


Information-Theoretic approaches to Information Flow

Catuscia Palamidessi
INRIA Saclay & Ecole Polytechnique

based on joint work with
Mário S. Alvim and Miguel E. Andrés


The problem

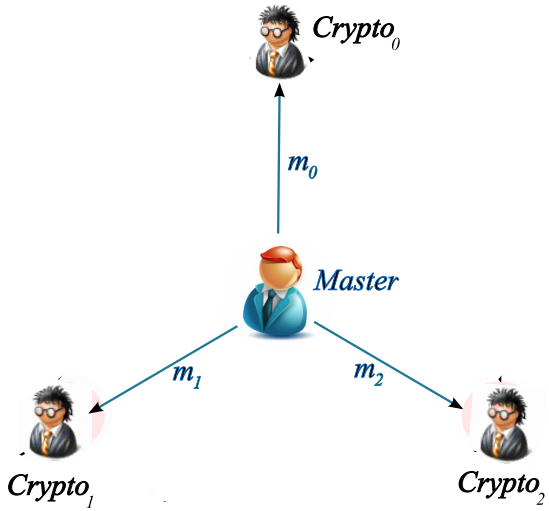
Control the **information leakage**

i.e.

the amount of **secret information** that an adversary can **infer** from what he can **observe**

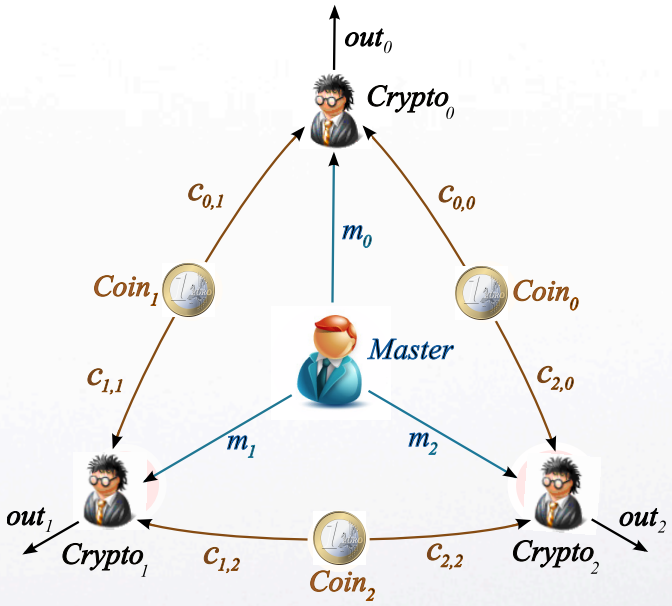
An example to illustrate the problem: The Dining Cryptographers (Chaum, 1988)

- Three cryptographers have a dinner
- Their master informs each of them separately whether he should pay for the (whole) bill or not. If none of them pays, the master will pay
- The cryptographers are allowed to try to find out whether the master has asked one of them to pay, **but they should not know whom**



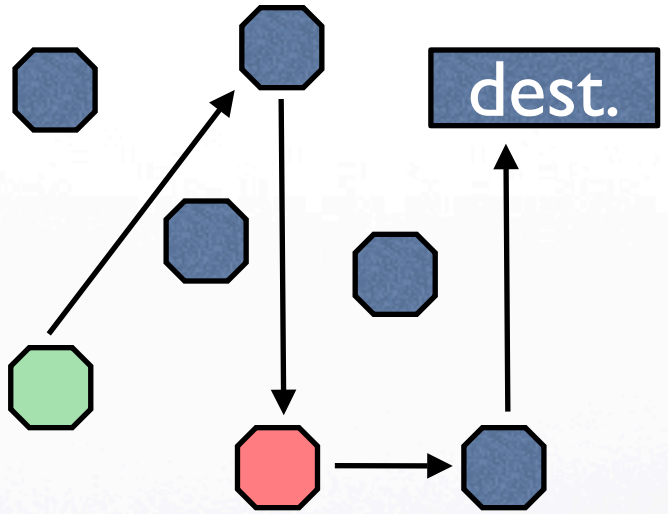
Dining Cryptographers: The solution proposed by Chaum

