Advanced Tools from Modern Cryptography

Lecture 7

Basics: Computational Indistinguishability

Security Definitions

- So far: Perfect secrecy
  - Achieved in Shamir secret-sharing, passive BGW and passive GMW (given a trusted party for OT)
- But for 2PC using Yao's Garbled circuit (even given a trusted party for OT) security only against computationally bounded adversary
  - We haven't defined such security yet!
- Plan
  - Computational Indistinguishability (Today)
  - Simulation-based security (Next time)

Because, the obvious definition obtained by replacing perfect secrecy by computational secrecy turns out to be slightly weak

Recall

Relaxing Secrecy Requirement

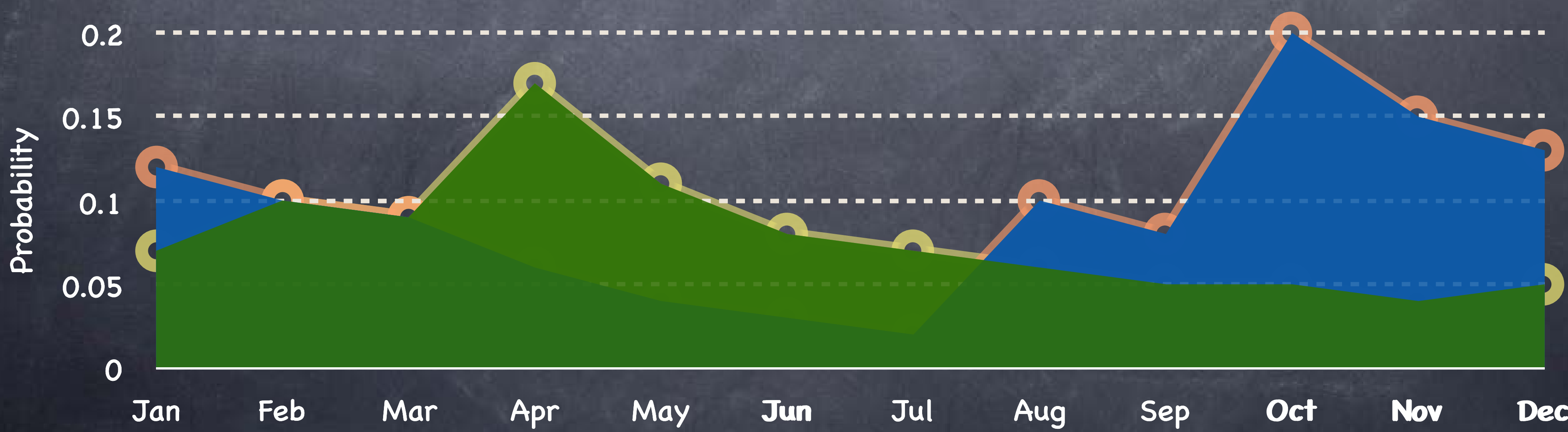
- When view is not exactly independent of the message
  - Next best: view close to a distribution that is independent of the message
  - Two notions of closeness: Statistical and Computational

Recall

a.k.a. Statistical Distance or Total Variation Distance

Statistical Difference

- Given two distributions A and B over the same sample space, how well can a test T distinguish between them?
  - T given a single sample drawn from A or B
  - How differently does it behave in the two cases?
- $\Delta(A,B) := \max_T \mid Pr_{x \leftarrow A}[T(x)=1] - Pr_{x \leftarrow B}[T(x)=1] \mid$



Indistinguishability

- Two distributions are **statistically indistinguishable** from each other if the statistical difference between them is "negligible"
- What is negligible? 2^{-20} ? 2^{-40} ? 2^{-80} ? Let the "user" decide!
- Security guarantees will be given asymptotically as a function of the **security parameter**
  - A knob that can be used to set the security level
- Given $\{A_k\}$, $\{B_k\}$, $\Delta(A_k,B_k)$ is a function of the security parameter k
- Negligible: reduces "very quickly" as the knob is turned up
  - "Very quickly": quicker than 1/poly for any polynomial poly
    - So that if negligible for one sample, remains negligible for polynomially many samples

Indistinguishability

- Distribution ensembles $\{A_k\}$, $\{B_k\}$ are **statistically indistinguishable** if $\exists$ negligible $\nu(k)$ s.t. $\Delta(A_k,B_k) \leq \nu(k)$
  - $\Delta(A_k,B_k) := \max_T | Pr_{x\leftarrow A_k}[T(x)=1] - Pr_{x\leftarrow B_k}[T(x)=1] |$
  - $\nu(k)$ is said to be **negligible** if $\forall\ d \geq 0$, $\exists$ N s.t. $\forall$ k>N, $\nu(k) < 1/k^d$
- Can rewrite as: $\forall$ tests T, $\exists$ negligible $\nu(k)$ s.t.

