_{y \in \mathcal{Y}} \left| \sum_{(x,\rho) \in I_y} (P(x) - Q(x)) \Pr[\rho] \right| \\
&\leq \frac{1}{2} \sum_{y \in \mathcal{Y}} \sum_{(x,\rho) \in I_y} |P(x) - Q(x)| \Pr[\rho] \\
&= \frac{1}{2} \sum_{x \in \mathcal{X}} \sum_{\rho \in \{0,1\}^*} |P(x) - Q(x)| \Pr[\rho] \\
&= \frac{1}{2} \sum_{x \in \mathcal{X}} |P(x) - Q(x)| \\
&= d(P, Q)
\end{aligned}$$

Another way to prove this is to use the equivalence between the statistical distance and the advantage of the best distinguisher limited to one sample. We have $d(P^T, Q^T) = \mathsf{Adv}(\mathcal{A})$ for some distinguisher $\mathcal{A}(Y)$ who gets one sample Y and produce a bit. Let $X \in \mathcal{X}$. We define $\mathcal{B}(X)$ as follows:

1: *pick ρ at random*
2: *output $\mathcal{A}(T(X;\rho))$*

Clearly, $\mathcal{B}$ is a distinguisher between P and Q and has advantage $\mathsf{Adv}(\mathcal{B}) = \mathsf{Adv}(\mathcal{A})$. Hence,

$$d(P^T, Q^T) = \mathsf{Adv}(\mathcal{A}) = \mathsf{Adv}(\mathcal{B}) \leq d(P, Q)$$

Q.4 Use the previous question to prove that $d(P \otimes P', Q \otimes Q') \leq d(P, Q) + d(P', Q')$.

HINT: use first the triangular inequality $d(P \otimes P', Q \otimes Q') \leq d(P \otimes P', Q \otimes P') + d(Q \otimes P', Q \otimes Q')$.

Since d is a distance, we can use the triangular inequality $d(P \otimes P', Q \otimes Q') \leq d(P \otimes P', Q \otimes P') + d(Q \otimes P', Q \otimes Q')$. Next, we show that $d(P \otimes P', Q \otimes P') \leq d(P, Q)$ and $d(Q \otimes P', Q \otimes Q') \leq d(P', Q')$ by the same technique. We show it only for the first one.
To show $d(P \otimes P', Q \otimes P') \leq d(P, Q)$, we use a sampling algorithm $G(\rho)$ which converts random coins ρ into a random variable following the distribution P'. Then, we define $T(z; \rho) = (z, G(\rho))$. We observe that $P^T = P \otimes P'$ and $Q^T = Q \otimes P'$. So, $d(P \otimes P', Q \otimes P') = d(P^T, Q^T) \leq d(P, Q)$, following the previous question.

Q.5 With the notations from Q.2, deduce that $n_t \geq \frac{t}{d(P,Q)}$.

We have

$$t \leq d(P^{\otimes n_t}, Q^{\otimes n_t}) \leq n_t \times d(P, Q)$$