- Place a binary coin between each two cryptographers and toss them
- Each cryptographer makes the binary sum of the adjacent coins. The payer (if any) adds 1. The results are announced
- The binary sum of the results is 1 iff one of them is a payer
- If the coins are fair, we have perfect anonymity



Example: Crowds (Rubin and Reiter'98)

- Problem: A user (initiator) wants to send a message anonymously to another user (dest.)
- Crowds: A group of n users who agree to participate in the protocol.
- The initiator selects randomly another user (forwarder) and forwards the request to him
- A forwarder randomly decides whether to send the message to another forwarder or to dest.
- ... and so on



Probable innocence: under certain conditions, an attacker who intercepts the message from x cannot attribute more than 0.5 probability to x to be the initiator

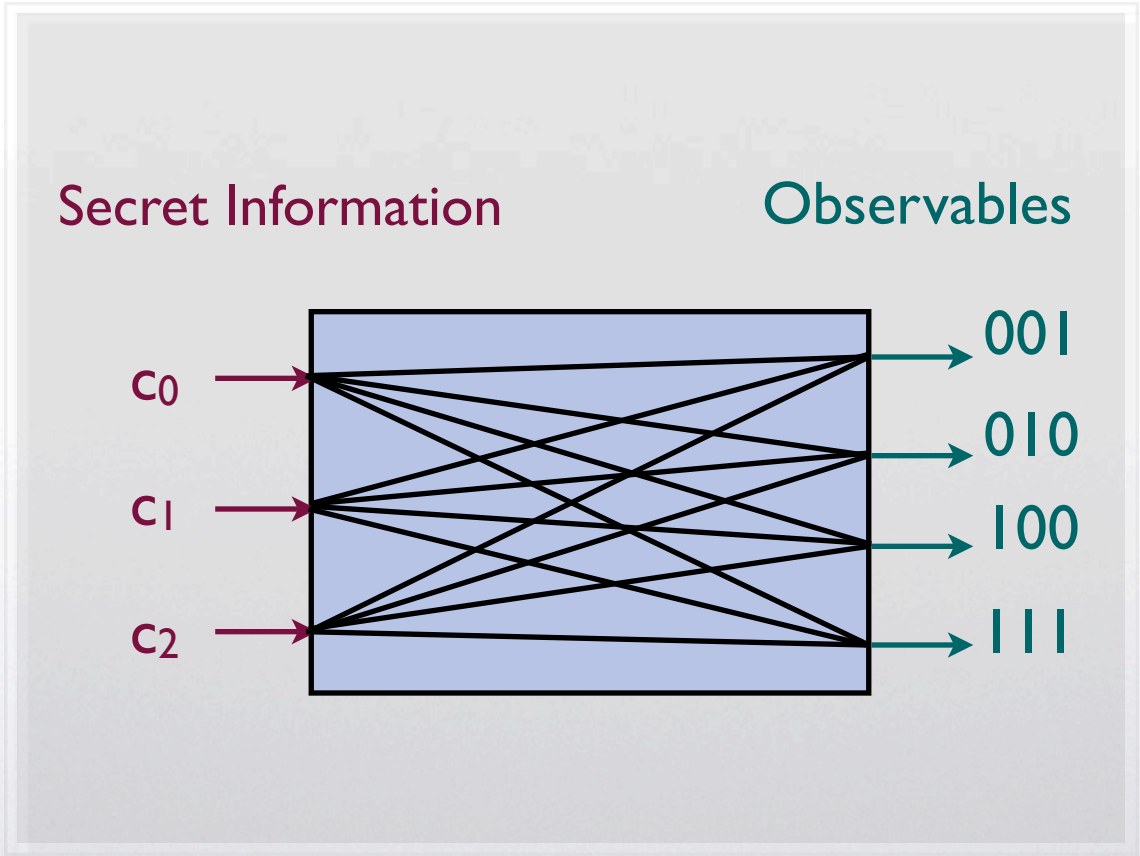

Our problem: Formalize the notion of information leakage

- No agreement on the subject. (Here we present our proposal.)
- There is not even agreement on the true-false notions:
  - **Perfect anonymity:** my favorite notion is the one by Chaum: for each observation, the *a posteriori probability* that c_i is the payer is the same as the *a priori probability*
  - **Probable innocence:** Reiter and Rubin defined it only informally and other researchers got it wrong
- We are interested in a **quantitative** notion, i.e. how much information does the system leak


