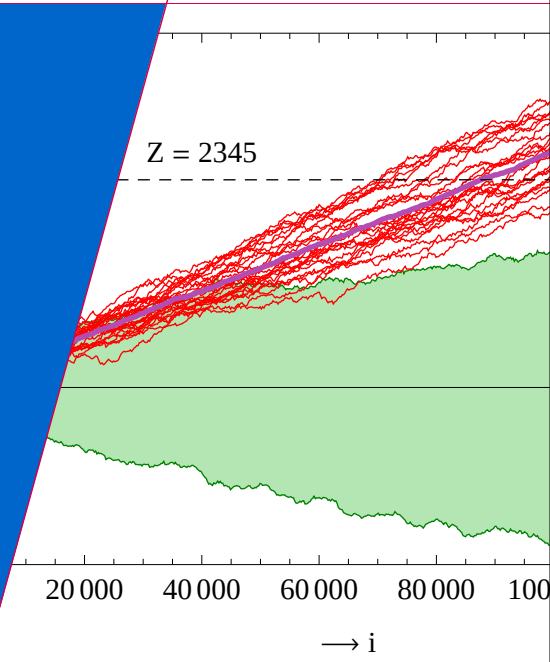
Dynamic Tardos traitor tracing schemes

Thijs Laarhoven



Technische Universiteit
Eindhoven
University of Technology

DIAMANT, November 24th, 2011

Contents

Introduction: Illegal redistribution

Alice	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Bob	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Charlie	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
David	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Eve	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Fred	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
George	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...

These days a lot of digital content is sold and distributed, e.g. movies, software.

Introduction: Illegal redistribution

Alice	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Bob	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Charlie	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
David	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Eve	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Fred	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
George	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...
Copy	0	1	1	1	0	0	1	1	1	0	1	1	0	0	1	0	...

These days a lot of digital content is sold and distributed, e.g. movies, software.

Problem: Easy to copy and distribute. Who did it?

Introduction: Embed watermarks

Alice	0	1	1	1	0	0	1	1	1	0	1	1	0	1	0	0	...
Bob	0	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	...
Charlie	0	1	0	1	0	1	0	1	1	0	0	1	1	0	1	0	...
David	0	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	...
Eve	0	1	0	1	0	1	0	1	1	0	1	1	1	0	0	0	...
Fred	0	1	0	1	0	0	1	1	1	0	0	1	0	1	0	0	...
George	0	1	1	1	0	1	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	0	1	0	1	1	1	1	0	0	1	0	1	1	0	...

Solution: Embed watermarks in data to link copies to users.

Introduction: Embed watermarks

Alice	0	1	1	1	0	0	1	1	1	0	1	1	0	1	0	0	...
Bob	0	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	...
Charlie	0	1	0	1	0	1	0	1	1	0	0	1	1	0	1	0	...
David	0	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	...
Eve	0	1	0	1	0	1	0	1	1	0	1	1	1	0	0	0	...
Fred	0	1	0	1	0	0	1	1	1	0	0	1	0	1	0	0	...
George	0	1	1	1	0	1	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	0	1	0	1	1	1	1	0	0	1	0	1	1	0	...
Copy	0	1	0	1	0	1	0	1	1	0	1	1	1	0	0	0	...

Solution: Embed watermarks in data to link copies to users.

Someone buys the content, copies it and distributes the copy.

Introduction: Embed watermarks

Alice	0	1	**1**	1	0	**0**	**1**	1	1	0	**1**	1	**0**	**1**	**0**	0	...
Bob	0	1	**1**	1	0	**1**	**0**	1	1	0	**1**	1	**1**	**1**	**1**	0	...
Charlie	0	1	**0**	1	0	**1**	**0**	1	1	0	**0**	1	**1**	**0**	**1**	0	...
David	0	1	**1**	1	0	**0**	**0**	1	1	0	**1**	1	**0**	**0**	**0**	0	...
Eve	0	1	**0**	1	0	**1**	**0**	1	1	0	**1**	1	**1**	**0**	**0**	0	...
Fred	0	1	**0**	1	0	**0**	**1**	1	1	0	**0**	1	**0**	**1**	**0**	0	...
George	0	1	**1**	1	0	**1**	**1**	1	1	0	**1**	1	**0**	**0**	**1**	0	...
Henry	0	1	**0**	1	0	**1**	**1**	1	1	0	**0**	1	**0**	**1**	**1**	0	...
Copy	0	1	**0**	1	0	**1**	**0**	1	1	0	**1**	1	**1**	**0**	**0**	0	...

Solution: Embed watermarks in data to link copies to users.

Someone buys the content, copies it and distributes the copy.

The distributor detects the copy and traces it to the guilty user.

Introduction: Collusion-attacks

Alice	0	1	1	1	0	0	1	1	1	0	1	1	0	1	0	0	...
Bob	0	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	...
Charlie	0	1	0	1	0	1	0	1	1	0	0	1	1	0	1	0	...
David	0	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	...
Eve	0	1	0	1	0	1	0	1	1	0	1	1	1	0	0	0	...
Fred	0	1	0	1	0	0	1	1	1	0	0	1	0	1	0	0	...
George	0	1	1	1	0	1	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	0	1	0	1	1	1	1	0	0	1	0	1	1	0	...

Problem: Users may collude and compare their content to find the watermark.

Introduction: Collusion-attacks

Alice	0	1	**1**	1	0	**0**	**1**	1	1	0	1	1	**0**	**1**	0	0	...
Bob	0	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	...
Charlie	0	1	0	1	0	1	0	1	1	0	0	1	1	0	1	0	...
David	0	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	...
Eve	0	1	**0**	1	0	**1**	**0**	1	1	0	1	1	**1**	**0**	0	0	...
Fred	0	1	0	1	0	0	1	1	1	0	0	1	0	1	0	0	...
George	0	1	1	1	0	1	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	0	1	0	1	1	1	1	0	0	1	0	1	1	0	...

Problem: Users may collude and compare their content to find the watermark.

Introduction: Collusion-attacks

Alice	0	1	**1**	1	0	**0**	**1**	1	1	0	1	1	**0**	**1**	0	0	...
Bob	0	1	1	1	0	1	0	1	1	0	1	1	1	1	1	0	...
Charlie	0	1	0	1	0	1	0	1	1	0	0	1	1	0	1	0	...
David	0	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	...
Eve	0	1	**0**	1	0	**1**	**0**	1	1	0	1	1	**1**	**0**	0	0	...
Fred	0	1	0	1	0	0	1	1	1	0	0	1	0	1	0	0	...
George	0	1	1	1	0	1	1	1	1	0	1	1	0	0	1	0	...
Henry	0	1	0	1	0	1	1	1	1	0	0	1	0	1	1	0	...
Copy	0	1	**1**	1	0	**1**	**0**	1	1	0	1	1	**0**	**1**	0	0	...

Problem: Users may collude and compare their content to find the watermark.

On detectable positions they choose a bit, the rest they simply copy.

Introduction: Collusion-attacks

Alice	0	1	**1**	1	0	**0**	**1**	1	1	0	**1**	1	**0**	**1**	**0**	0	...
Bob	0	1	**1**	1	0	**1**	**0**	1	1	0	**1**	1	**1**	**1**	**1**	0	...
Charlie	0	1	**0**	1	0	**1**	**0**	1	1	0	**0**	1	**1**	**0**	**1**	0	...
David	0	1	**1**	1	0	**0**	**0**	1	1	0	**1**	1	**0**	**0**	**0**	0	...
Eve	0	1	**0**	1	0	**1**	**0**	1	1	0	**1**	1	**1**	**0**	**0**	0	...
Fred	0	1	**0**	1	0	**0**	**1**	1	1	0	**0**	1	**0**	**1**	**0**	0	...
George	0	1	**1**	1	0	**1**	**1**	1	1	0	**1**	1	**0**	**0**	**1**	0	...
Henry	0	1	**0**	1	0	**1**	**1**	1	1	0	**0**	1	**0**	**1**	**1**	0	...
Copy	0	1	**1**	1	0	**1**	**0**	1	1	0	**1**	1	**0**	**1**	**0**	0	...

Problem: Users may collude and compare their content to find the watermark.

On detectable positions they choose a bit, the rest they simply copy.

Now the distributor cannot find the guilty users.

Introduction: Our problem

Alice	1	0	1	1	0	1	0	...
Bob	1	1	0	1	1	1	1	...
Charlie	0	1	0	0	1	0	1	...
David	1	0	0	1	0	0	0	...
Eve	0	1	0	1	1	0	0	...
Fred	0	0	1	0	0	1	0	...
George	1	1	1	1	0	0	1	...
Henry	0	1	1	0	0	1	1	...
Copy	1	1	0	1	0	1	0	...

So we need **collusion-resistant traitor tracing schemes**, consisting of:

- Codeword generation: Assigning symbols to users.
- Tracing algorithm: Tracing copies back to guilty users.

We only focus on the embedded watermarks, not on the data itself.

Model: Abstraction

Alice	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$X_{1,5}$	$X_{1,6}$	$\dots$	$X_{1,\ell}$
Bob	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$X_{2,5}$	$X_{2,6}$	$\dots$	$X_{2,\ell}$
Charlie	$X_{3,1}$	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$	$X_{3,5}$	$X_{3,6}$	$\dots$	$X_{3,\ell}$
David	$X_{4,1}$	$X_{4,2}$	$X_{4,3}$	$X_{4,4}$	$X_{4,5}$	$X_{4,6}$	$\dots$	$X_{4,\ell}$
Eve	$X_{5,1}$	$X_{5,2}$	$X_{5,3}$	$X_{5,4}$	$X_{5,5}$	$X_{5,6}$	$\dots$	$X_{5,\ell}$
Fred	$X_{6,1}$	$X_{6,2}$	$X_{6,3}$	$X_{6,4}$	$X_{6,5}$	$X_{6,6}$	$\dots$	$X_{6,\ell}$
George	$X_{7,1}$	$X_{7,2}$	$X_{7,3}$	$X_{7,4}$	$X_{7,5}$	$X_{7,6}$	$\dots$	$X_{7,\ell}$
Henry	$X_{8,1}$	$X_{8,2}$	$X_{8,3}$	$X_{8,4}$	$X_{8,5}$	$X_{8,6}$	$\dots$	$X_{8,\ell}$
Copy	y_1	y_2	y_3	y_4	y_5	y_6	$\dots$	y_ℓ

Notation: Users $j = 1, \dots, n$, positions $i = 1, \dots \ell$, code matrix $X = (X_{j,i})$.

