Practical Traffic Analysis:

Extending and resisting statistical disclosure

by Nick Mathewson and Roger Dingledine
The Free Haven Project
{nickm,arma}@freehaven.net

May 26, 2004

PET2004

Summary

We extend earlier work on
end-to-end traffic analysis attacks
against
high-latency anonymity networks.

We simulate these attacks, and note some cases in which they may be impractical.

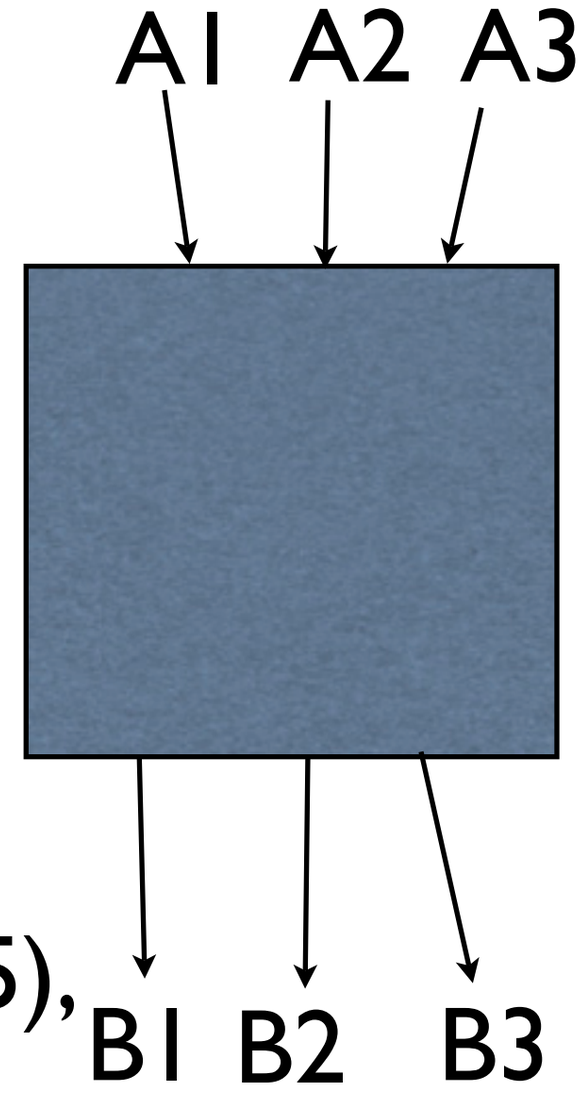
We close with recommendations.

Anonymity Networks (what are we attacking?)

- Many senders ("Alice"), many recipients ("Bob")
- Alice wants to hide Alice/recipient connection
- ... from recipients
- ... from attackers (active and passive)
- ... from the infrastructure itself

Anonymity Networks (how do they work?)

- Receive encrypted messages
- Decrypt, learn next hop
- Delay to hide timing correlations
  (High-latency systems only!)
- Deliver towards recipient



Ex: Mix-nets (1981), Mixmaster (1995),
Babel (1996), Mixminion (2003)

Attack Category: Long-term Intersection

The Goal:

- Link targeted senders to their recipients

The Attack:

- Alice has a set of regular recipients
- When Alice has sent a message, those recipients are likelier to receive
- So, *watch for a long time*, and see who receives more when Alice has been sending

Previous work: The Disclosure Attack

(Kesdogan, Agrawal, and Penz, 2002)

- Batch mix (get *b* messages, then relay)
- NP-complete computation
- Identifies Alice's recipients with certainty

Previous work: Statistical Disclosure

(Danezis, 2003)

- Easier to implement
- Statistical: Identifies *probable* recipients
- Method: Compute mean recipient distribution when Alice is sending; compare to (known) background distribution

