Intuition from Khurana's presentation | Certified Deletion

Story: One can "lift" the secret sharing construction to obtain encryption w/ certified deletion w/ certified deletion.

: First, let's see how one defines such a scheme, then we see a construction & finally, outline how the proof works. } Semi-formal

: In the next "section" we formally state & prove this

Def. Def^n: Encryption w/ certified deletion:

A tuple of algorithms

(Gen, Enc, Dec, Delete, Verify)

(NB: I use Del & Ver for brevity)

w/ the following syntax.

$(vk, ct) \leftarrow Enc_{pk}(b)$

$(|ct_1\rangle, ct_2) \leftarrow Dec_{sk}(ct)$

m

$v \leftarrow Del(ct)$

$\top/\bot \leftarrow Ver(v, vk)$

acc/rej

(where everything except $|ct_1\rangle$ is classical.)

is an encryption scheme w/ certified deletion

if
- (Gen, Enc, Dec) is a valid encryption scheme,

  i.e. (Gen, Enc) satisfy IND security &
  $m = Dec_{sk}(Enc_{pk}(m))$ where $(pk, sk) \leftarrow Gen(1^\lambda)$.
- it holds that

$$TD(Exp_0(1^\lambda), Exp_1(1^\lambda)) \leq negl(1^\lambda)$$

where $Exp_b(1^\lambda)$ denotes the density matrix output by the following game/experiment.

Challenger		Adversary
$pk, sk \leftarrow Gen(1^\lambda)$		
$(vk, ct) \leftarrow Enc_{pk}(b)$	pk ct →	
	← V, ρ	ρ: leftover state held by the adversary, after it produces V.
if $1 = Ver(vk, V)$ output ρ.		

Else, output $\perp$.

& also

$$1 = Ver(del_{pk}(Enc_{pk}(b)))$$

Construction:

Suppose (Gen, Enc, Dec) is any encryption scheme satisfying IND security (or better).

Define $Gen'(1^\lambda) := Gen(1^\lambda)$

$Enc'_{pk}(b) :=$ Sample $\theta \leftarrow \{0,1\}^\lambda$
$x \leftarrow \{0,1\}^\lambda$
$b' := b \oplus Parity_{\theta=0}(x)$

$ct := \left(H^\theta |x\rangle, Enc_{pk}(\theta, b')\right)$
$vk := (\theta, x)$

$Dec'_{sk}(ct) :=$ Parse $ct = (|\psi\rangle, s)$
Decrypt s using $Dec_{sk}(s) = (\tilde{\theta}, \tilde{b}')$
Apply $H^{\tilde{\theta}}$ to $|\psi\rangle$ & measure to get $\tilde{x}$,
& use that to recover the bit as $\tilde{b} := \tilde{b}' \oplus Parity_{\tilde{\theta}=0}(\tilde{x})$
return $\tilde{b}$

$Del(ct) :=$ Parse $ct = (|\psi\rangle, s)$
& return $\tilde{x} \leftarrow$ "Measure $|\psi\rangle$ in Hadamard"

$Ver_{vk}(cert) :=$ Check if cert matches x when restricted to $\{i : \theta_i = 1\}$.

NB: Verification key vk is private in this scheme.

Idea behind the security proof

step 0: Write down the security game using our construction

Exp_b

Challenger

$x \leftarrow \{0,1\}^\lambda$

$\theta \leftarrow \{0,1\}^\lambda$

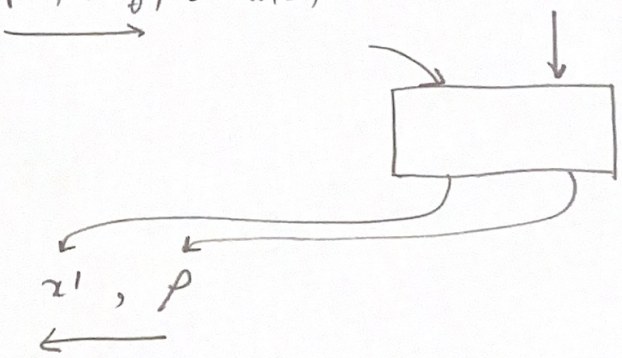
$pk, sk \leftarrow Gen(1^\lambda)$

$b' = b \oplus parity_{\theta=0}(x)$

Adversary (σ)

$\dots \mu^\theta |x\rangle$

$pk, |x\rangle_\theta, Enc(\theta, b')$



x', ρ

if $\left[\forall i \text{ s.t. } \theta_i = 1, \; x_i = x_i'\right]$ then output ρ

Else output $\perp$

Claim: $Exp_0 \overset{??}{\approx} Exp_1$

(want to prove)

Story: The difficulty here is that b is computationally hidden & we want to make a statistical statement

So want to remove the b dependence, or at least defer it, for as long as we can.

Idea 1: Defer the "d" dependence.