Pirates: A coalition C of c pirates, generates $\vec{y}$ such that $y_i \in \{X_{j,i} : j \in C\}$.

Tracing algorithm: Maps output $\vec{y}$ to a set of accused users $\hat{C} \stackrel{?}{=} C$.

Successful if $\hat{C} \subseteq C$ and either $\hat{C} \cap C \neq \emptyset$ or (ideally) $C \subseteq \hat{C}$.

Model: Static vs. dynamic

Static schemes: **Distribute all symbols at the start.**

- Codewords do not depend on pirate output.
- Requirement: Catch at least one pirate.
- Applications: Video on demand, software.

Dynamic schemes: **Distribute symbols $X_{j,i}$ after receiving y_{i-1}.**

- Codewords may depend on previous pirate output.
- Users may be disconnected from the system before distributing new content.
- Requirement: Catch all pirates.
- Applications: Live broadcasts, pay-tv.

This presentation: Both types.

Model: Deterministic vs. probabilistic

Deterministic schemes: **No error in accusations.**

- Do not exist for $c \geq 2$ and binary alphabet: need bigger alphabet.

Probabilistic schemes: **Accusation errors bounded by** $\epsilon_1, \epsilon_2 > 0$**.**

- Accuse no innocent users with probability at least $1 - \epsilon_1$.
- Static: Catch at least one guilty user w.p. at least $1 - \epsilon_2$.
- Dynamic: Catch all guilty users w.p. at least $1 - \epsilon_2$.

This presentation: Only probabilistic (binary) schemes.

Results: Related work

ℓ codelength, c colluders, n users, $\epsilon_{1,2}$ innocent/guilty error probabilities

	Codelength (small coalitions)	Codelength (large coalitions)
Static schemes	$\ell \geq \Omega(c^2 \ln(n/\epsilon_1))$	$\ell \geq 1.38c^2 \ln(n/\epsilon_1)$
- Boneh, Shaw	$\ell \approx 32c^4 \ln(n/\epsilon_1) \ln(c/\epsilon_1)$	$\ell \approx 32c^4 \ln(n/\epsilon_1) \ln(c/\epsilon_1)$
- Tardos	$\ell = 100c^2 \ln(n/\epsilon_1)$	$\ell = 100c^2 \ln(n/\epsilon_1)$
- Vladimirova et al.	$\ell \approx 90c^2 \ln(n/\epsilon_1)$	$\ell \approx 39c^2 \ln(n/\epsilon_1)$
- Blayer, Tassa	$\ell = 85c^2 \ln(n/\epsilon_1)$	$\ell \approx 20c^2 \ln(n/\epsilon_1)$
- Škorić et al.	$\ell \approx 50c^2 \ln(n/\epsilon_1)$	$\ell \approx 10c^2 \ln(n/\epsilon_1)$
- Nuida et al.	$\ell \approx 5c^2 \ln(n/\epsilon_1)$	$\ell \approx 5.4c^2 \ln(n/\epsilon_1)$
- Laarhoven, De Weger	$\ell \approx 24c^2 \ln(n/\epsilon_1)$	$\ell \approx 4.9c^2 \ln(n/\epsilon_1)$
Dynamic schemes	?	?
- Tassa	$\ell = O(c^4 \log(n) \ln(c/\epsilon_1))$	$\ell = O(c^4 \log(n) \ln(c/\epsilon_1))$

Results: Contributions

ℓ codelength, c colluders, n users, $\epsilon_{1,2}$ innocent/guilty error probabilities

	Codelength (small coalitions)	Codelength (large coalitions)
Static schemes	$\ell \geq \Omega(c^2 \ln(n/\epsilon_1))$	$\ell \geq 1.38c^2 \ln(n/\epsilon_1)$
- Boneh, Shaw	$\ell \approx 32c^4 \ln(n/\epsilon_1) \ln(c/\epsilon_1)$	$\ell \approx 32c^4 \ln(n/\epsilon_1) \ln(c/\epsilon_1)$
- Tardos	$\ell = 100c^2 \ln(n/\epsilon_1)$	$\ell = 100c^2 \ln(n/\epsilon_1)$
- Vladimirova et al.	$\ell \approx 90c^2 \ln(n/\epsilon_1)$	$\ell \approx 39c^2 \ln(n/\epsilon_1)$
- Blayer, Tassa	$\ell = 85c^2 \ln(n/\epsilon_1)$	$\ell \approx 20c^2 \ln(n/\epsilon_1)$
- Škorić et al.	$\ell \approx 50c^2 \ln(n/\epsilon_1)$	$\ell \approx 10c^2 \ln(n/\epsilon_1)$
- Nuida et al.	$\ell \approx 5c^2 \ln(n/\epsilon_1)$	$\ell \approx 5.4c^2 \ln(n/\epsilon_1)$
- Laarhoven, De Weger	$\ell \approx 24c^2 \ln(n/\epsilon_1)$	$\ell \approx 4.9c^2 \ln(n/\epsilon_1)$
Dynamic schemes	?	?
- Tassa	$\ell = O(c^4 \log(n) \ln(c/\epsilon_1))$	$\ell = O(c^4 \log(n) \ln(c/\epsilon_1))$
- Dynamic Tardos	$\ell = O(c^2 \ln(n/\epsilon_1))$	$\ell \approx \mathbf{4.9}c^2 \ln(n/\epsilon_1)$
- Universal Tardos	$\ell = O(c^2 \ln(n/\epsilon_1))$	$\ell \approx \mathbf{4.9}c^2 \ln(n/\epsilon_1)$

Results: Outline

ℓ codelength, c colluders, n users, $\epsilon_{1,2}$ innocent/guilty error probabilities

	Codelength (small coalitions)	Codelength (large coalitions)
Static schemes	$\ell \geq \Omega(c^2 \ln(n/\epsilon_1))$	$\ell \geq 1.38c^2 \ln(n/\epsilon_1)$
- Boneh, Shaw	$\ell \approx 32c^4 \ln(n/\epsilon_1) \ln(c/\epsilon_1)$	$\ell \approx 32c^4 \ln(n/\epsilon_1) \ln(c/\epsilon_1)$
- Tardos	$\ell = 100c^2 \ln(n/\epsilon_1)$	$\ell = 100c^2 \ln(n/\epsilon_1)$
- Vladimirova et al.	$\ell \approx 90c^2 \ln(n/\epsilon_1)$	$\ell \approx 39c^2 \ln(n/\epsilon_1)$
- Blayer, Tassa	$\ell = 85c^2 \ln(n/\epsilon_1)$	$\ell \approx 20c^2 \ln(n/\epsilon_1)$
- Škorić et al.	$\ell \approx 50c^2 \ln(n/\epsilon_1)$	$\ell \approx 10c^2 \ln(n/\epsilon_1)$
- Nuida et al.	$\ell \approx 5c^2 \ln(n/\epsilon_1)$	$\ell \approx 5.4c^2 \ln(n/\epsilon_1)$
- Laarhoven, De Weger	$\ell \approx 24c^2 \ln(n/\epsilon_1)$	$\ell \approx 4.9c^2 \ln(n/\epsilon_1)$
Dynamic schemes	?	?
- Tassa	$\ell = O(c^4 \log(n) \ln(c/\epsilon_1))$	$\ell = O(c^4 \log(n) \ln(c/\epsilon_1))$
- Dynamic Tardos	$\ell = O(c^2 \ln(n/\epsilon_1))$	$\ell \approx \mathbf{4.9}c^2 \ln(n/\epsilon_1)$
- Universal Tardos	$\ell = O(c^2 \ln(n/\epsilon_1))$	$\ell \approx \mathbf{4.9}c^2 \ln(n/\epsilon_1)$

The Tardos scheme: Introduction

Codeword generation: **Randomized code**

- For each position i, first select a bias $p_i \in [\delta, 1-\delta]$ from $F_\delta(p)$.
- Then, for each user j, select $X_{j,i} = 1$ with probability p_i.

Tracing algorithm: **Assign scores to users**

- For each position i and user j, calculate $S_{j,i} = g(X_{j,i}, y_i, p_i)$.
- Calculate the total user score as $S_j(\ell) = \sum_i S_{j,i}$.
- Accusation: Accuse user j if $S_j(\ell) > Z$.

Depends on scheme parameters ℓ, Z, δ.

The Tardos scheme: Example

Parameters we choose:

- $n = 8$: In total there will be 8 users in the system.
- $c = 3$: The simulated coalition consists of 3 colluders.
- $\epsilon_1 = 0.01$: With 99% certainty no innocent users are accused.
- $\epsilon_2 = 0.01$: With 99% certainty at least one pirate is caught.

Parameters for the scheme that roll out:

- $\ell = 1208$: The codelength of the scheme is 1208.
- $Z = 146$: If a user's final score exceeds 146, he will be accused.
- $\delta = 0.0115$: The values of p_i will be in the interval $[\delta, 1 - \delta]$.

