A comparison of time-memory trade-off attacks on stream ciphers

AfricaCrypt 2013

Fabian van den Broek

& Erik Poll

Institute for Computing and Information Sciences – Digital Security
Radboud University Nijmegen

23 June 2013

The Model

When inverting a “random” function $f(x) = y$, with $x, y \in N$

The Model

When inverting a “random” function $f(x) = y$, with $x, y \in N$

Possibilities:

- Brute force x: 2^{N-1} computations
- Complete Dictionary attack: 2^{N} computations and storage

The Model

When inverting a “random” function $f(x) = y$, with $x, y \in N$

Possibilities:

- Brute force x: 2^{N-1} computations
- Complete Dictionary attack: 2^N computations and storage

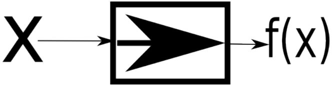
TMTO attacks lie inbetween these extremes

The Model II

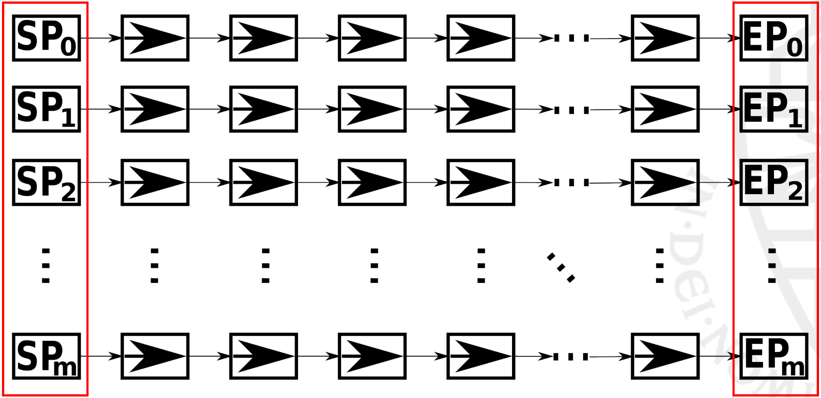
In $f(x) = y$, f can be:

- a hash function
- a block cipher $f(x) = f'(x, m)$
- a stream cipher $f'(x) = f(x) \oplus m$

