Problem 1: Classical time vaults

Below, I describe four candidates for (non-revocable) time vaults. Each of them has at least one serious drawback for use in our setting.

Think about all kind of things: practicality (does it have some features that may make it impractical to use), security (can we break it, perhaps with small but non-negligible probability?), predictability of adversary speed (the adversary should not be able to decrypt the vault much faster just because he pays 1000 times more for his computing hardware).

(a) **Rivest-Shamir-Wagner.**

Producing the time vault for message $m$: Pick $n = pq$ as the product of two large primes $p, q$. Compute $k := 2^{2T} \mod n$. Output $TV(m) := (T, n, Enc(k, m))$ where $Enc$ is a suitable symmetric encryption scheme. Note that $k$ can be computed quickly as follows: $k = 2^{2T} \mod n = 2^{2T} \mod (p-1)(q-1) \mod n$. The latter needs only $\log T + \log n \ll T$ modular exponentiations when using square-and-multiply.

Decrypting $TV(m) = (T, n, c)$: Compute $k := 2^{2T} \mod n$ by squaring $2T$ times. (This takes time $T$ if we count one multiplication as one time step.) Compute $m := Dec(k, c)$.

(b) **Iterated hash.**

Producing the time vault for message $m$: Pick a random $x$. Compute $k := H(H(\ldots H(x) \ldots))$ ($T$ applications of $H$) for a suitable hash function $H$. Output $TV(m) := (T, x, Enc(k, m))$.

Decrypting $TV(m) = (T, x, c)$: Compute $k := H(H(\ldots H(x) \ldots))$ ($T$ applications of $H$). Compute $m := Dec(k, c)$.

(c) **Hash preimage.**

Producing the time vault for message $m$: Pick $r \overset{\$}{\leftarrow} \{1, \ldots, T\}$, and pick bitstrings $x, y \in \{0, 1\}^\eta$ (where $\eta$ is the security parameter). Let $z := H(x||r)$ where $H$ is a suitable hash function and $||$ denotes concatenation. Let $k := H(y||r)$. Output $TV(m) := (T, x, y, z, Enc(k, m))$.

Decrypting $TV(m) = (T, x, y, z, c)$: Try all $r = 1, \ldots, T$ until $H(x||r) = z$. With this value $r$, compute $k := H(y||r)$. Compute $m := Dec(k, c)$.

(d) **Multiple hash preimage.**

Producing the time vault for message $m$: Pick $r_i \overset{\$}{\leftarrow} \{1, \ldots, T/\eta\}$ for $i = 1, \ldots, \eta$, and pick bitstrings $x_i, y \in \{0, 1\}^\eta$ for $i = 1, \ldots, \eta$. Let $z_i := H(x_i || r_i)$. (So far, this
is like \( \eta \) instances of the preceding time vault.) Let \( k := H(y\|r_1\|\ldots\|r_\eta) \). Output \( TV(m) := (T, x_1, \ldots, x_\eta, y, z_1, \ldots, z_\eta, Enc(k, m)) \).

Decrypting \( TV(m) = (T, x_1, \ldots, x_\eta, y, z_1, \ldots, z_\eta, Enc(k, m)) \): For each \( i = 1, \ldots, \eta \), try all \( r_i = 1, \ldots, T/\eta \) until \( H(x_i\|r_i) = z_i \). With these values \( r_i \), compute \( k := H(y\|r_1\|\ldots\|r_\eta) \). Compute \( m := Dec(k, c) \).

**Problem 2: The Bell test in the quantum time vault**

In the proof of [Lemma 24](#) in the lecture notes (security of \( QTV \)), the final step of the proof was to analyze the following game:

**Game 4 (Using fake TV)**

(a) \( V \leftarrow TV(0) \).

(b) Initialize \( XY \) with \( |\beta_{00}\rangle^{\otimes \eta} \).

(c) Adversary \( B^* \) gets \( V, X \).

(d) \( B^* \leftarrow \{+\}, \times \}^\eta \).

(e) Apply measurement \( P_B^+ \) to \( XY \). \( ok := 1 \) if measurement succeeds.

(f) Apply measurement \( P_B^{Bell} \) to \( XY \). \( tErr := 1 \) if measurement succeeds.

In the lecture notes, it says that \( \Pr[tErr = 0 \land ok = 1] \) is negligible.

Show that this is indeed the case.

**Hint:** You should look at the analysis of the Bell test in the lectures 2012-10-30 and 2012-10-31.

(a) Show that if the state before step (d) of Game 4 would be \( \rho = |\tilde{x}\rangle\langle\tilde{x}| \otimes \rho_B \) for some \( x \in \{0,1\}^{2\eta} \), then \( \Pr[tErr = 0 \land ok = 1 : \rho] \leq 2^{-t-1} \) if \( \omega(x) > t \) and \( \Pr[tErr = 0 \land ok = 1 : \rho] = 0 \) if \( \omega(x) \leq t \). (Here \( \omega(x) \) denotes the number of non-00-bitpairs in \( x \). And \( \Pr[E : \rho] \) denotes the probability that \( E \) holds when \( \rho \) is the state before step (d) in Game 4.)

**Hint:** You may use the fact that \( P_+ = P_{bf} \) and \( P_\times = P_{pf} \) where \( P_+ := |00\rangle\langle00| + |11\rangle\langle11| \) is the measurement operator measuring whether two qubits are equal in the computational basis \(+\), and \( P_\times := |++\rangle\langle++| + |--\rangle\langle--| \) is the measurement operator measuring whether two qubits are equal in the computational basis, and

\[
\begin{align*}
P_{bf} &:= |\beta_{00}\rangle\langle\beta_{00}| + |\beta_{10}\rangle\langle\beta_{10}|, \\
P_{pf} &:= |\beta_{00}\rangle\langle\beta_{00}| + |\beta_{01}\rangle\langle\beta_{01}|.
\end{align*}
\]

(b) Show that if the state before step (d) of Game 4 would be \( \rho = \sum_x p_x |\tilde{x}\rangle\langle\tilde{x}| \times \rho_B^x \) (with \( \sum p_x = 1 \)), then \( \Pr[tErr = 0 \land ok = 1 : \rho] \leq 2^{-t-1} \).

(c) Show that \( \Pr[tErr = 0 \land ok = 1] \leq 2^{-t-1} \).