The Tardos scheme: Codewords

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

For each i, generate p_i from $F_\delta(p)$.

p_i	p_1	p_2	p_3	p_4	p_5	p_6	$\cdots$	p_{1208}
Alice	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$X_{1,5}$	$X_{1,6}$	$\cdots$	$X_{1,1208}$
Bob	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$X_{2,5}$	$X_{2,6}$	$\cdots$	$X_{2,1208}$
Charlie	$X_{3,1}$	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$	$X_{3,5}$	$X_{3,6}$	$\cdots$	$X_{3,1208}$
David	$X_{4,1}$	$X_{4,2}$	$X_{4,3}$	$X_{4,4}$	$X_{4,5}$	$X_{4,6}$	$\cdots$	$X_{4,1208}$
Eve	$X_{5,1}$	$X_{5,2}$	$X_{5,3}$	$X_{5,4}$	$X_{5,5}$	$X_{5,6}$	$\cdots$	$X_{5,1208}$
Fred	$X_{6,1}$	$X_{6,2}$	$X_{6,3}$	$X_{6,4}$	$X_{6,5}$	$X_{6,6}$	$\cdots$	$X_{6,1208}$
George	$X_{7,1}$	$X_{7,2}$	$X_{7,3}$	$X_{7,4}$	$X_{7,5}$	$X_{7,6}$	$\cdots$	$X_{7,1208}$
Henry	$X_{8,1}$	$X_{8,2}$	$X_{8,3}$	$X_{8,4}$	$X_{8,5}$	$X_{8,6}$	$\cdots$	$X_{8,1208}$

The Tardos scheme: Codewords

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

For each i, generate p_i from $F_\delta(p)$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$X_{1,5}$	$X_{1,6}$	...	$X_{1,1208}$
Bob	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$X_{2,5}$	$X_{2,6}$	...	$X_{2,1208}$
Charlie	$X_{3,1}$	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$	$X_{3,5}$	$X_{3,6}$	...	$X_{3,1208}$
David	$X_{4,1}$	$X_{4,2}$	$X_{4,3}$	$X_{4,4}$	$X_{4,5}$	$X_{4,6}$	...	$X_{4,1208}$
Eve	$X_{5,1}$	$X_{5,2}$	$X_{5,3}$	$X_{5,4}$	$X_{5,5}$	$X_{5,6}$	...	$X_{5,1208}$
Fred	$X_{6,1}$	$X_{6,2}$	$X_{6,3}$	$X_{6,4}$	$X_{6,5}$	$X_{6,6}$	...	$X_{6,1208}$
George	$X_{7,1}$	$X_{7,2}$	$X_{7,3}$	$X_{7,4}$	$X_{7,5}$	$X_{7,6}$	...	$X_{7,1208}$
Henry	$X_{8,1}$	$X_{8,2}$	$X_{8,3}$	$X_{8,4}$	$X_{8,5}$	$X_{8,6}$	...	$X_{8,1208}$

The Tardos scheme: Codewords

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

For each i, j, take $X_{j,i} = 1$ with probability p_i.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$X_{1,5}$	$X_{1,6}$	...	$X_{1,1208}$
Bob	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$X_{2,5}$	$X_{2,6}$	...	$X_{2,1208}$
Charlie	$X_{3,1}$	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$	$X_{3,5}$	$X_{3,6}$	...	$X_{3,1208}$
David	$X_{4,1}$	$X_{4,2}$	$X_{4,3}$	$X_{4,4}$	$X_{4,5}$	$X_{4,6}$	...	$X_{4,1208}$
Eve	$X_{5,1}$	$X_{5,2}$	$X_{5,3}$	$X_{5,4}$	$X_{5,5}$	$X_{5,6}$	...	$X_{5,1208}$
Fred	$X_{6,1}$	$X_{6,2}$	$X_{6,3}$	$X_{6,4}$	$X_{6,5}$	$X_{6,6}$	...	$X_{6,1208}$
George	$X_{7,1}$	$X_{7,2}$	$X_{7,3}$	$X_{7,4}$	$X_{7,5}$	$X_{7,6}$	...	$X_{7,1208}$
Henry	$X_{8,1}$	$X_{8,2}$	$X_{8,3}$	$X_{8,4}$	$X_{8,5}$	$X_{8,6}$	...	$X_{8,1208}$

The Tardos scheme: Codewords

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

For each i, j, take $X_{j,i} = 1$ with probability p_i.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	0	0	1	1	1	0	...	0
Bob	1	0	1	1	1	0	...	1
Charlie	1	0	0	1	0	1	...	0
David	0	0	1	1	1	0	...	0
Eve	0	0	1	0	1	0	...	0
Fred	1	0	1	0	1	1	...	0
George	0	0	1	0	1	0	...	0
Henry	0	0	0	1	1	1	...	0

The Tardos scheme: Codewords

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The code is complete, and is embedded in the content.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	0	0	1	1	1	0	...	0
Bob	1	0	1	1	1	0	...	1
Charlie	1	0	0	1	0	1	...	0
David	0	0	1	1	1	0	...	0
Eve	0	0	1	0	1	0	...	0
Fred	1	0	1	0	1	1	...	0
George	0	0	1	0	1	0	...	0
Henry	0	0	0	1	1	1	...	0

The Tardos scheme: Coalition

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

Pirates buy their copies, ...

Charlie	1	0	0	1	0	1	...	0
Eve	0	0	1	0	1	0	...	0
Henry	0	0	0	1	1	1	...	0
Copy	y_1	y_2	y_3	y_4	y_5	y_6	...	y_{1208}

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Coalition

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

Pirates buy their copies, compare them ...

Charlie	**1**	0	**0**	**1**	**0**	**1**	...	0
Eve	**0**	0	**1**	**0**	**1**	**0**	...	0
Henry	**0**	0	**0**	**1**	**1**	**1**	...	0
Copy	y_1	y_2	y_3	y_4	y_5	y_6	...	y_{1208}

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Coalition

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

Pirates buy their copies, compare them and generate some $\vec{y}$.

Charlie	**1**	0	**0**	**1**	**0**	**1**	...	0
Eve	**0**	0	**1**	**0**	**1**	**0**	...	0
Henry	**0**	0	**0**	**1**	**1**	**1**	...	0
Copy	**0**	0	**0**	**1**	**1**	**0**	...	0

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Coalition

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The copy is distributed, and the distributor detects $\vec{y}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	0	0	1	1	1	0	...	0
Bob	1	0	1	1	1	0	...	1
Charlie	1	0	0	1	0	1	...	0
David	0	0	1	1	1	0	...	0
Eve	0	0	1	0	1	0	...	0
Fred	1	0	1	0	1	1	...	0
George	0	0	1	0	1	0	...	0
Henry	0	0	0	1	1	1	...	0
Copy	0	0	0	1	1	0	...	0

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

For each j, i, he calculates a score $S_{j,i}$ based on $p_i, X_{j,i}$ and y_i.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	0	0	1	1	1	0	...	0
Bob	1	0	1	1	1	0	...	1
Charlie	1	0	0	1	0	1	...	0
David	0	0	1	1	1	0	...	0
Eve	0	0	1	0	1	0	...	0
Fred	1	0	1	0	1	1	...	0
George	0	0	1	0	1	0	...	0
Henry	0	0	0	1	1	1	...	0
Copy	0	0	0	1	1	0	...	0

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

For each j, i, he calculates a score $S_{j,i}$ based on $p_i, X_{j,i}$ and y_i.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5
Copy	0	0	0	1	1	0	...	0

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(0)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	0
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	0
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	0
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	0
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	0
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	0
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	0
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	0
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(1)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.5
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−2.0
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	−2.0
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.5
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+0.5
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−2.0
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+0.5
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+0.5
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(2)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.7
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−1.8
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	−1.8
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.7
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+0.7
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−1.8
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+0.7
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+0.7
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(3)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.4
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−2.1
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	+1.0
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.4
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+0.4
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−2.1
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+0.4
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+3.5
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(4)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.9
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−1.6
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	+1.5
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+0.9
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−1.6
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−4.1
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−1.6
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+4.0
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(5)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+1.0
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−1.5
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	−5.7
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+1.0
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−1.4
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−3.9
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−1.4
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+4.1
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(6)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+1.3
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−1.2
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	−9.0
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+1.3
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−1.1
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−7.2
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−1.1
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+0.8
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Scores

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

The user score $S_j(\ell)$ is then calculated by adding up the $S_{j,i}$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(1208)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+14
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−19
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	+291
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+29
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+292
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−53
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−42
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+269
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Accusation

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

Finally the distributor accuses all users j with $S_j(\ell) > Z$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(1208)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+14
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−19
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	+291
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+29
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+292
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−53
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−42
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+269
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie}, \text{Eve}, \text{Henry}\}$$

The Tardos scheme: Accusation

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.