Exp_b'

Challenger — Adversary (σ)

$\theta, x \leftarrow \{0,1\}^\lambda$

$\tilde{b} \leftarrow \{0,1\}$

$pk, sk \leftarrow Gen(1^\lambda)$

$b' := \tilde{b} \oplus \underset{\theta=0}{Parity}(x)$

$pk, |x\rangle_\theta, Enc(\theta, b')$ →

← x', ρ

If $\left[\begin{array}{l} \tilde{b} = b \\ \wedge \\ \forall i \text{ s.t. } \theta_i = 1 \\ \quad x_i = x_i' \end{array}\right]$ then output ρ

Else, output $\perp$

Claim: $TD[Exp_0', Exp_1'] = \frac{1}{2} TD[Exp_0, Exp_1]$

Story: So it suffices to show that $TD[Exp_0', Exp_1']$ is negligible

Idea 2: Purify the experiment (so we don't have to choose

Exp_b^3

Challenger

Adversary

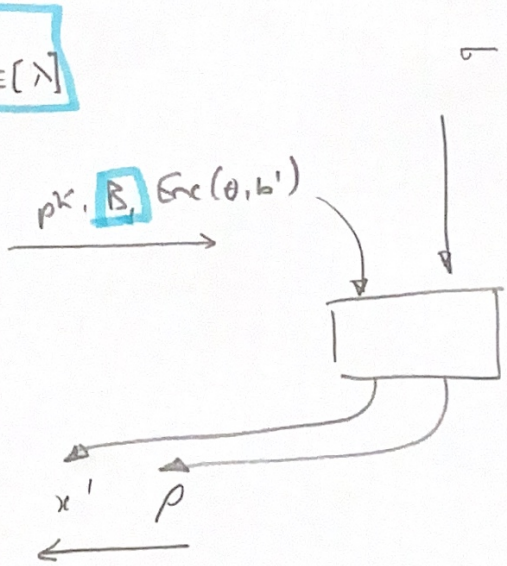
$sk, pk \leftarrow Gen(1^\lambda)$

$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$ on $\{A_i B_i\}_{i \in [\lambda]}$

$b' \leftarrow \{0,1\}$

$\theta \leftarrow \{0,1\}^\lambda$

$pk, B, Enc(\theta, b')$



Measure A_i in basis θ_i to get x_i

Compute $\tilde{b} = b' \oplus Parity_{\theta=0}(x)$

If $\tilde{b} \neq b$, abort (i.e. $\perp$)

Else $\begin{cases} \text{If } [\forall i \text{ s.t. } \theta_i = 1, x_i = x_i'] \text{ output } \rho \\ \text{Else output } \perp \end{cases}$

Claim: It suffices to prove $TD[Exp_0^3, Exp_1^3] = negl(\lambda)$.

Exp_b^q

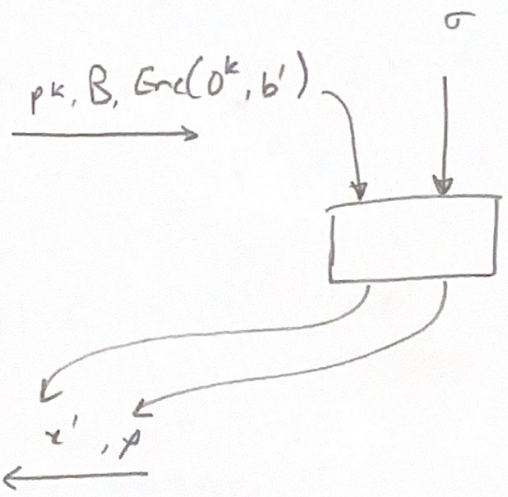
Challenger | Adversary

$sk, pk \leftarrow Gen(1^\lambda)$

$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$ on $\{A_i, B_i\}_{i \in [\lambda]}$

$b' \leftarrow \{0,1\}$

$\theta \leftarrow \{0,1\}^k$



Measure A_i in basis θ_i to get x_i

Compute $\tilde{b} = b' \oplus Parity_{\theta=0}(x)$

If $\tilde{b} \neq b$ abort

Else [If [$\forall i$ s.t. $\theta_i = 1$, $x_i = x_i'$] output ρ ; Else output $\perp$]

Claim: $TD[Exp_0^q, Exp_1^q] = negl$

Story. Using prior tools, one can show that the claim above holds.

The key remaining difficulty is showing that

Exp^5 & Exp^4 are close in TD.

What is the difficulty?

(a) Want to reason after everything is done

(b) But in the end, there's statistical security while we used computational security in the preceding step.

How does one avoid this problem?

I. Using an "efficiently checkable predicate" on the state of the challenger.

& argue that this predicate cannot depend on b — else the encryption scheme breaks.

II Once this is done, it's "easy" to show

$Exp^4_0 \approx_{TD} Exp^4_1$, & using I,

$Exp^5_0 \approx_{TD} Exp^5_1$.