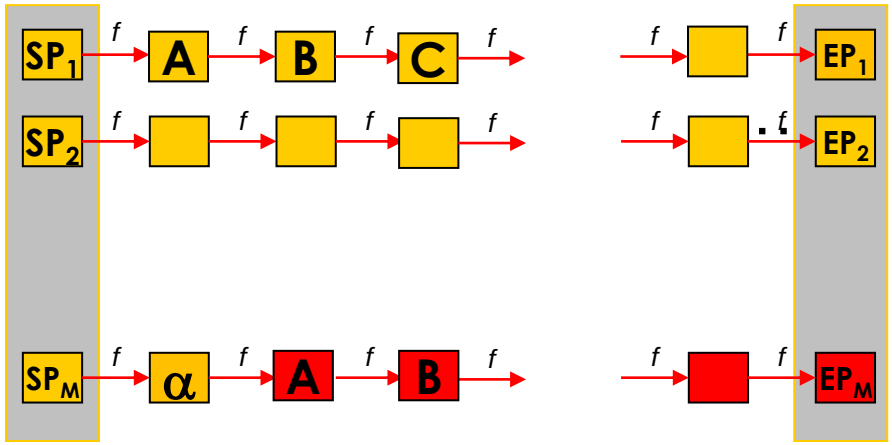
Perfect Hellman Tables using Bloom Filters

Meltem Sönmez Turan
NIST, meltem.turan@nist.gov
Crypto 2012 Rump Session
August 21, 2012

Hellman's TMTO Attack [Hellman, 1980]

Goal: Invert a one way function *f*

Domain size of *f* is **N** = **M** rows × **T** columns× **R** tables



Random Case: SPs are selected randomly

⇒ # distinct elements in R tables is ≈ 0.55 N

⇒ Success prob. ≈ 0.55.

Perfect Table: Ensure all distinct elements per table

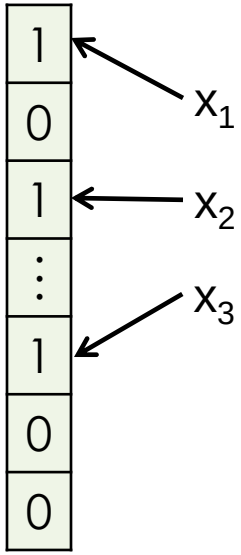
⇒ Success prob. ≈ 0.63.

Perfect Hellman Tables

How? Add row $[x_1, x_2, \ldots, x_T]$, only if all elements are new.