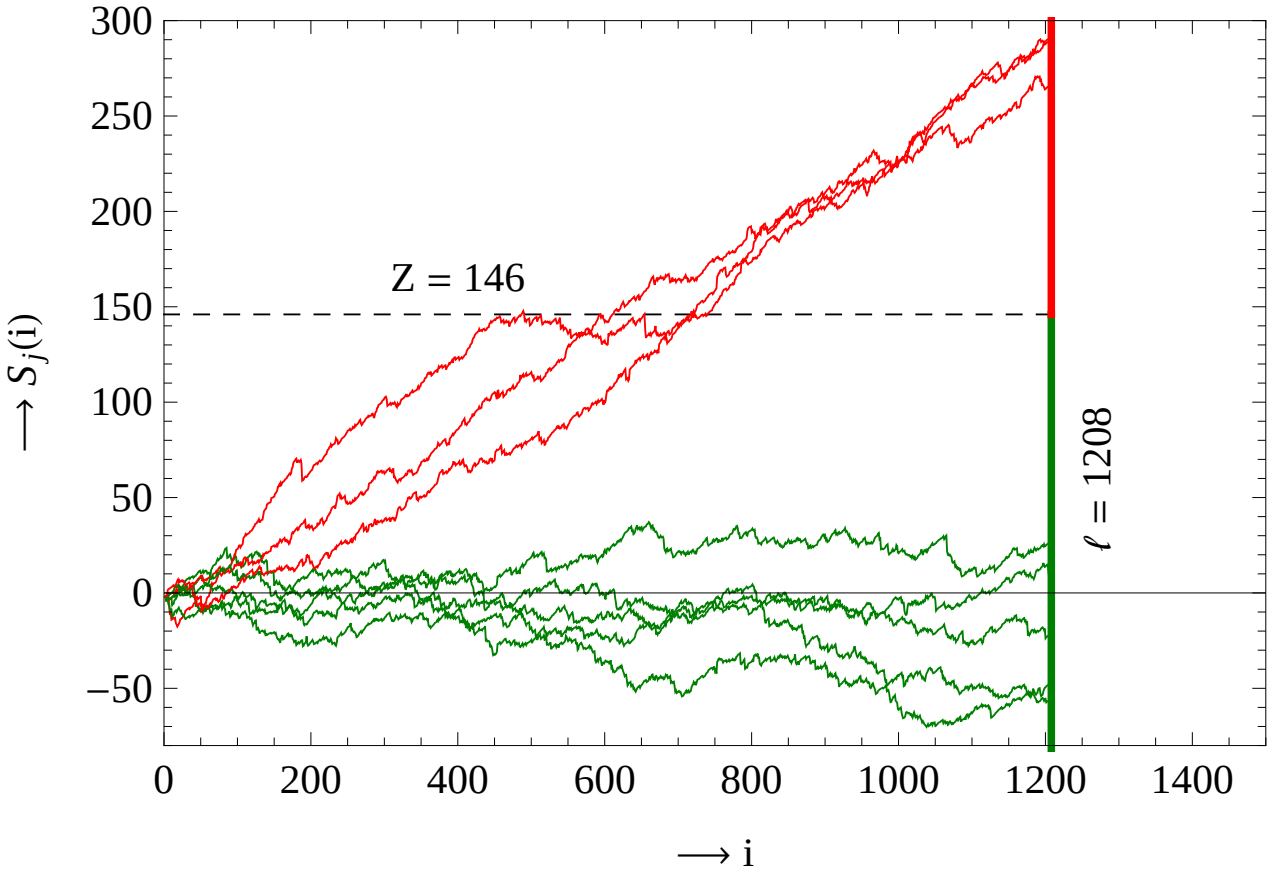
Finally the distributor accuses all users j with $S_j(\ell) > Z$.

p_i	0.20	0.05	0.88	0.79	0.98	0.09	...	0.18	$S_j(1208)$
Alice	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+14
Bob	−2.0	+0.2	−0.4	+0.5	+0.1	+0.3	...	−2.1	−19
Charlie	−2.0	+0.2	+2.7	+0.5	−7.2	−3.3	...	+0.5	+291
David	+0.5	+0.2	−0.4	+0.5	+0.1	+0.3	...	+0.5	+29
Eve	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	+292
Fred	−2.0	+0.2	−0.4	−1.9	+0.1	−3.3	...	+0.5	−53
George	+0.5	+0.2	−0.4	−1.9	+0.1	+0.3	...	+0.5	−42
Henry	+0.5	+0.2	+2.7	+0.5	+0.1	−3.3	...	+0.5	+269
Copy	0	0	0	1	1	0	...	0	

$$C = \{\text{Charlie, Eve, Henry}\}$$
$$\hat{C} = \{\text{Charlie, Eve, Henry}\}$$

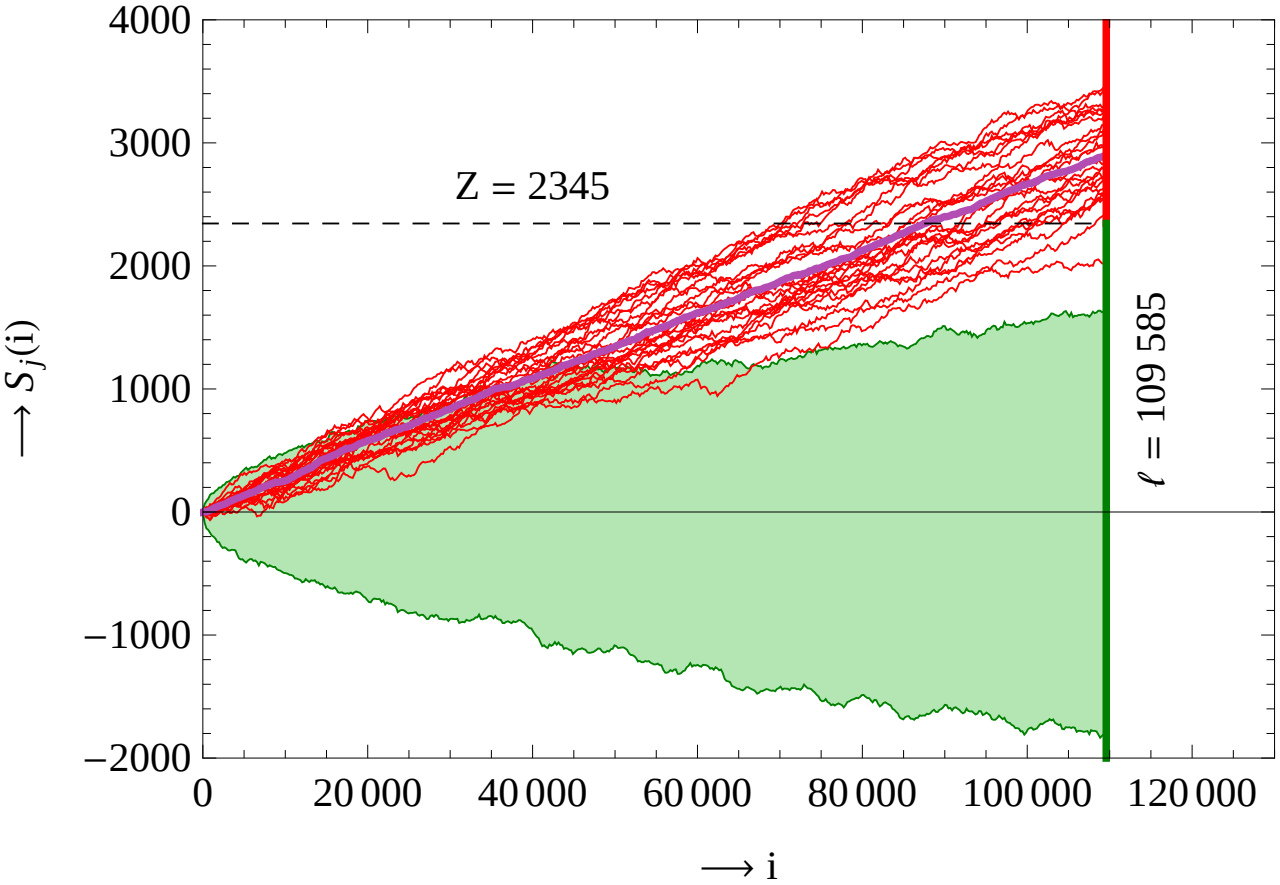
The Tardos scheme: Accusation

Let $n = 8$, $c = 3$, $\epsilon_{1,2} = 0.01$, $\ell = 1208$, $Z = 146$, $\delta = 0.0115$, $p_i \in [\delta, 1 - \delta]$.



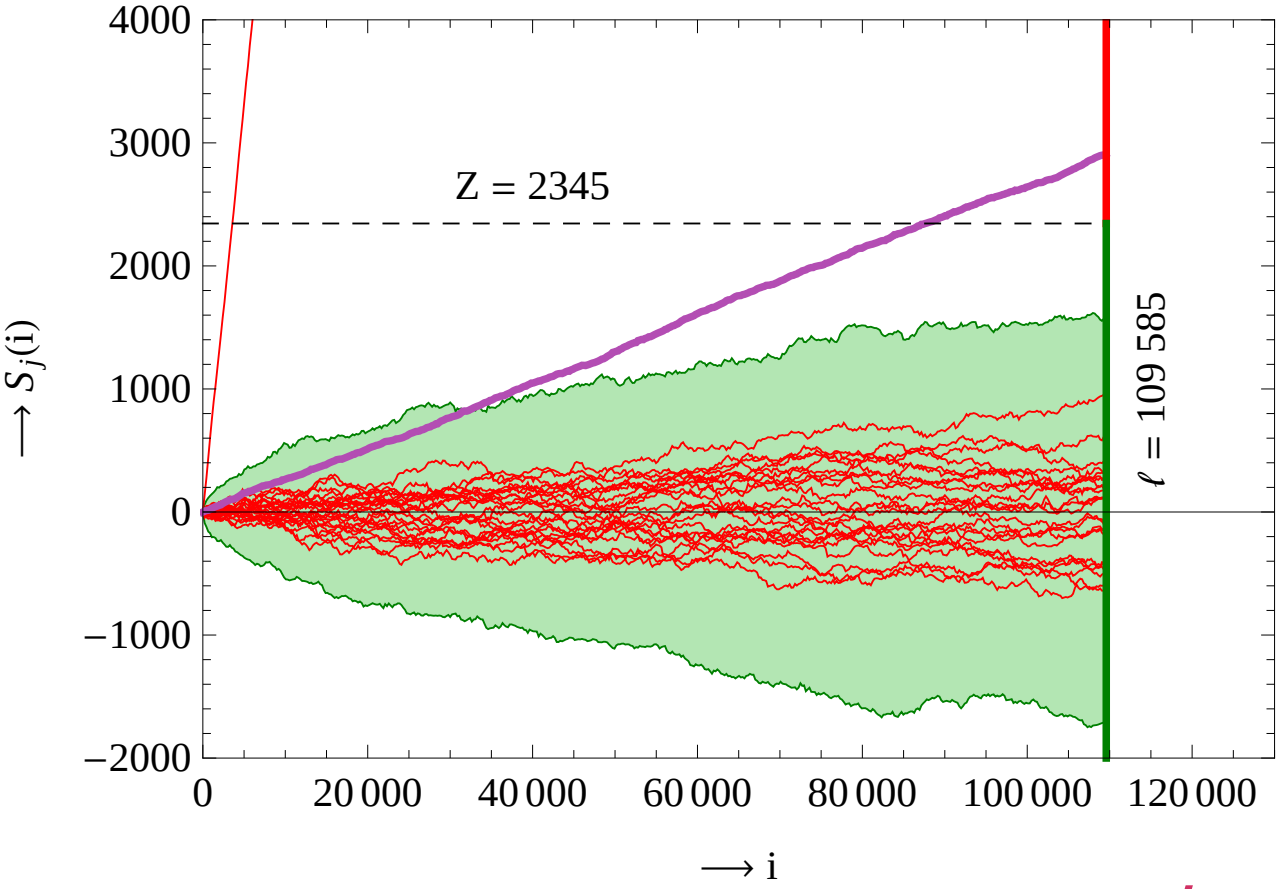
The Tardos scheme: Example

Example: $n = 10^6$ users, $c = 25$ colluders, error probabilities $\epsilon_{1,2} = 0.001$.