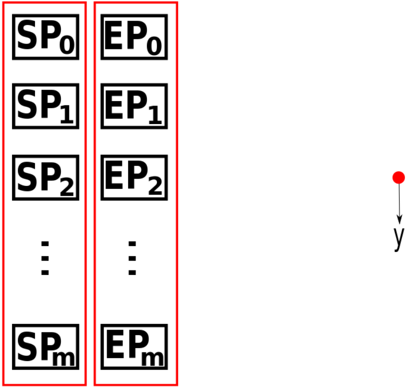
General TMTO



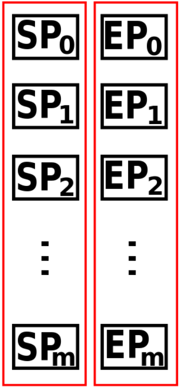
General TMTO

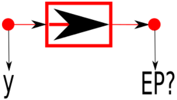


General TMTO

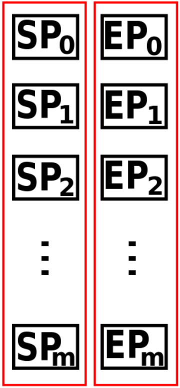


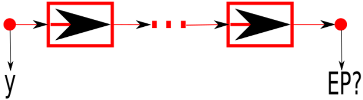
General TMTO



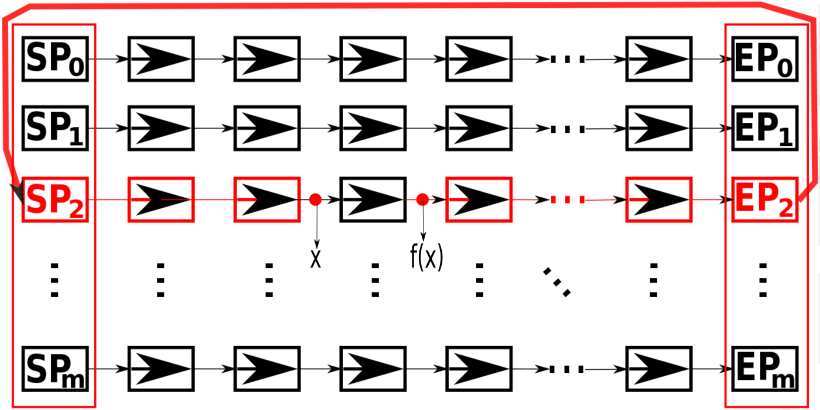


General TMTO

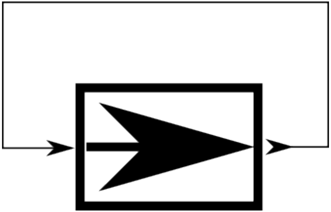




General TMTO

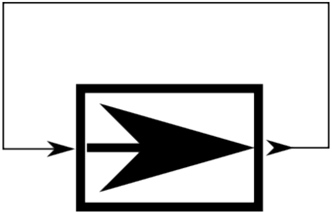


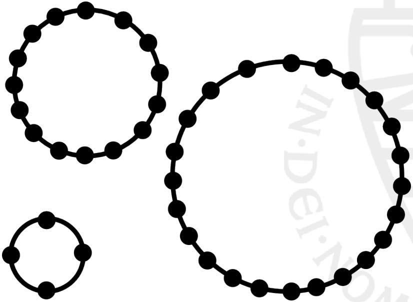
Coverage



Coverage

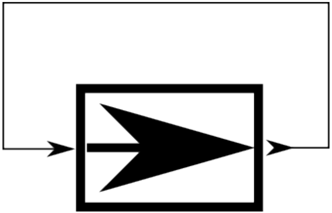
Permutation?

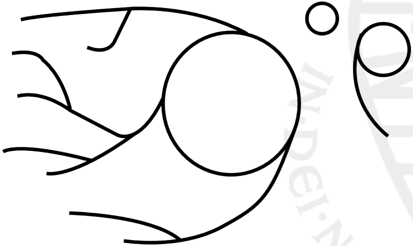




Coverage

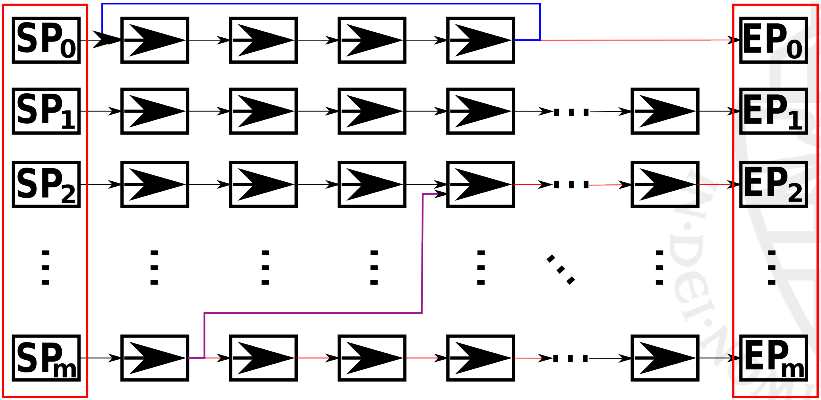
A random iterative function



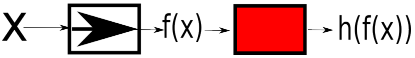


General TMTO: Another problem

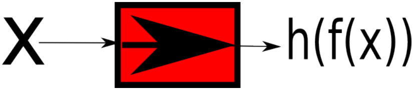
General TMTO: Another problem



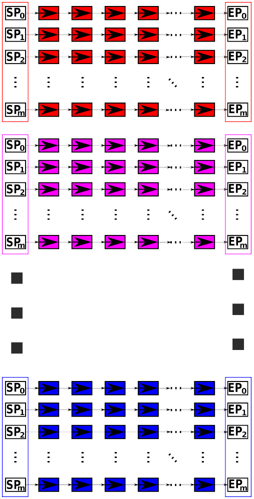
Hellman’s solution



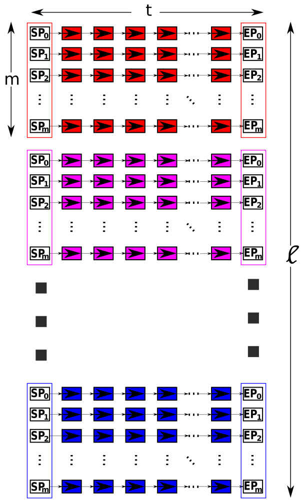
Hellman's solution



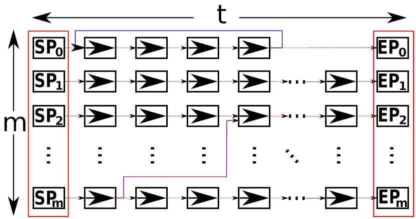
Hellman's solution



Hellman's solution



Hellman's solution



Succes chance bounded by:

$$(1/N)\sum_{i=1}^{m}\sum_{j=0}^{t-1}[(N-it)/N)]^{j+1} \leq \mathbb{P} \leq (mt/N)$$

Proven by Hellman using the matrix stopping rule: $mt^2 = N$

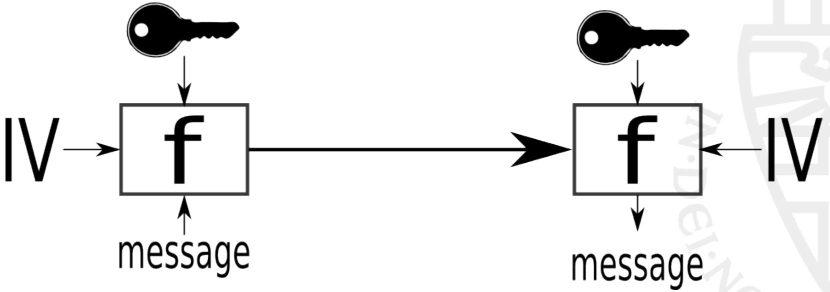
Hellman's solution

For $N = 2^n$ and $mt^2 = N$: Hellman needs 2^n pre-computation encryptions, stored in $2^{2n/3}$ values
$2^{2n/3}$ encryptions then reverse the function f with success chance ≈ 0.55

TMTO Improvements

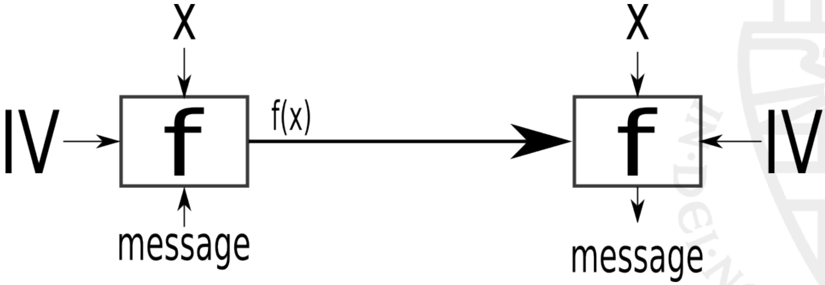
TMTO Improvements: for Streamciphers

For a block cipher:



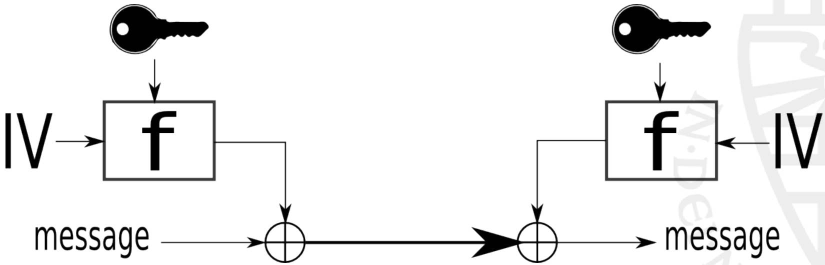
TMTO Improvements: for Streamciphers

For a block cipher:



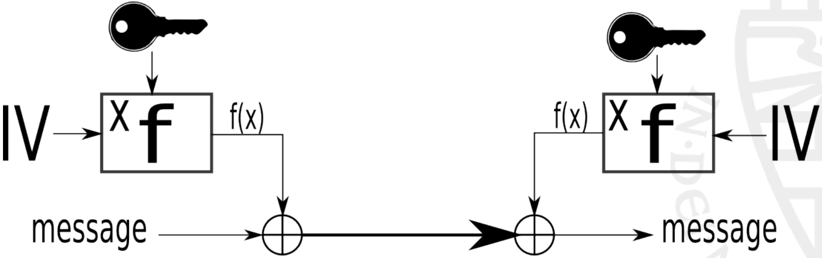
TMTO Improvements: for Streamciphers

For a stream cipher:



TMTO Improvements: for Streamciphers

For a stream cipher:



TMTO Improvements: for Streamciphers

Even better:

Suppose you created TMTO tables for $|y| = 6$
And you obtain 9 bits:
001101011

TMTO Improvements: for Streamciphers

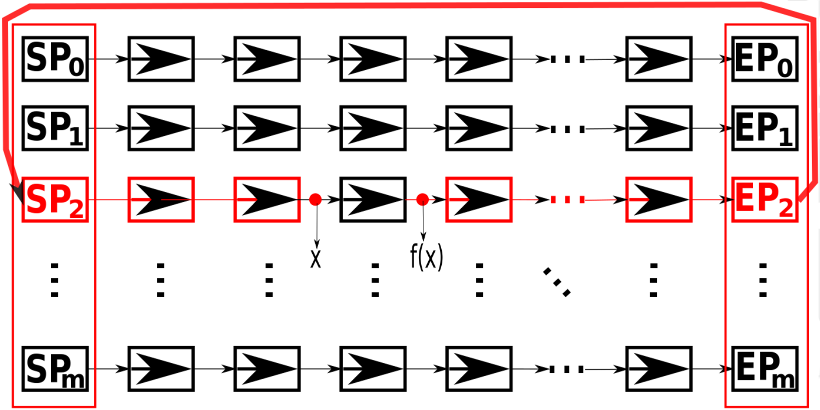
Even better:

Suppose you created TMTO tables for $|y| = 6$
And you obtain 9 bits:
001101011

That's 4 samples:
001101
011010
110101
101011

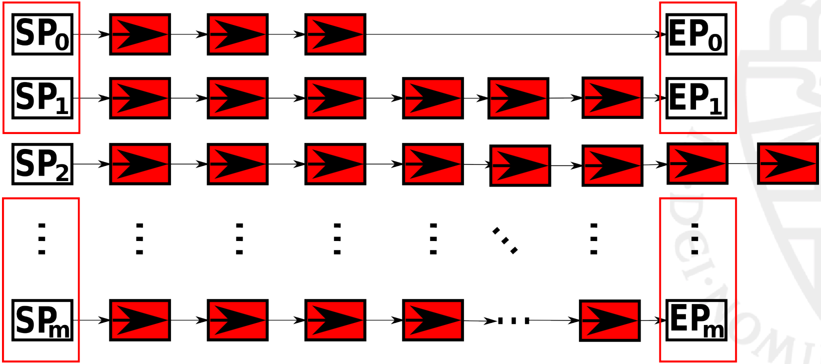
TMTO Improvements: Distinguished Points

Problem: Hellman's attack needs *t* diskseeks per sample per table

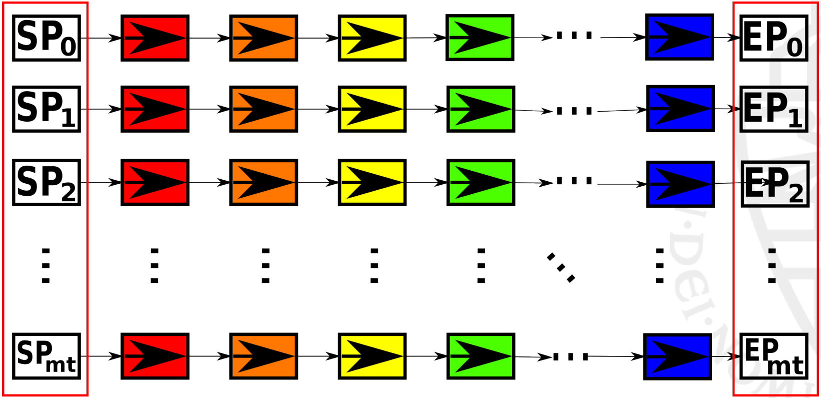


TMTO Improvements: Distinguished Points

End chains in distinguished points (e.g. $0^k||x$)