Common features in Information Flow

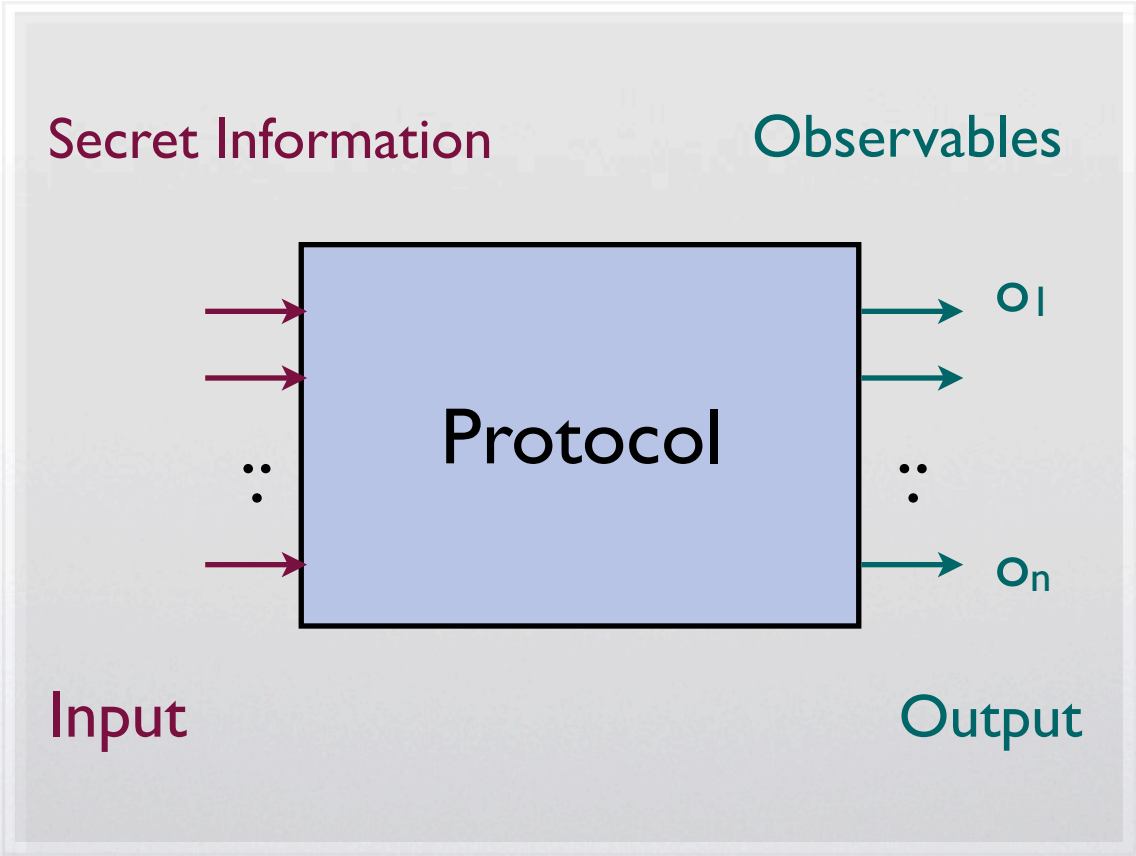
- There is information that we want to keep secret
  - the payer in DC
  - the initiator in Crowds
- There is information that is revealed (observables)
  - the declarations in DC
  - the users who forward messages to a corrupted user in Crowds
- The value of the secret information may be chosen probabilistically, and the system may use randomization (maybe even in purpose, to hide the link between secrets and observables)
  - coin tossing in DC
  - random forwarding to another user in Crowds