The Tardos scheme: Example

Example: $n = 10^6$ users, $c = 25$ colluders, error probabilities $\epsilon_{1,2} = 0.001$.



The Tardos scheme: Details

Why are no innocent users accused?

- Random walk behaviour: $E(S_j(\ell)) = 0$, $\mathrm{Var}(S_j(\ell)) = O(Z^2)$.
- Proof: $P(S_j(\ell) > Z) \leq \epsilon_1/n$ for innocent j.

Why is at least one guilty user accused?

- The average pirate score $S_C(i) = \frac{1}{c}\Sigma_{j\in C}\, S_j(i) \ldots$

  ... increases on undetectable positions.

  ... approximately remains the same on detectable positions.
- Random walk behaviour: $E(S_C(\ell)) = \Omega(Z)$ and $\mathrm{Var}(S_C(\ell)) = O(1)$.
- Proof: $P(S_C(\ell) \leq Z) \leq \epsilon_2$.

The Tardos scheme: Summary

Comparison with other static schemes: **Short codelengths**

- Optimal order codelength: $\ell = O(c^2 \ln(n/\epsilon_1))$.
- Asymptotic codelength: $\ell \approx 4.93c^2 \ln(n/\epsilon_1)$.
  (Theoretical lower bound: $\ell \approx 1.39c^2 \ln(n/\epsilon_1)$.)

Comparison with dynamic schemes: **Only catch one colluder**

- Short codelengths, small alphabet size.
- Simple codeword generation, accusation algorithm.
- Problem: No guarantee of catching multiple colluders.
- Problem: Need to know c in advance.

The Tardos scheme: Summary

Comparison with other static schemes: **Short codelengths**

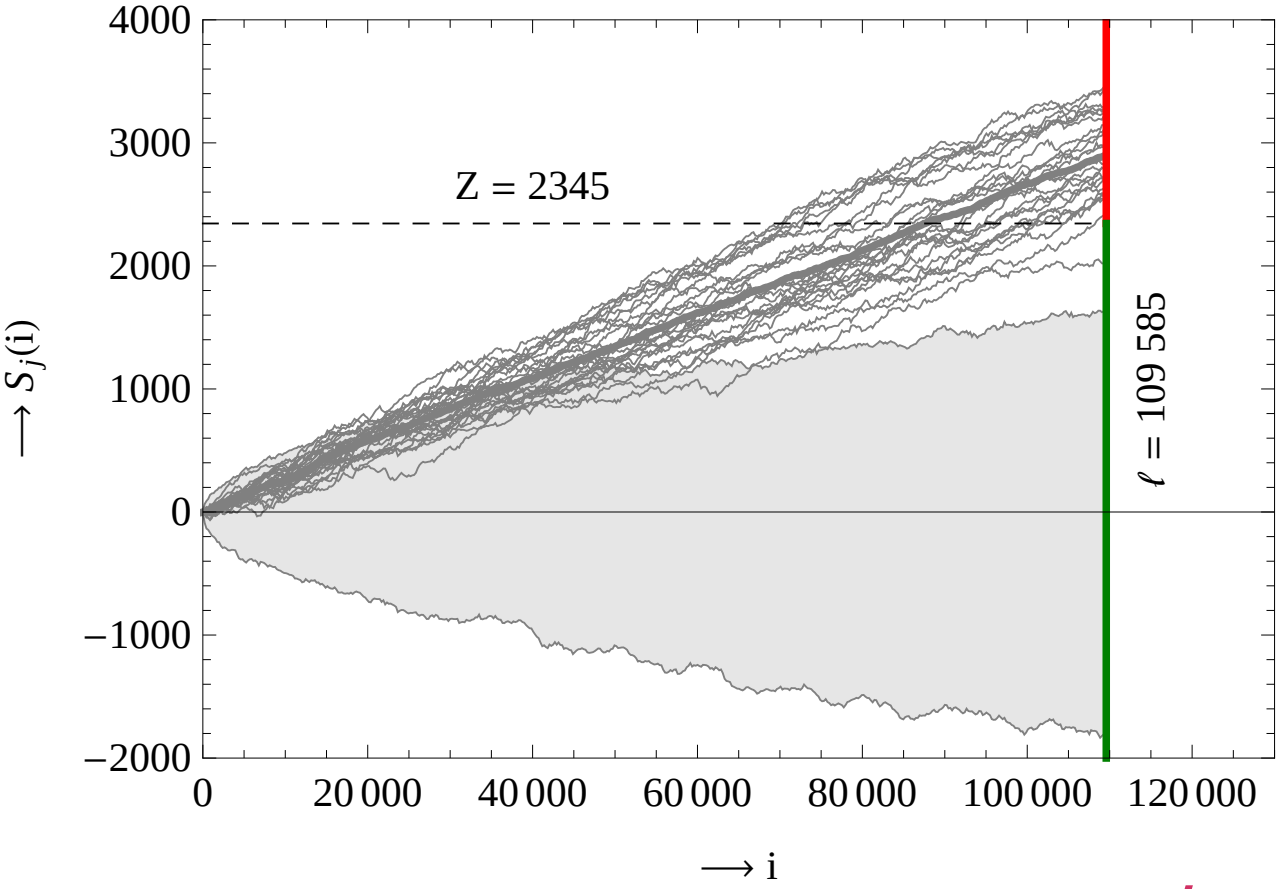
- Optimal order codelength: $\ell = O(c^2 \ln(n/\epsilon_1))$.
- Asymptotic codelength: $\ell \approx 4.93c^2 \ln(n/\epsilon_1)$.
  (Theoretical lower bound: $\ell \approx 1.39c^2 \ln(n/\epsilon_1)$.)

Comparison with dynamic schemes: **Only catch one colluder**

- Short codelengths, small alphabet size.
- Simple codeword generation, accusation algorithm.
- **Problem: Never guarantee of catching multiple colluders.**
- Problem: Need to know c in advance.

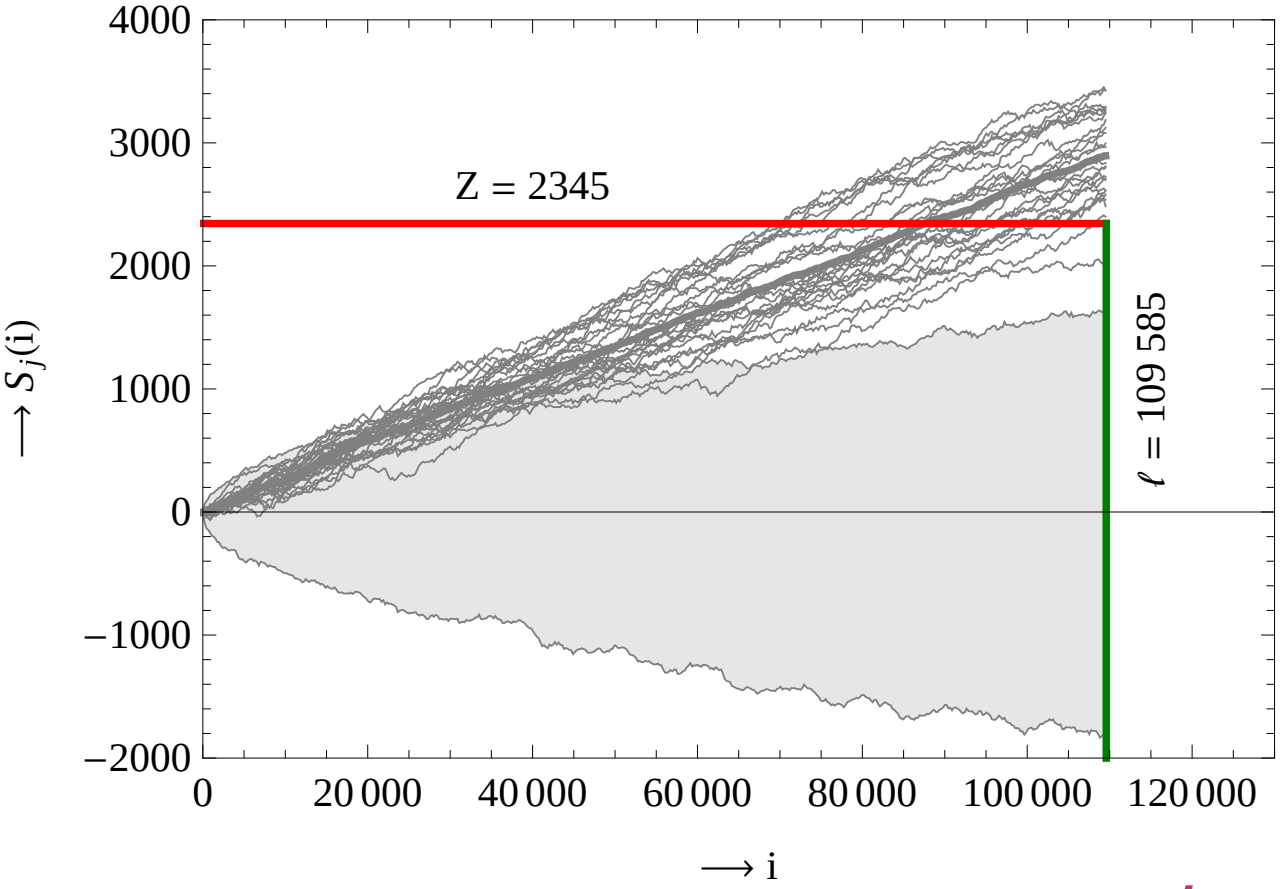
The dynamic Tardos scheme: Intro

Static Tardos scheme: At time ℓ, accuse all users with $S_j(\ell) > Z$.