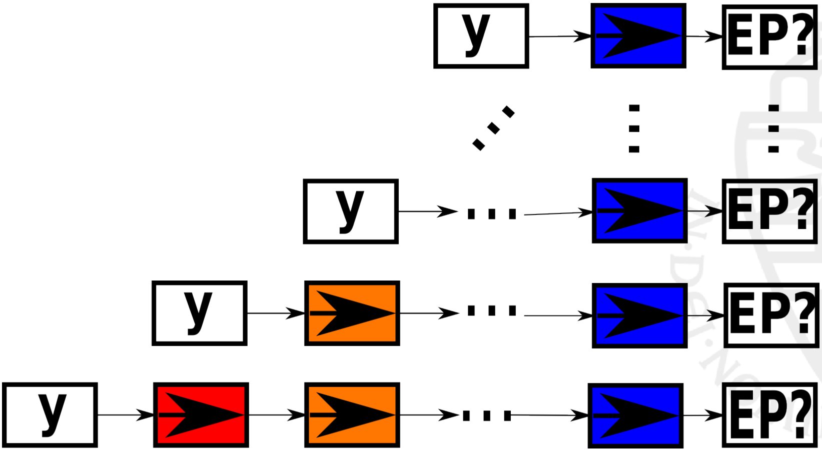


TMTO Improvements: Rainbow Tables



TMTO Improvements: Rainbow Tables

TMTO Improvements: Rainbow Tables

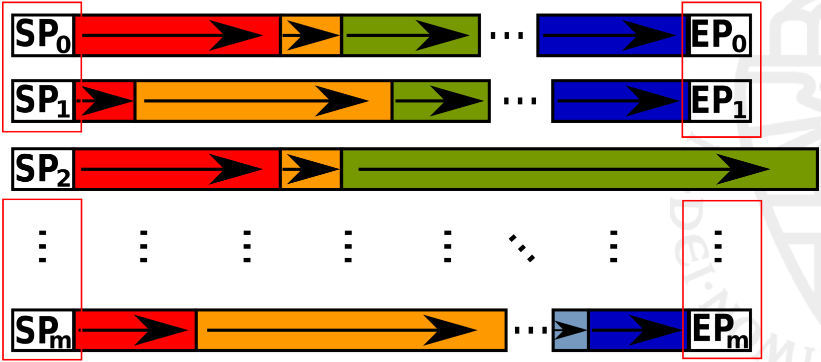


~~Kraken~~ Fuzzy rainbowtables

How to combine DP with RT?

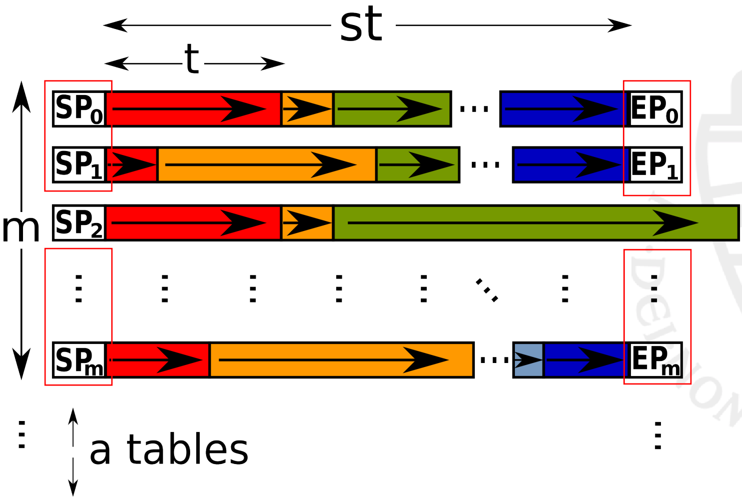
~~Kraken~~ Fuzzy rainbowtables

How to combine DP with RT?