Example: Dining Cryptographers



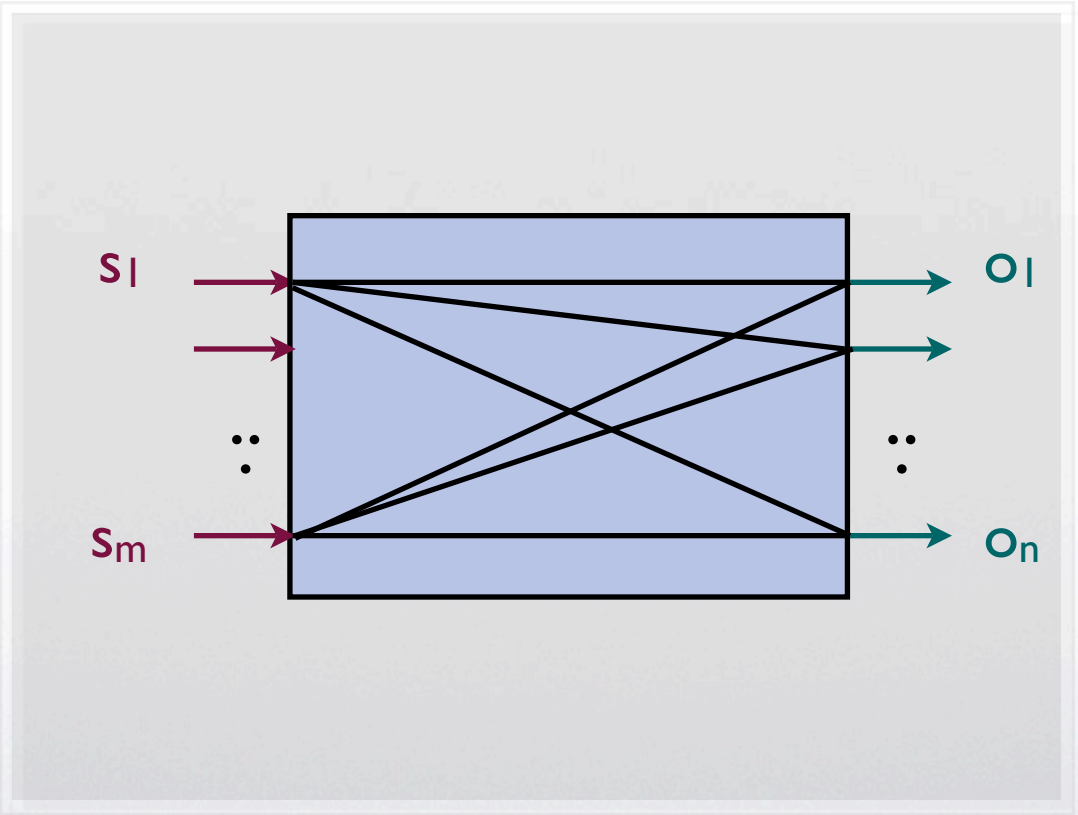
An intriguing analogy:

Systems as Information-Theoretic channels



Information-Theoretic channels are **noisy** channels:

- an input can generate different outputs (according to a prob. distr.)
- an output can be generated by different inputs (even in **det. syst.**)



$p(o_j|s_i)$: the conditional probability to observe o_j given that the secret is s_i


Towards a quantitative def. of leakage

- A general principle (on which most people agree):

> Leakage = a priori uncertainty − a posteriori uncertainty

- But what is ``uncertainty''? (and here people disagree)
- Our answer is that there is no unique answer: it depends on
  - the model of attack, and
  - how we measure it success


Uncertainty, this unknown

- Kopf and Basin model of attack: assume an oracle who answers yes/no to questions of a certain form. The attack is then defined by the form of the questions
- **Example 1:** The questions are of the form "is $S \in P$?", and the measure of success is: the expected number of questions of this kind needed to determine the value of S

then

uncertainty corresponds to Shannon entropy

- For instance, guessing the last bit of a password


Uncertainty, this unknown

- **Example 2:** The questions are of the form "is S = v ?", and the measure of success is: the probability of determining the value of S with just one try

then

uncertainty corresponds to Renyi's min entropy

- For instance, guessing a password by trying it

- In any case, leakage can be modeled as mutual information:

$$I(S ; O) = H(S) - H(S \mid O)$$

Computing the leakage by model checking e.g. reachability analysis

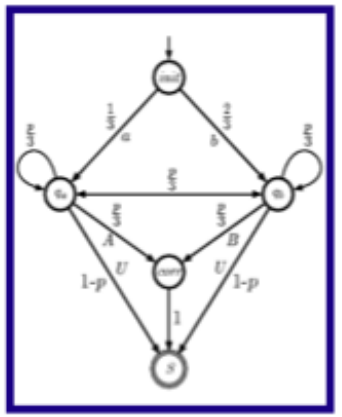
- **Example**

$$
\begin{array}{lll}
x^{aA}_{init} = \frac{1}{3}\cdot x^{A}_{q_a}, & x^{A}_{q_a} = \frac{p}{3}\cdot x^{A}_{q_a} + \frac{p}{3}\cdot x^{A}_{q_b} + \frac{p}{3}\cdot x^{\epsilon}_{corr}, & x^{A}_{corr} = x^{A}_{S},\\
x^{bA}_{init} = \frac{2}{3}\cdot x^{A}_{q_b}, & x^{A}_{q_b} = \frac{p}{3}\cdot x^{A}_{q_a} + \frac{p}{3}\cdot x^{A}_{q_b} + \frac{p}{3}\cdot x^{A}_{corr}, & x^{A}_{S} = 0,\\
x^{aB}_{init} = \frac{1}{3}\cdot x^{B}_{q_a}, & x^{B}_{q_a} = \frac{p}{3}\cdot x^{B}_{q_a} + \frac{p}{3}\cdot x^{B}_{q_b} + \frac{p}{3}\cdot x^{B}_{corr}, & x^{B}_{corr} = x^{B}_{S},\\
x^{bB}_{init} = \frac{2}{3}\cdot x^{B}_{q_b}, & x^{B}_{q_b} = \frac{p}{3}\cdot x^{B}_{q_a} + \frac{p}{3}\cdot x^{B}_{q_b} + \frac{p}{3}\cdot x^{\epsilon}_{corr}, & x^{B}_{S} = 0,\\
x^{aU}_{init} = \frac{1}{3}\cdot x^{U}_{q_a}, & x^{U}_{q_a} = \frac{p}{3}\cdot x^{U}_{q_a} + \frac{p}{3}\cdot x^{U}_{q_b} + (1-p)\cdot x^{\epsilon}_{S}, & x^{\epsilon}_{corr} = x^{\epsilon}_{S},\\
x^{bU}_{init} = \frac{2}{3}\cdot x^{U}_{q_b}, & x^{U}_{q_b} = \frac{p}{3}\cdot x^{U}_{q_a} + \frac{p}{3}\cdot x^{U}_{q_b} + (1-p)\cdot x^{\epsilon}_{S}, & x^{\epsilon}_{S} = 1.
\end{array}
$$

$$
x^{aA}_{init} = \frac{7}{40}, \quad x^{aB}_{init} = \frac{3}{40}, \quad x^{aU}_{init} = \frac{1}{12},
$$

$$
x^{bA}_{init} = \frac{3}{20}, \quad x^{bB}_{init} = \frac{7}{20}, \quad x^{bU}_{init} = \frac{1}{6}.
$$

Solution



Crowds as a probabilistic automaton

	A	B	U
a	7/40	3/40	1/12
b	3/20	7/20	1/6

Matrix of joint probabilities

- **Complexity**
  - $O\left((|obs| \times |Q|)^3\right)$ In general
  - $O\left(|obs| \times |Q|^3\right)$ Some Scenarios (e.g observables at the end)


A digression on something that I find rather puzzling


Possibilistic approach

- Very popular, ‘cause it is simpler than the quantitative approaches
- Key principle: A system P has no leakage iff:
  For every pair of secret values a, b, P[a] “is equivalent” to P[b]
  - Uhu ???
    - It assumes that the scheduler “helps”
    - Problem with refinement

Example: Consider the following system

$$S \stackrel{\text{def}}{=} (c, out)(A \parallel H_1 \parallel H_2 \parallel Corr),$$