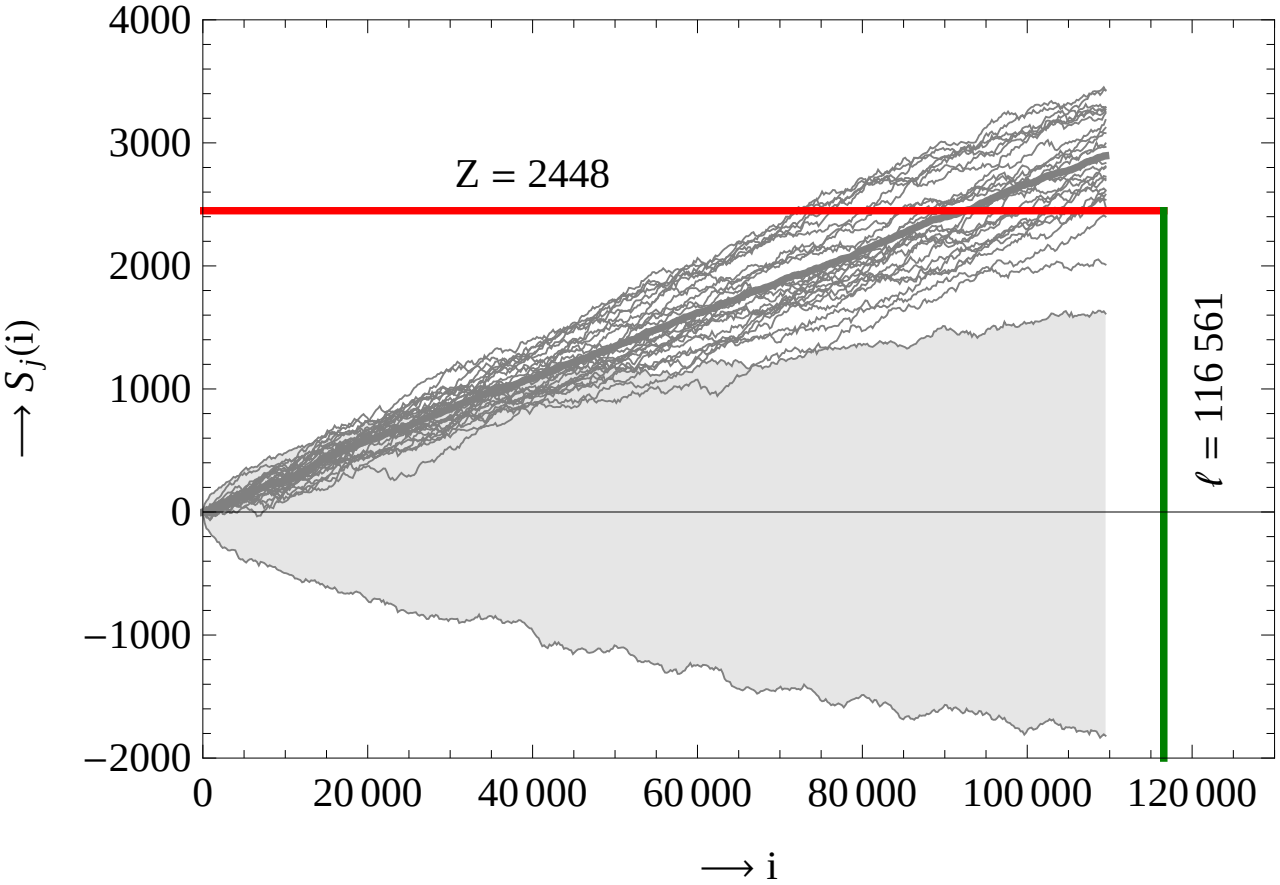
The dynamic Tardos scheme: Intro

Dynamic Tardos scheme: At each time i, disconnect users with $S_j(i) > Z$.



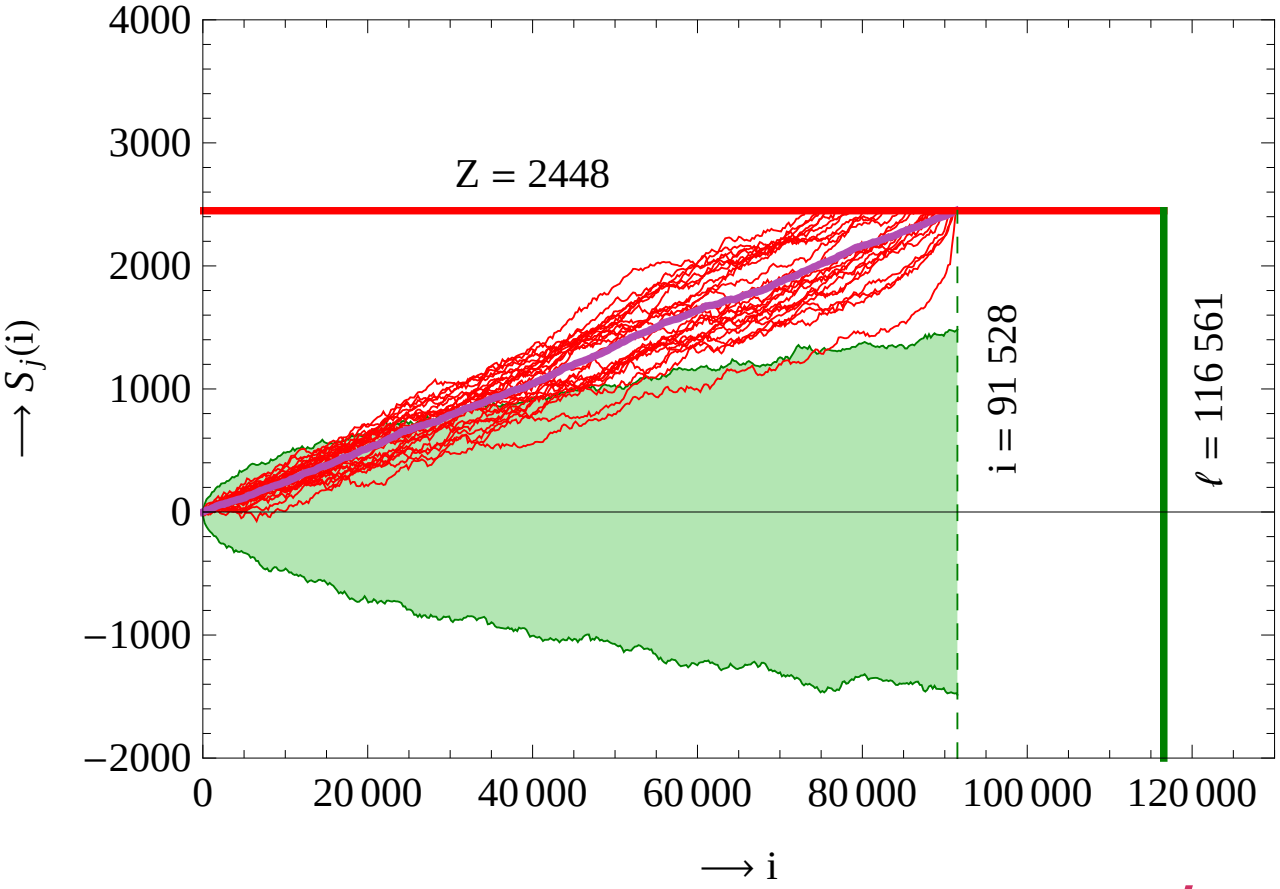
The dynamic Tardos scheme: Intro

Slightly longer theoretical codelengths, but now catch *all* pirates!