  $| Pr_{x\leftarrow A_k}[T(x)=1] - Pr_{x\leftarrow B_k}[T(x)=1] | \leq \nu(k)$

  In particular, the best test

- Distribution ensembles $\{A_k\}$, $\{B_k\}$ **computationally indistinguishable** if $\forall$ "efficient" tests T, $\exists$ negligible $\nu(k)$ s.t.

  $| Pr_{x\leftarrow A_k}[T(x)=1] - Pr_{x\leftarrow B_k}[T(x)=1] | \leq \nu(k)$

Indistinguishability

- Distribution ensembles $\{A_k\}$, $\{B_k\}$ **computationally indistinguishable** if $\forall$ "efficient" tests T, $\exists$ negligible $\nu(k)$ s.t.

  $$| \Pr_{x \leftarrow A_k}[T(x)=1] - \Pr_{x \leftarrow B_k}[T(x)=1] | \leq \nu(k)$$

$A_k \approx B_k$

Indistinguishability

- Distribution ensembles $\{A_k\}$, $\{B_k\}$ **computationally indistinguishable** if ∀ "efficient" tests T, ∃ negligible $\nu(k)$ s.t.

  $$| \Pr_{x \leftarrow A_k}[T(x)=1] - \Pr_{x \leftarrow B_k}[T(x)=1] | \leq \nu(k)$$

  $A_k \approx B_k$

- Efficient: Probabilistic Polynomial Time (PPT)

  Non-Uniform

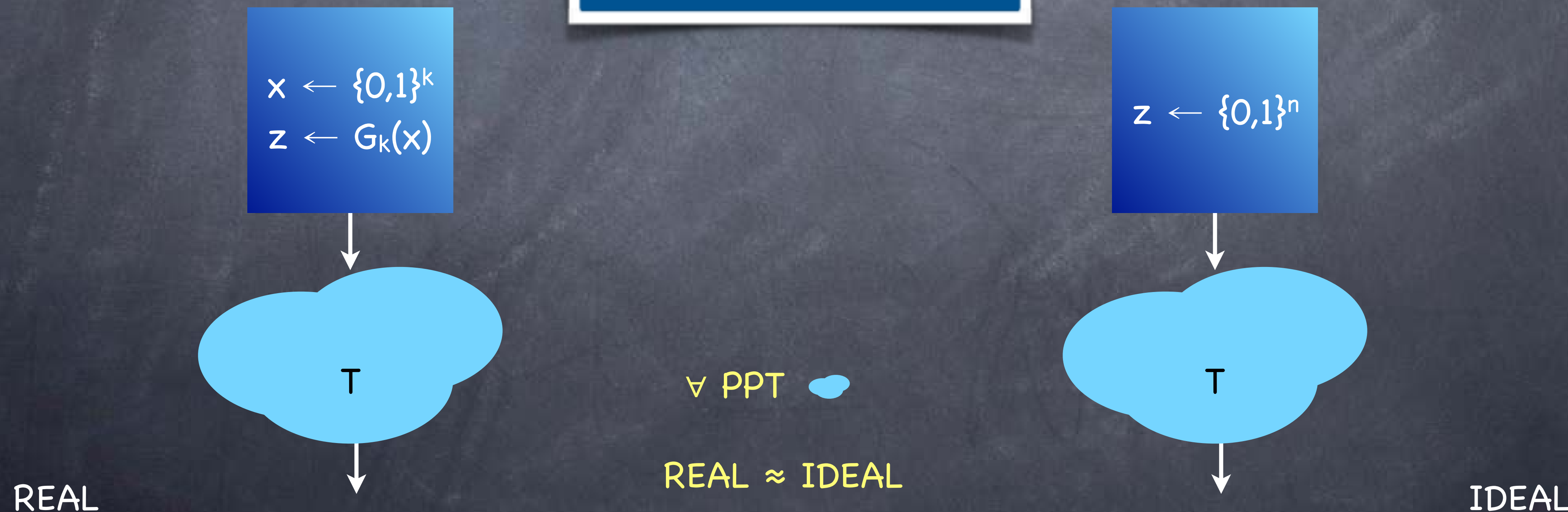
  - PPT T: a family of randomised programs T_k (one for each value of the security parameter k), s.t. there is a polynomial p with each T_k running for at most p(k) time
  - (Could restrict to uniform PPT. But by default, we'll allow non-uniform.)