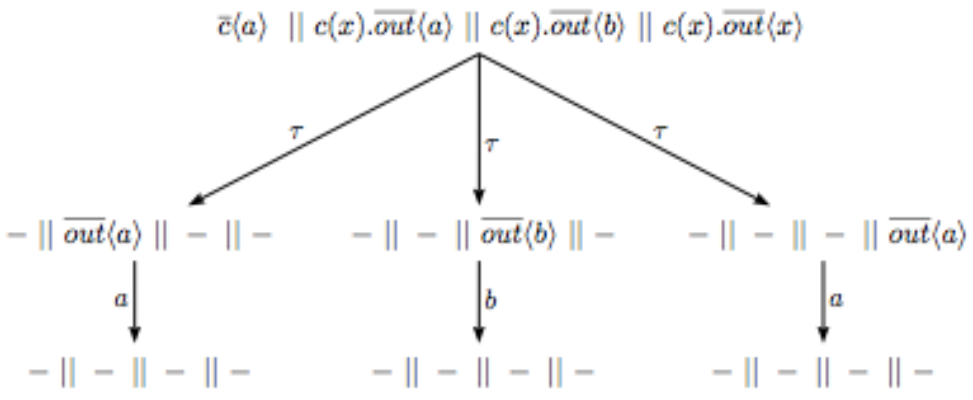
$$A \stackrel{\text{def}}{=} \bar{c}\langle sec \rangle, \quad H_1 \stackrel{\text{def}}{=} c(s).\overline{out}\langle a \rangle, \quad H_2 \stackrel{\text{def}}{=} c(s).\overline{out}\langle b \rangle, \quad Corr \stackrel{\text{def}}{=} c(s).\overline{out}\langle s \rangle$$

- *S*[*a*/*sec*] and *S*[*b*/*sec*] are bisimilar, so the system should have no leakage
- But: nondeterminism in concurrency is meant as underspecification
  - Some schedulers may always select *Corr* first
  - Standard implementation refinement (simulation) preserves properties of individual runs, but no-leakage is expressed as a global property.
- This problem is actually well known. (My understanding of) the main proposals to solve it are based on changing the notion of refinement: bisimulation instead than simulation. The actual implementation would be probabilistic, but it would be viewed as nondeterministic in order to prove bisimulation

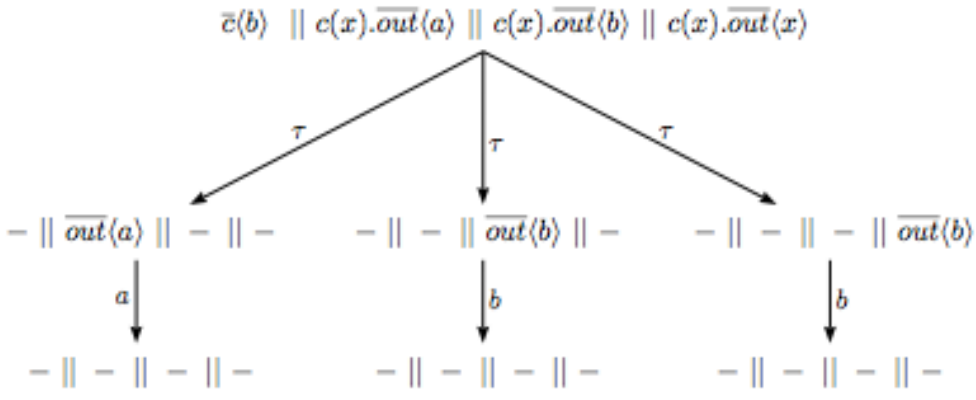
$$S \stackrel{\text{def}}{=} (c, out)(A \parallel H_1 \parallel H_2 \parallel Corr),$$

$$A \stackrel{\text{def}}{=} \overline{c}\langle sec \rangle, \quad H_1 \stackrel{\text{def}}{=} c(s).\overline{out}\langle a \rangle, \quad H_2 \stackrel{\text{def}}{=} c(s).\overline{out}\langle b \rangle, \quad Corr \stackrel{\text{def}}{=} c(s).\overline{out}\langle s \rangle$$

S[*a*/sec]



S[*b*/sec]




Thank you !