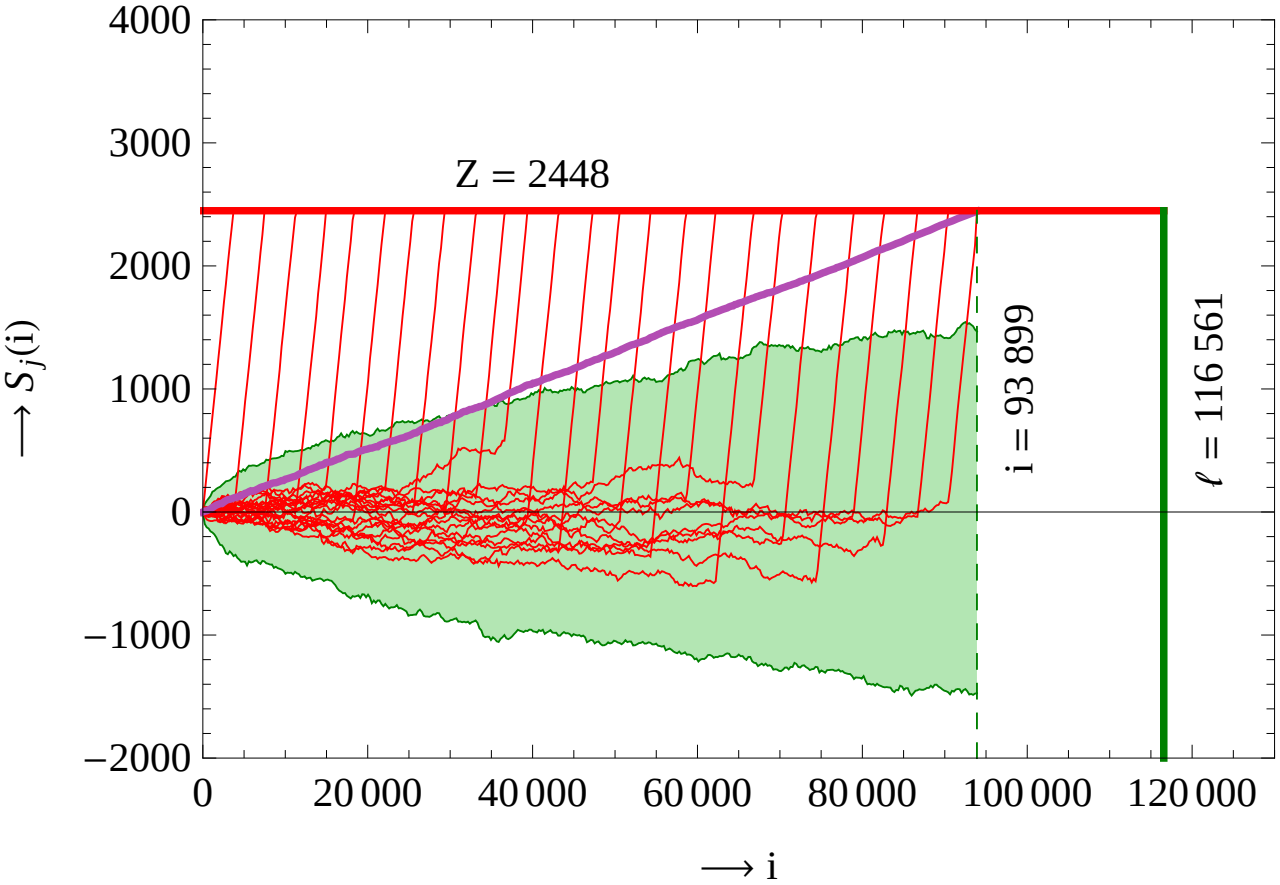
The dynamic Tardos scheme: Example

Example: $n = 10^6$ users, $c = 25$ colluders, error probabilities $\epsilon_{1,2} = 0.001$.



The dynamic Tardos scheme: Example

Example: $n = 10^6$ users, $c = 25$ colluders, error probabilities $\epsilon_{1,2} = 0.001$.



The dynamic Tardos scheme: Details

Why are no innocent users accused?

- Random walk behaviour: $E(S_j(\ell)) = 0$ and $\mathrm{Var}(S_j(\ell)) = O(Z^2)$.
- Random walk behaviour: $P(\exists\, i : S_j(i) > Z) \leq 2 \cdot P(S_j(\ell) > Z)$
- But Tardos proved $P(S_j(\ell) > Z) \leq \epsilon_1/n$.

Why are *all* guilty users accused?

- Random walk behaviour: $E(S_C(\ell)) = \Omega(Z)$ and $\mathrm{Var}(S_C(\ell)) = O(1)$.
- If not all pirates are caught, then $S_C(\ell) \lesssim Z$.
- But Tardos proved $P(S_C(\ell) \leq Z) \leq \epsilon_2$.

Problem: Need to know c in advance.

The dynamic Tardos scheme: Details

Why are no innocent users accused?

- Random walk behaviour: $E(S_j(\ell)) = 0$ and $\mathrm{Var}(S_j(\ell)) = O(Z^2)$.
- Random walk behaviour: $P(\exists\, i : S_j(i) > Z) \leq 2 \cdot P(S_j(\ell) > Z)$
- But Tardos proved $P(S_j(\ell) > Z) \leq \epsilon_1/n$.

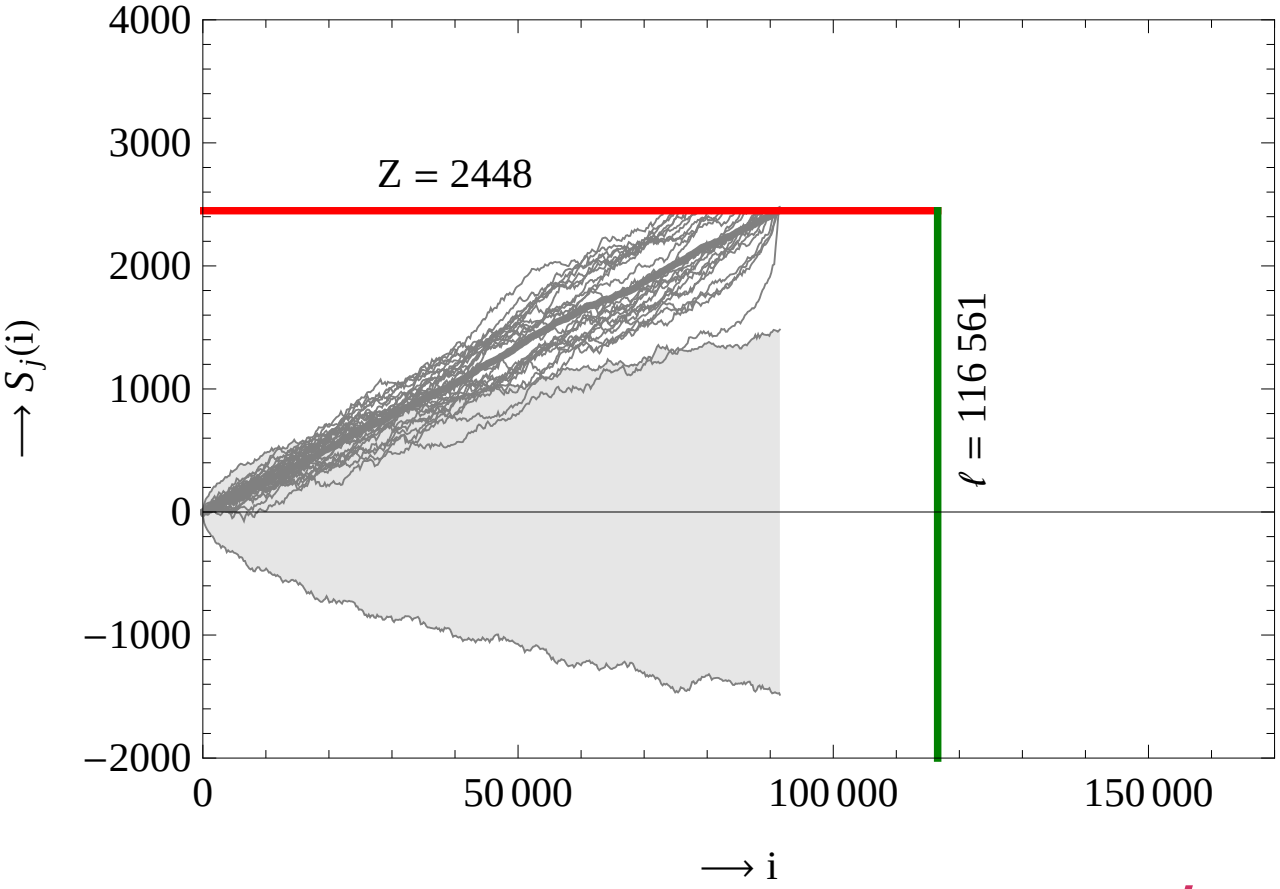
Why are *all* guilty users accused?

- Random walk behaviour: $E(S_C(\ell)) = \Omega(Z)$ and $\mathrm{Var}(S_C(\ell)) = O(1)$.
- If not all pirates are caught, then $S_C(\ell) \lesssim Z$.
- But Tardos proved $P(S_C(\ell) \leq Z) \leq \epsilon_2$.

Problem: Need to know c in advance.

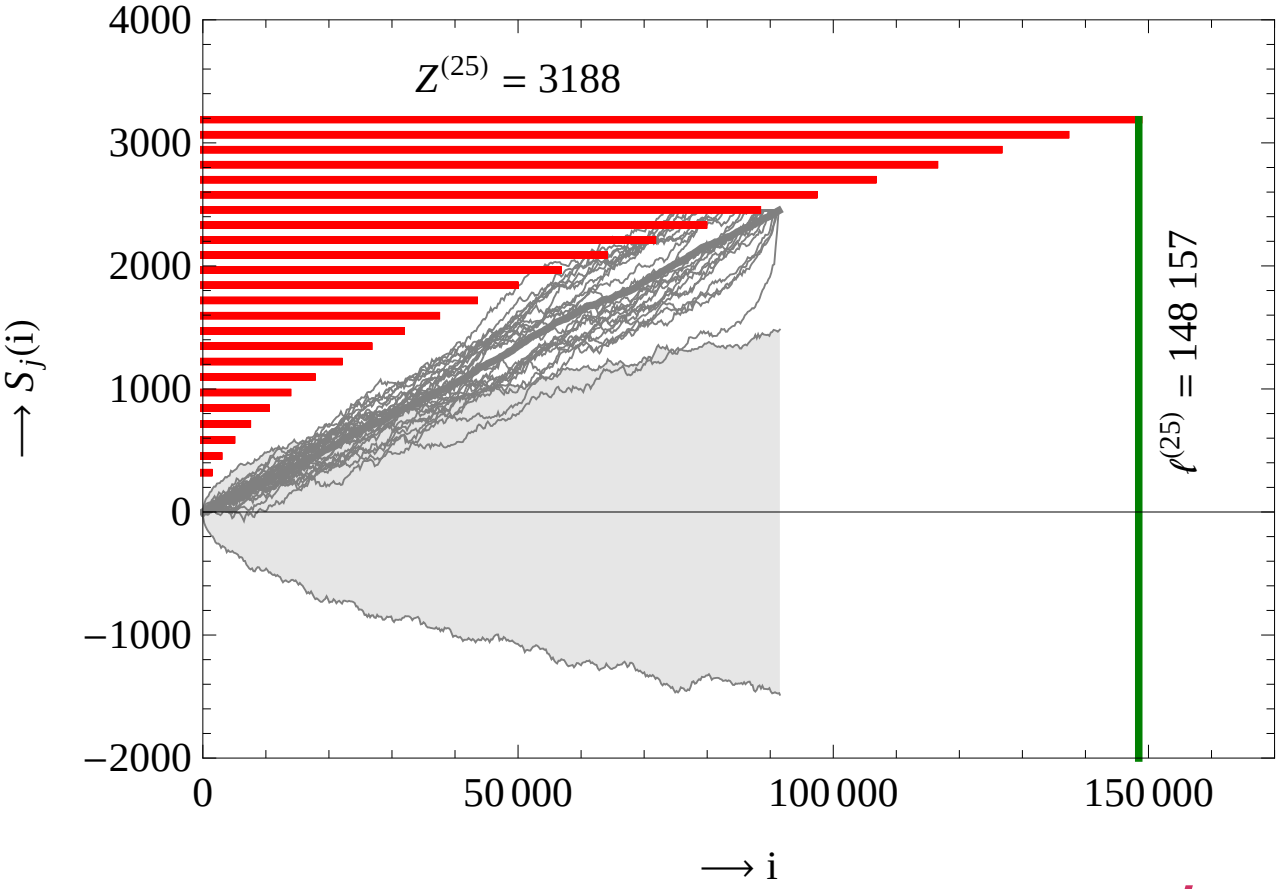
The universal Tardos scheme: Intro

Dynamic Tardos scheme: One code X, one set of parameters ℓ, Z, δ.