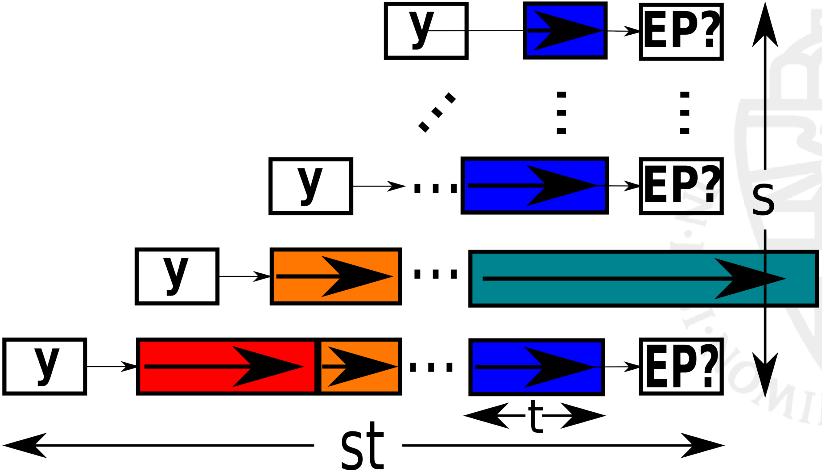
~~Kraken~~ Fuzzy rainbowtables

How to combine DP with RT?



~~Kraken~~ Fuzzy rainbowtables

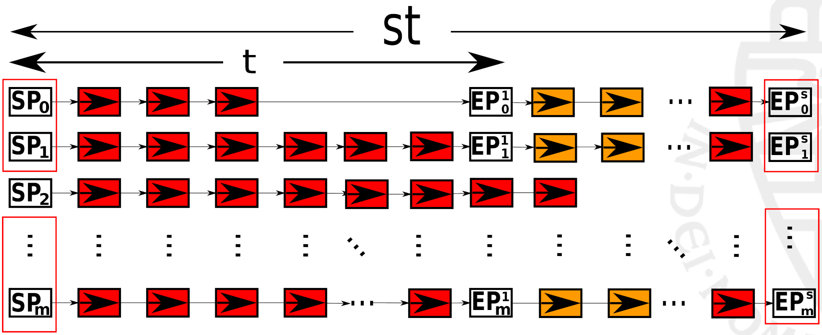
How to combine DP with RT?



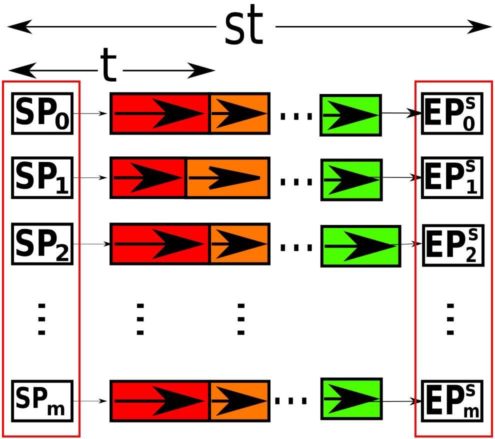
Fuzzy rainbowtables

Essentially an extra Time-Memory trade-off within a TMTO

Fuzzy rainbowtables



Fuzzy rainbowtables



Comparing TMTOs

- $TM^2 = N^2$
- $TM^2 D^2 = N^2$

Comparing TMTOs

What to measure w.r.t. N?

- Pre-computation costs, P
- Memory costs, M
- Attack time costs, T
  - Computation costs, T_c
  - Seek time costs, T_s
- Coverage, C
- Pre-computation ratio, ρ
- Success chance, $\mathbb{P}$

Comparing TMTOs

You might assume $\mathbb{P} = \rho = \frac{C}{N}$

Comparing TMTOs

However: $\mathbb{P} \neq \rho \neq \frac{C}{N}$

- Chain mergers
- Multiple samples
- Not all outcomes of $f(X)$ need to be equally likely

Comparison

For $D\rho = 1$ and $mt^2 = n$:

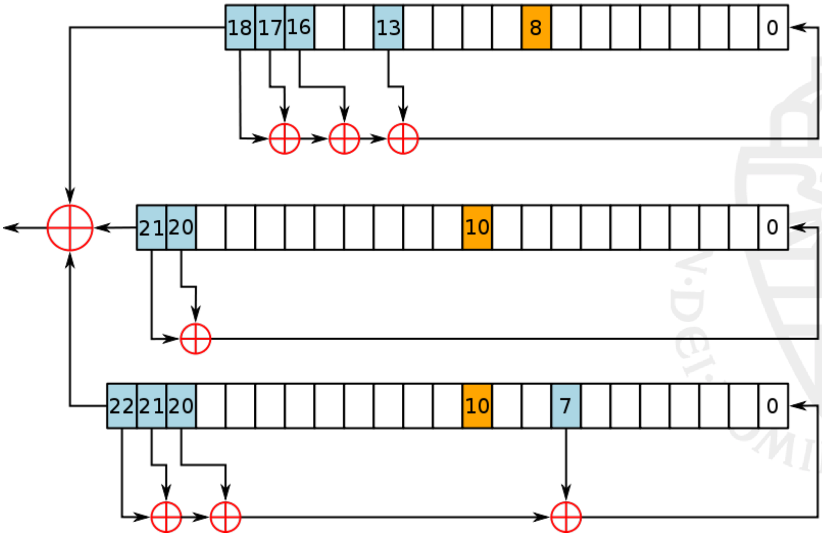
TMTO technique	M	T_c	T_s
Hellman's attack	$2mt/D$	t^2	t^2 in m entries
Dist. Point	$2mt/D$	t^2	t in m entries
Rainbow Table	$2mt/D$	$\frac{t(t+1)}{2}D$	tD in mt/D entries
Fuzzy Rainbow	$2mt/sD$	$\frac{(s+1)}{2}t^2$	t in m entries

But this comparison is unfair

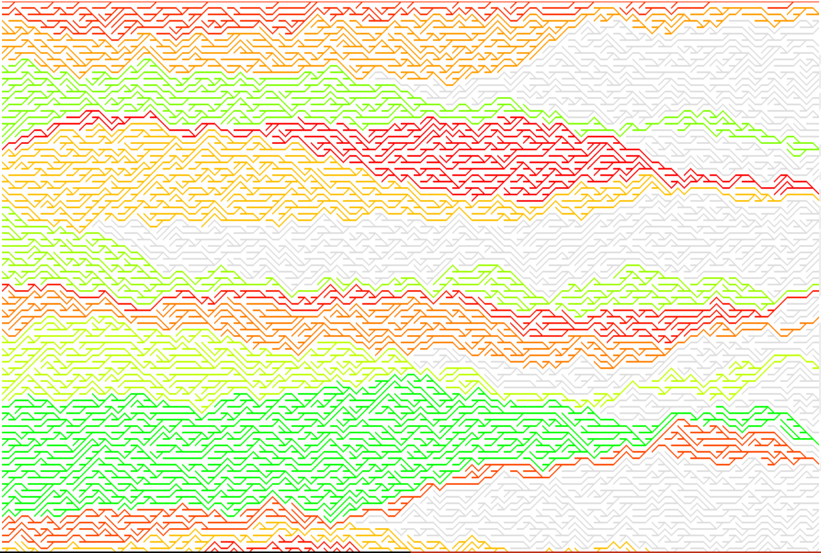
- No measure on chain mergers
- False alarms
- Perfect / non-perfect tables
- What value to choose for s?

A5/1

A5/1



A5/1



Preparing the attack

- Should fit in 2TB
- Should accept 64 bit keystream samples

Preparing the attack

- Should fit in 2TB
- Should accept 64 bit keystream samples

Cipher mode complete

01 01 08 06 32 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b 2b

The Kraken Numbers

- 12 bit distinguished points, $k = 12$
- 8 colors, $s = 8$
- 40 tables (in 1.6TB), $l = 40$
- 8662000000 rows per table ($\approx 2^{33}$)
- In total covers around $2^{53.3}$
- Attack can run ≤ 1 minute

Making perfect tables

- 2^{33} rows per table
- now throw away chains ending in the same endpoint

Making perfect tables

- 2^{33} rows per table
- now throw away chains ending in the same endpoint
- $2^{32.5}$ rows left.
- 29% of all chains merged.

Making perfect tables

- 2^{33} rows per table
- now throw away chains ending in the same endpoint
- $2^{32.5}$ rows left.
- 29% of all chains merged.
- The perfect tables cover around $2^{52.8}$
- $\mathbb{P} \approx 0.2$

Independent line of work

Hong et al. find the fuzzy rainbowtable approach better for most cases and in their comparison account for chain mergers.
Using $mt^2s \approx N$ as matrix stopping rule.

- Jin Hong and Sunghwan Moon, "A Comparison of Cryptanalytic Tradeoff Algorithms", ePrint 2012-09.
- Byoung-Il Kim and Jin Hong, "Analysis of the Non-Perfect Table Fuzzy Rainbow Tradeoff", ACISP 2013

Questions