1 Definitions of Leakages

- We have considered two definitions of leakage.
- The first notion requires the simulator to produce the same view (encrypted database) as the real encryption scheme given only the leakage.

$\underline{\text{Real}_{\mathcal{A}}(n)}$

1 : $\mathsf{k} \leftarrow_{\$} \mathsf{KGen}(1^n)$
2 : $DB \leftarrow \mathcal{A}(1^n)$
3 : $EDB \leftarrow \mathsf{Enc}(1^n, \mathsf{k}, DB)$
4 : **return** EDB

$\underline{\text{Sim}_{\mathcal{A},\mathcal{S}}(n)}$

1 : $\mathsf{k} \leftarrow_{\$} \mathsf{KGen}(1^n)$
2 : $DB \leftarrow \mathcal{A}(1^n)$
3 : $l \leftarrow \mathcal{L}_{\mathcal{M}}(1^n, \mathsf{k}, DB)$
4 : $EDB \leftarrow \mathcal{S}(1^n, l)$
5 : **return** EDB

- The second notion uses an additional adversary to produce some information on the real encrypted database and the simulated one. The information produced is required to be indistinguishable. There is also a slight difference where the first adversary generates some state together with the database.

$\underline{\text{Real}_{\mathcal{A}}(n)}$

1 : $\mathsf{k} \leftarrow_{\$} \mathsf{KGen}(1^n)$
2 : $(DB, \mathsf{st}) \leftarrow \mathcal{A}_1(1^n)$
3 : $EDB \leftarrow \mathsf{Enc}(1^n, \mathsf{k}, DB)$
4 : $info \leftarrow \mathcal{A}_2(1^n, EDB, \mathsf{st})$
5 : **return** $info$

$\underline{\text{Sim}_{\mathcal{A},\mathcal{S}}(n)}$

1 : $\mathsf{k} \leftarrow_{\$} \mathsf{KGen}(1^n)$
2 : $(DB, \mathsf{st}) \leftarrow \mathcal{A}_1(1^n)$
3 : $l \leftarrow \mathcal{L}_{\mathcal{M}}(1^n, \mathsf{k}, DB)$
4 : $EDB \leftarrow \mathcal{S}(1^n, l)$
5 : $info \leftarrow \mathcal{A}_2(1^n, EDB, \mathsf{st})$
6 : **return** $info$

- We show the two definitions are in fact equivalent under quantifications 'there exists a simulator, for all adversaries'. i.e. if there is a distinguisher for one notion, there is a distinguisher for the other.

2 Proof of Equivalence between the two Notions

- We call the first notion `notion1` and second notion `notion2`. It is easy to see that indistinguishability in `notion2` implies that in `notion1` for free by letting $\mathcal{A}_2$ return its second argument. So we will prove implication in the other direction.
- Suppose there is a distinguisher for `notion2`, then the distinguisher must work for $\mathcal{A}_1$ that returns a constant state, and $\mathcal{A}_2$ that returns its second argument. So the distinguisher can break `notion1` with non-negligible probability.