Example: Pseudorandomness Generator (PRG)

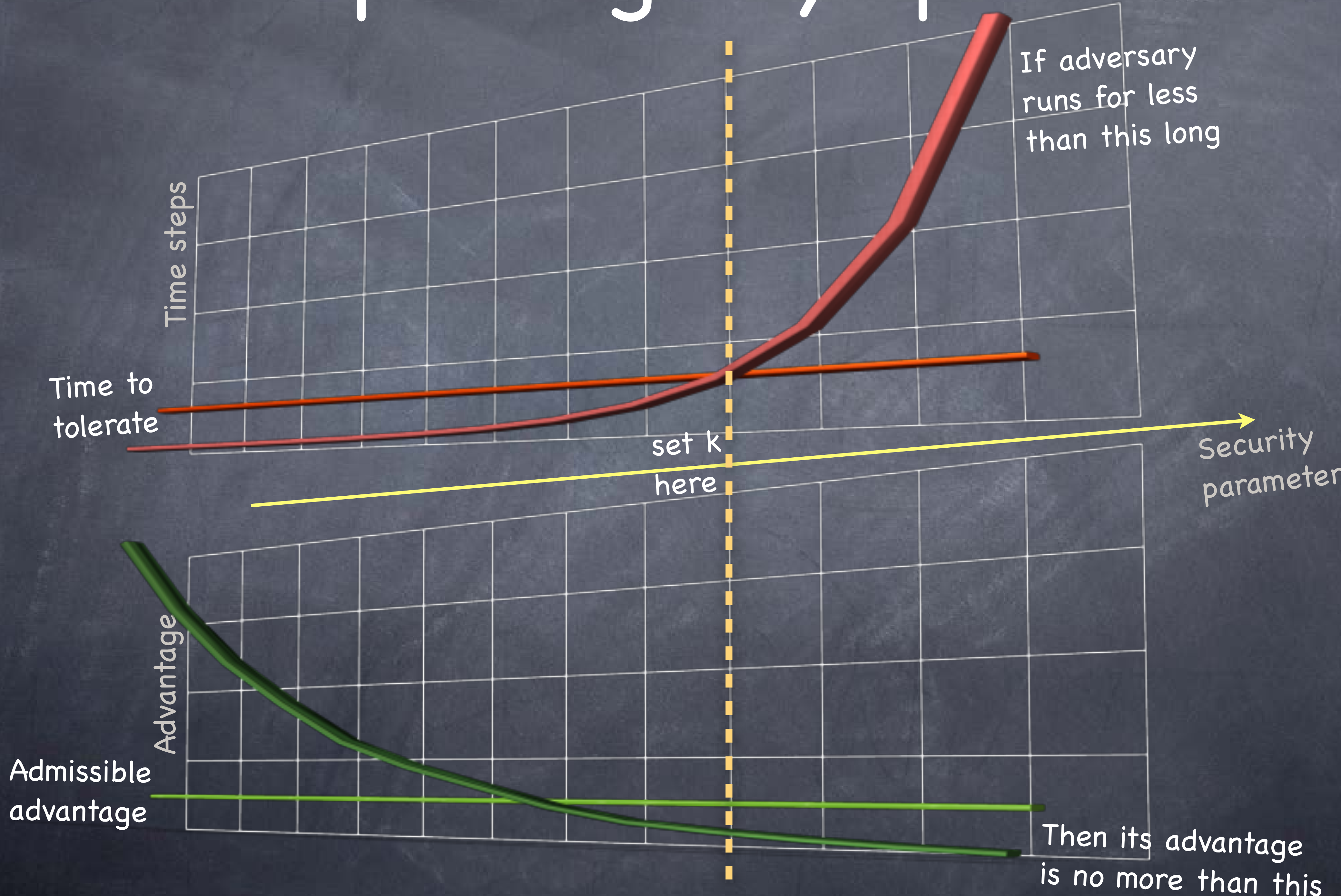
- Takes a short seed and (deterministically) outputs a long string
  - $G_k: \{0,1\}^k \rightarrow \{0,1\}^{n(k)}$ where $n(k) > k$
- Security definition: Output distribution induced by random input seed should be "pseudorandom"
  - i.e., Computationally indistinguishable from uniformly random
  - $\boxed{\{G_k(x)\}_{x \leftarrow \{0,1\}^k} \approx U_{n(k)}}$
  - Note: $\{G_k(x)\}_{x \leftarrow \{0,1\}^k}$ cannot be statistically indistinguishable from $U_{n(k)}$ unless $n(k) \leq k$ (Why?)
    - i.e., no non-trivial PRG against unbounded adversaries

Example: Pseudorandomness Generator (PRG)