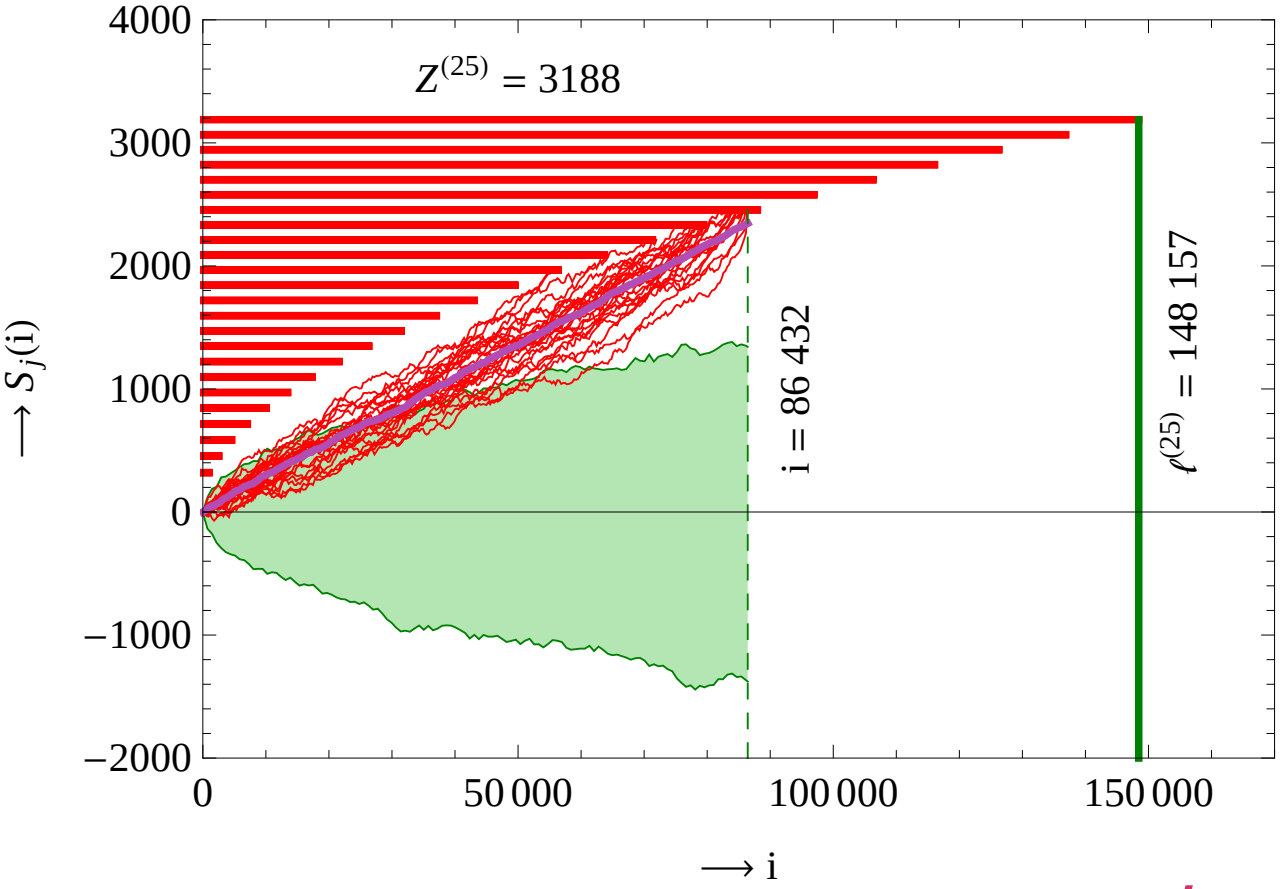
The universal Tardos scheme: Intro

Universal Tardos scheme: One code X, but parameters ℓ, Z, δ for *each* c.



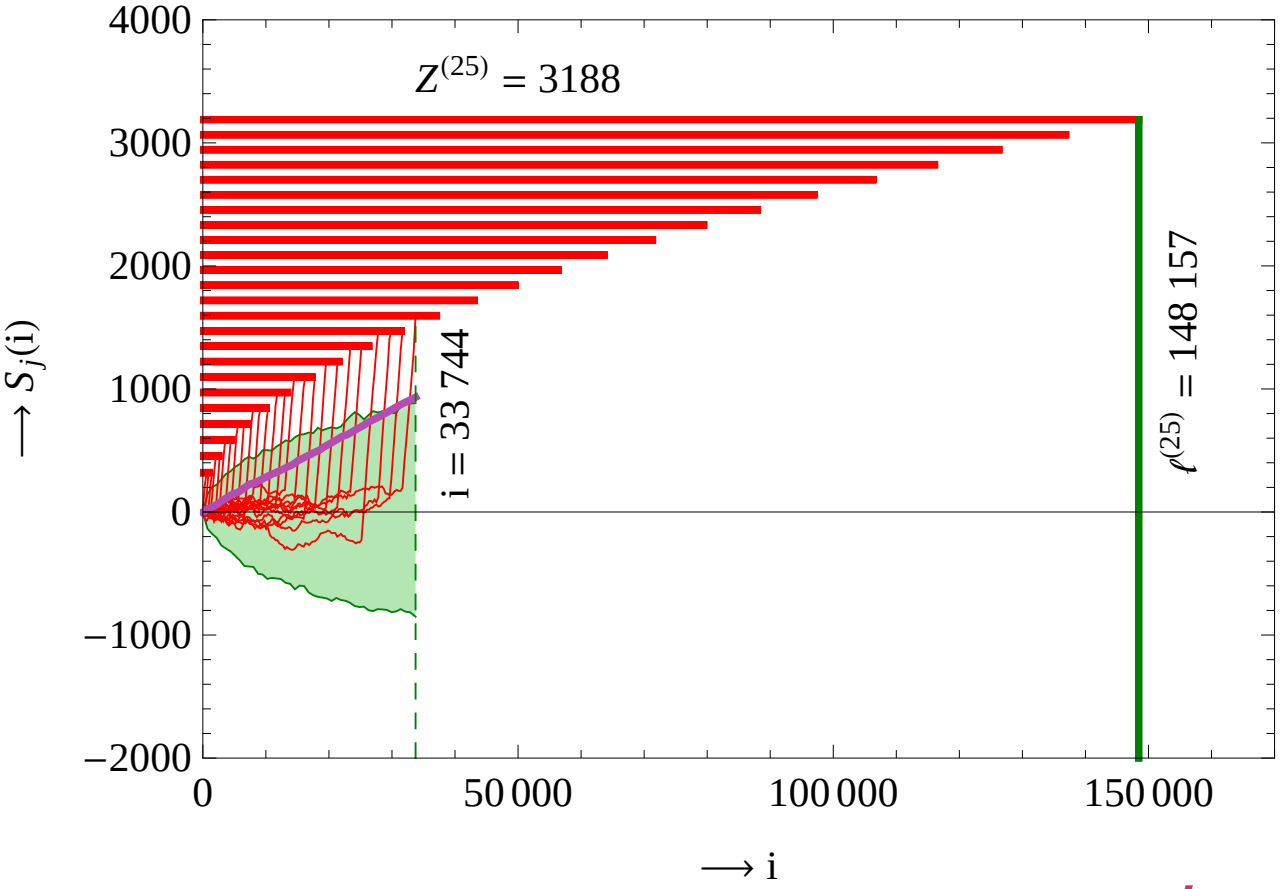
The universal Tardos scheme: Example

Example: $n = 10^6$ users, $c = 25$ colluders, error probabilities $\epsilon_{1,2} = 0.001$.



The universal Tardos scheme: Example

Example: $n = 10^6$ users, $c = 25$ colluders, error probabilities $\epsilon_{1,2} = 0.001$.



The universal Tardos scheme: Details

Why are no innocent users accused?

- Divide ϵ_1 over all values of c, e.g. $\epsilon_1^{(c)} = O(\epsilon_1/c^2)$.
- Total error bounded by $\sum \epsilon_1^{(c)} \leq \epsilon_1$.

Why are all guilty users accused?

- Probability of catching pirates increases compared to dynamic Tardos.

The universal Tardos scheme: Summary

Comparison with static Tardos scheme: **Catch all pirates**

- Catch all members of any coalition efficiently.
- Same order codelengths: $\ell = O(c^2 \ln(n/\epsilon_1))$.
- Same asymptotic codelength: $\ell \approx 4.93c^2 \ln(n/\epsilon_1)$.

Comparison with dynamic schemes: **Short codelengths**

- Simple codeword generation, accusation algorithm.
- Short codelengths, small alphabet size.
- In all aspects better than Tassa's scheme.

Questions

Thank you for your attention! Any questions?