No poker face for computers

If a machine contains secret data (bank code, secret communication key...), one can learn about them as soon as they influence the outputs of a program executed on this machine. Preventing the existence of critical leaks is the goal of cryptography.

But computers are like us: thinking produces unintentional behaviours (computation time, temperature, power consumption...) that may reveal sensible information! Exploiting these behaviours is called a side-channel attack.

The way of efficient mind reading

**Aggregation**

The important question is whether sensible knowledge can be aggregated across multiple observations: how many “questions” (and which ones) should be asked to the target to guess its secret? (from the observation of its reactions)

**Decomposition**

In general, secrets are too complex to be extractable at once. To obtain realistic attacks, a divide-and-conquer approach is usually required: one isolates, within the whole observation, the influence of small parts of the secret to recover it little by little.

We formalise side-channel attacks using execution time (with noise in observations). When a program can be decomposed into n independent blocks, we prove that:

1. one can guess all leakable information from a number of well-chosen questions proportional to n...
2. ... and that $O(n \ln \frac{n}{\epsilon})$ random questions are actually sufficient (with probability $1 - \epsilon$).

What we bring

Boris Köpf and Itsaka Rakotonirina, Systematic Design of Timing Attacks, 2016-2017