- Takes a short seed and (deterministically) outputs a long string
  - $G_k: \{0,1\}^k \rightarrow \{0,1\}^{n(k)}$ where $n(k) > k$
- Security definition: $\boxed{\{G_k(x)\}_{x \leftarrow \{0,1\}^k} \approx U_{n(k)}}$



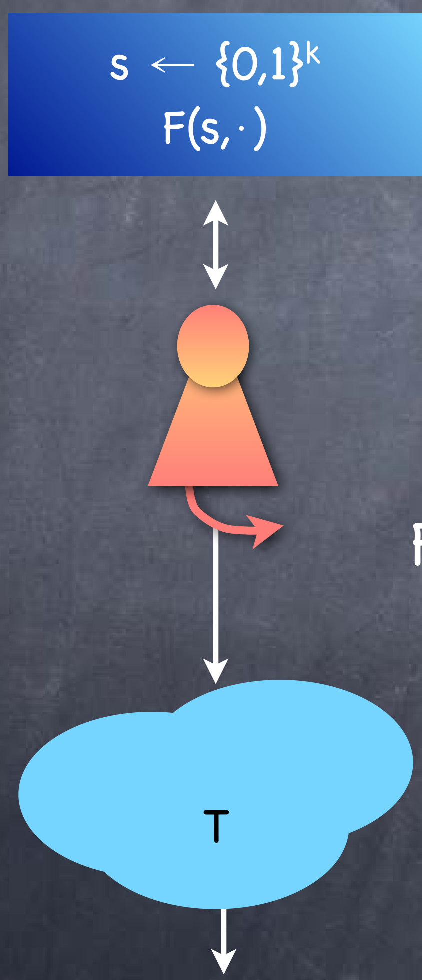
Interpreting Asymptotics



Pseudorandom Function (PRF)

- A compact representation of an exponentially long (pseudorandom) string
  - Allows "random-access" (instead of just sequential access)
    - A function F(s;i) outputs the i^{th} block of the pseudorandom string corresponding to seed s
    - Exponentially many blocks (i.e., large domain for i)
- Pseudorandom Function
  - Need to define pseudorandomness for a function (not a string)
  - Idea: the view of an adversary <u>arbitrarily interacting with the function</u> is indistinguishable from its view when interacting with a random function

Pseudorandom Function (PRF)



REAL

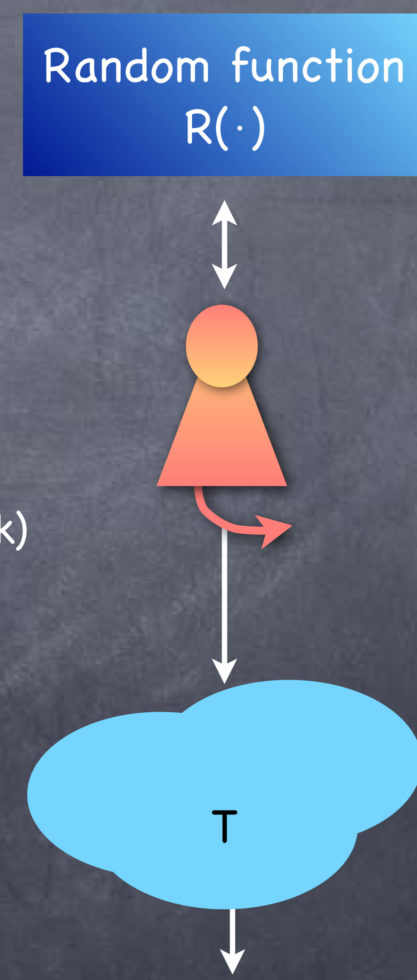
F: $\{0,1\}^k \times \{0,1\}^{m(k)} \to \{0,1\}^{n(k)}$

is a PRF if

$\forall$ PPT 

$\forall$ PPT 

REAL ≈ IDEAL



IDEAL

Security for MPC

- Recall: For passive security, secrecy is all the matters
- For a 2-party functionality f, with only Bob getting the output, perfect secrecy against corrupt Bob:
  i.e., $\forall$ x, x′, y s.t., f(x,y) = f(x′,y′), $\text{view}_{\text{Bob}}(x,y) = \text{view}_{\text{Bob}}(x',y)$
  - In particular, if (y, f(x,y)) uniquely determines x (i.e., if $f(x',y)=f(x,y) \Rightarrow x'=x$), then OK for view to reveal x
- In the computational setting, just replace = with ≈ ?
  - We should ask for more!
  - E.g., f is a decryption algorithm, with key x and ciphertext y
  - Often, a (long enough) ciphertext and message uniquely determines the key
    - But not OK to reveal the key to Bob!

Because,
uniquely determines
≠ reveals!