Pr (false positive) = 0

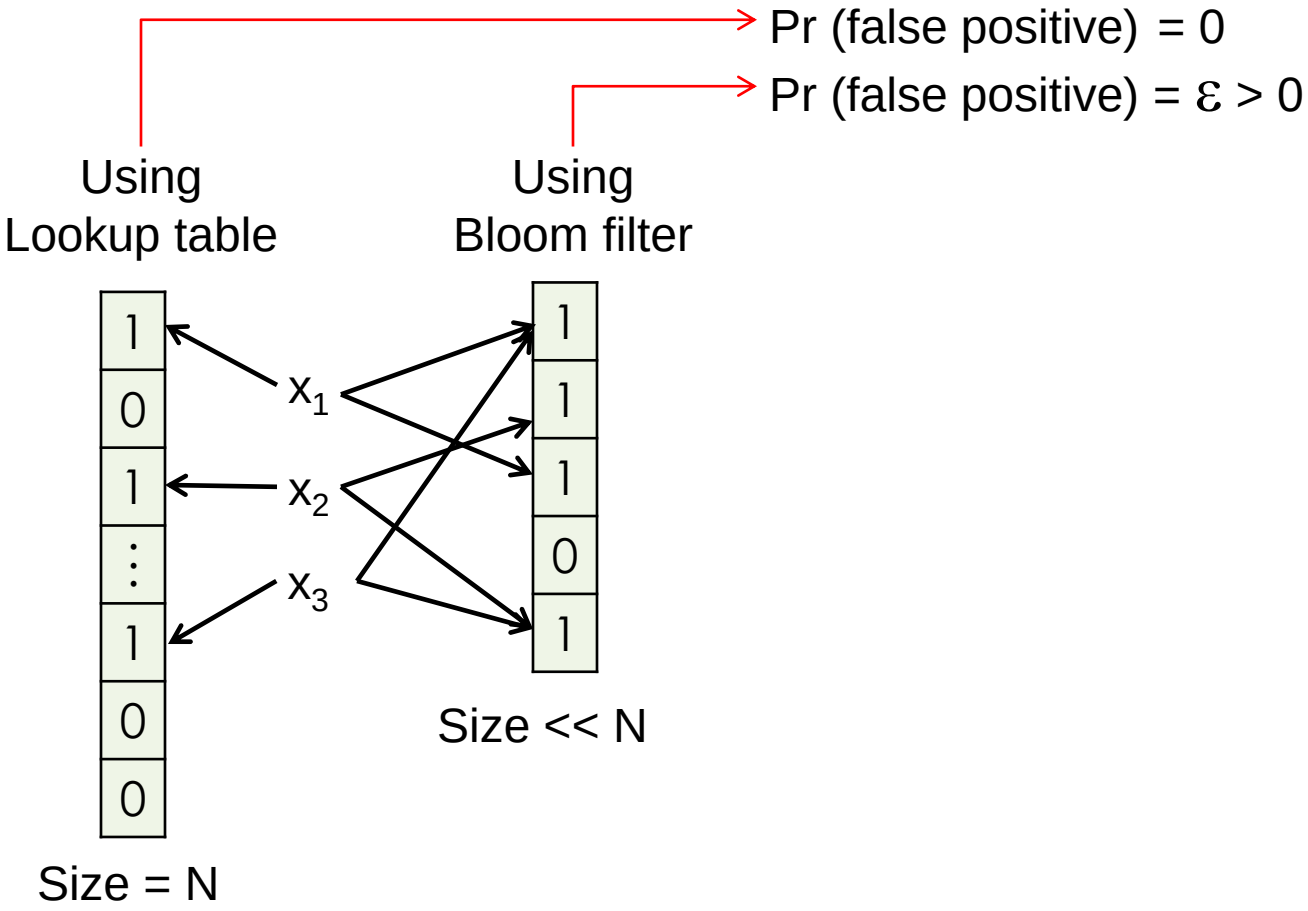
Using
Lookup table



Size = N

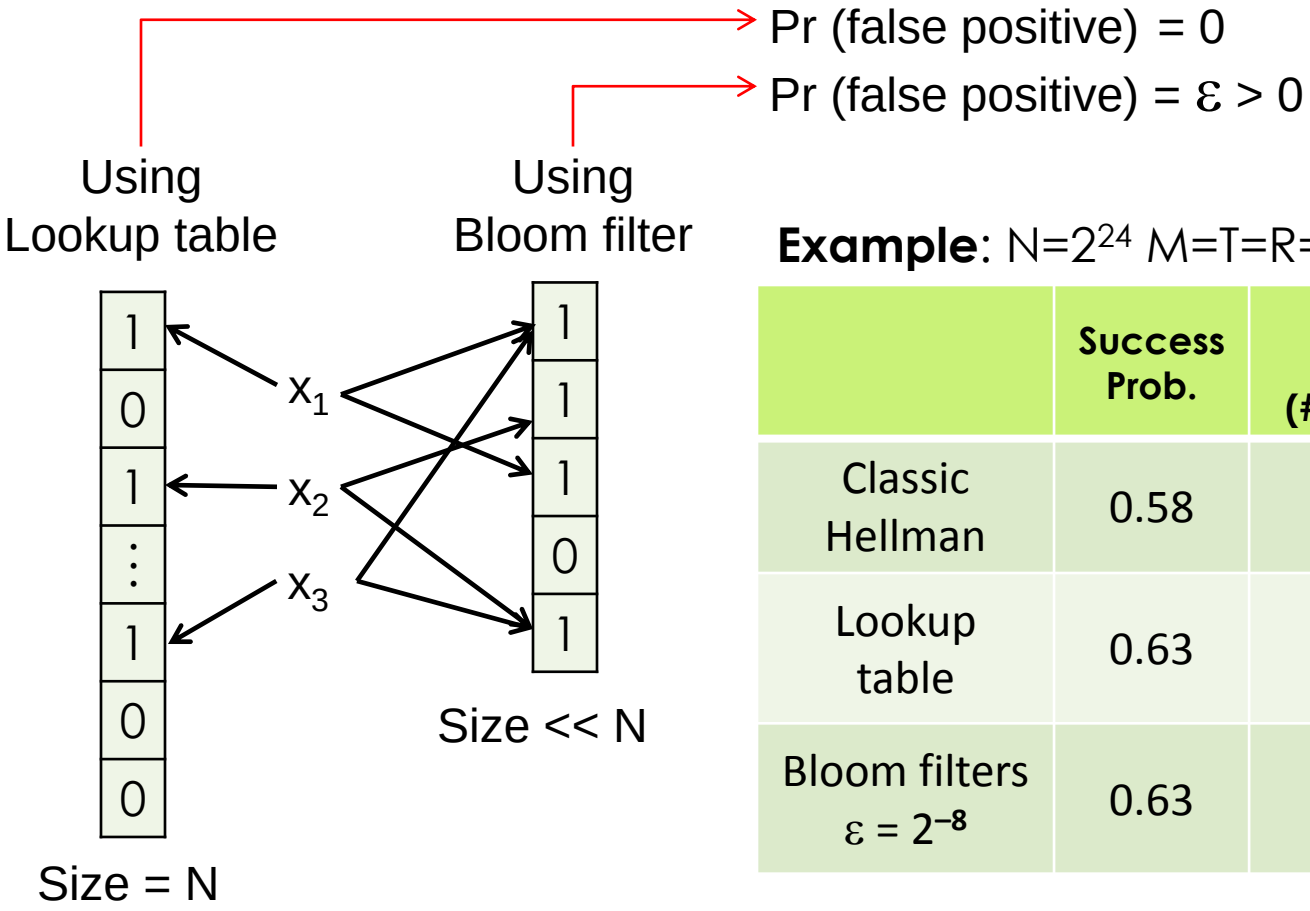
Perfect Hellman Tables

How? Add row $[x_1, x_2, \ldots, x_T]$, only if all elements are new.



Perfect Hellman Tables

How? Add row $[x_1, x_2, \ldots, x_T]$, only if all elements are new.



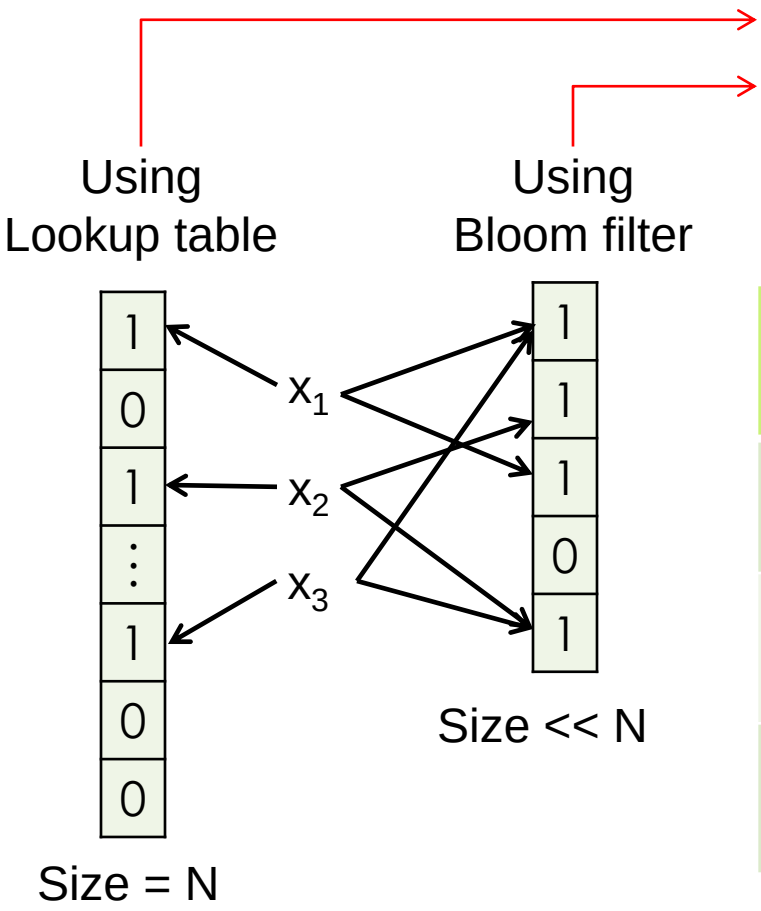
Example: $N=2^{24}$ $M=T=R=2^8$

	Success Prob.	Offline Time (# *f* calls)	Offline Memory (bits)
Classic Hellman	0.58	2^{24}	$2^{21.6}$ (= 2 x 24 × 2^{16})
Lookup table	0.63	$2^{24.5}$ (x 1.38)	$2^{24.2}$ (x 6.29)
Bloom filters $\varepsilon = 2^{-8}$	0.63	$2^{24.7}$ (x 1.62)	$2^{21.8}$ (x 1.16)

Online memory (bits): $2^{21.6}$
Online time (# f calls): 2^{16}

Perfect Hellman Tables

How? Add row $[x_1,x_2,\ldots,x_T]$, only if all elements are new.



Example: $N=2^{24}$ $M=T=R=2^8$

	Success Prob.	Offline Time (# *f* calls)	Offline Memory (bits)
Classic Hellman	0.58	2^{24}	$2^{21.6}$ ($= 2 \times 24 \times 2^{16}$)
Lookup table	0.63	$2^{24.5}$ (x 1.38)	$2^{24.2}$ (x 6.29)
Bloom filters $\varepsilon = 2^{-8}$	0.63	$2^{24.7}$ (x 1.62)	$2^{21.8}$ (x 1.16)

Online memory (bits): $2^{21.6}$
Online time (# f calls): 2^{16}

Ongoing Research

On how to use Bloom filters to improve the efficiency of TMTO attacks

THANKS!

meltem.turan@nist.gov