Delegatable PRFs and Applications

Aggelos Kiayias

Nikos Triandopoulos

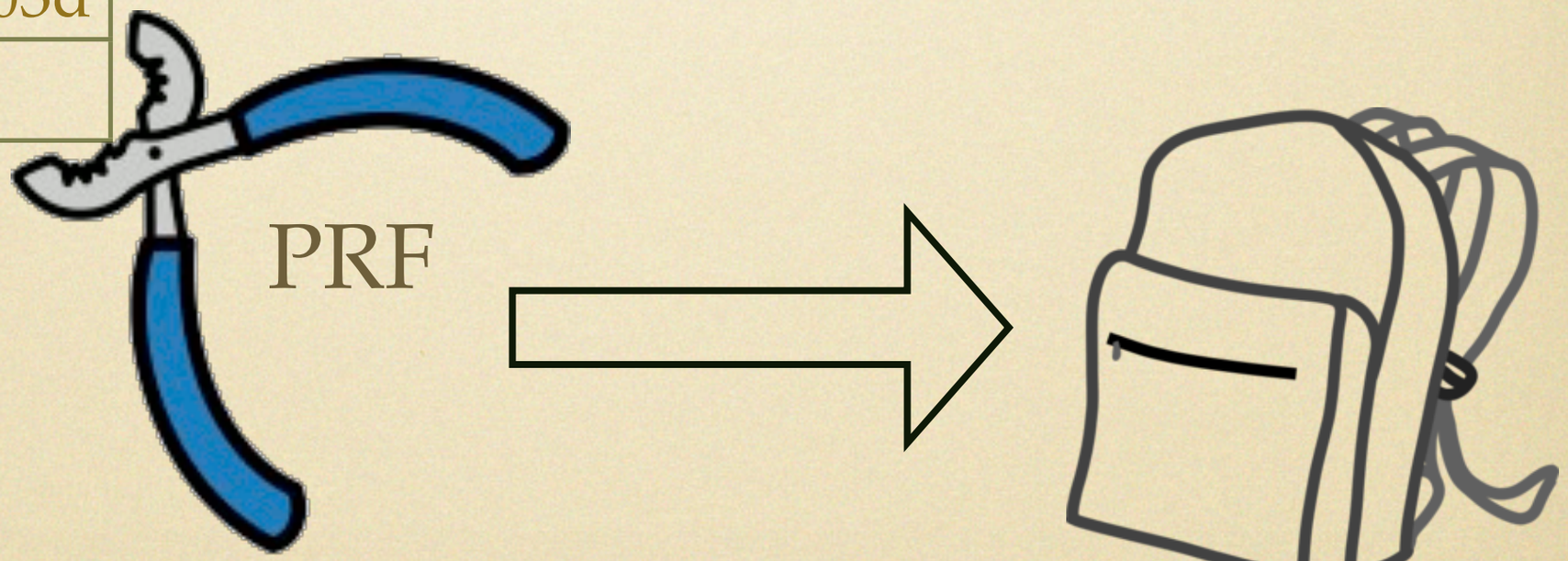
Stavros Papadopoulos

Thomas Zacharias

Pseudorandom Function

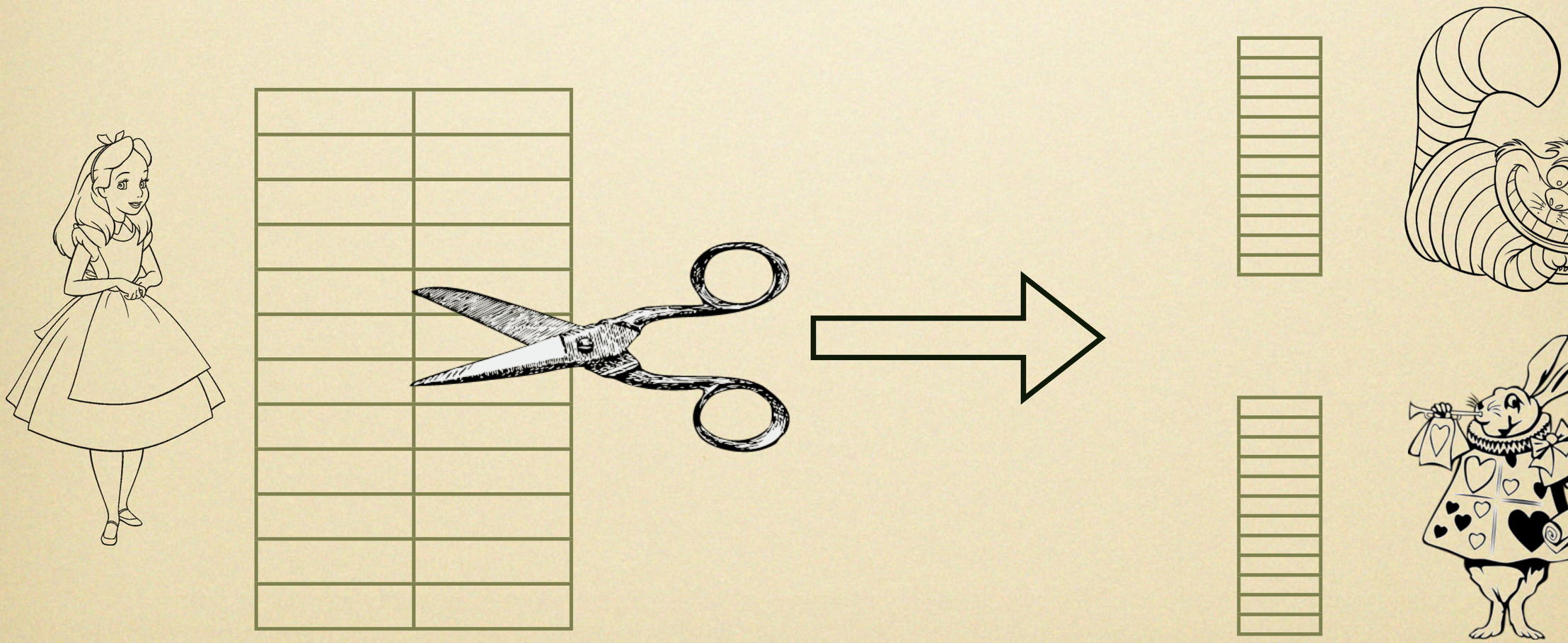
0	3adc619
1	a380232
2	1d406e7
3	5bcc03d
…	…

fits an exponential table of random entries into just polynomial space (in the eyes of a poly observer)



Delegating a function?

- motivation : if a function is in table form we can arbitrarily cut and distribute it on any subsets of its domain:



Delegating a PRF?

how to cut and distribute the PRF 'backpack' in arbitrary ways?

Delegating a PRF?

how to cut and distribute the PRF 'backpack' in arbitrary ways?

Delegatable PRFs

$\langle \mathcal{F}, T, C \rangle$ with respect to policy set $\mathcal{P}$

$\mathcal{F} = \{f_k \mid k \in \{0,1\}^\lambda\}$ PRF family

$T(k, P) \to \tau$ Trapdoor generation

$C(\tau) \to \{f_k(x) \mid x \in P\}$ Delegated set reconstruction

$|\tau| \ll |P|$ fundamental efficiency objective

PRF security

$\mathcal{A}^{f_k(\cdot), T(k,\cdot)}$

Policy Privacy

$\tau_P \approx \tau_{P'}$

Results

- Simple observation: GGM construction provides delegation w.r.t. prefix policies.
- Designing DPRF's becomes substantially more challenging for wider policies.
- We do 1D range policies (under general assumptions + policy privacy).

Many Applications

- Efficient **batch** searchable symmetric encryption.
- Broadcast encryption.
- RFid authentication.
- ....

Delegatable PRFs and Applications

http://eprint.iacr.org/2013/379

[to appear in ACM-CCS 2013]

Aggelos Kiayias

Stavros Papadopoulos

Nikos Triandopoulos

Thomas Zacharias