Our contribution

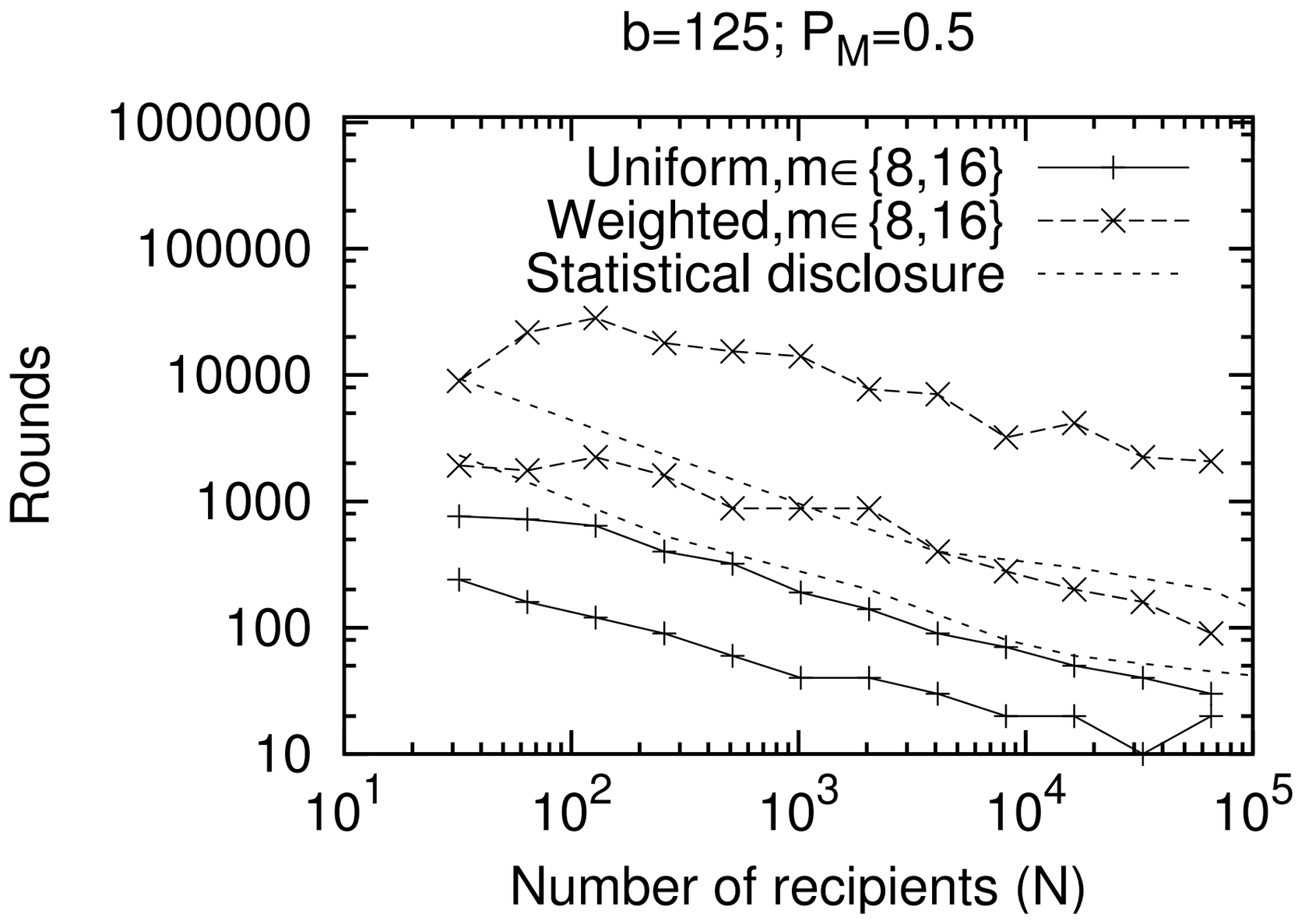
- Strengthen attack to work against better networks:
  - Unknown background distribution
  - Complex sender behavior
  - Pool mixes and mix-nets
  - Padding (“dummy”) messages
  - Non-global attacker
- (Also, ways to exploit additional info)

Simulation Model

- Scale-free network of recipients
- Alice sends with geometric distribution
- Background sends with normal distribution
- Global attacker
- No other linkable info in messages
- *Static, steady-state network*

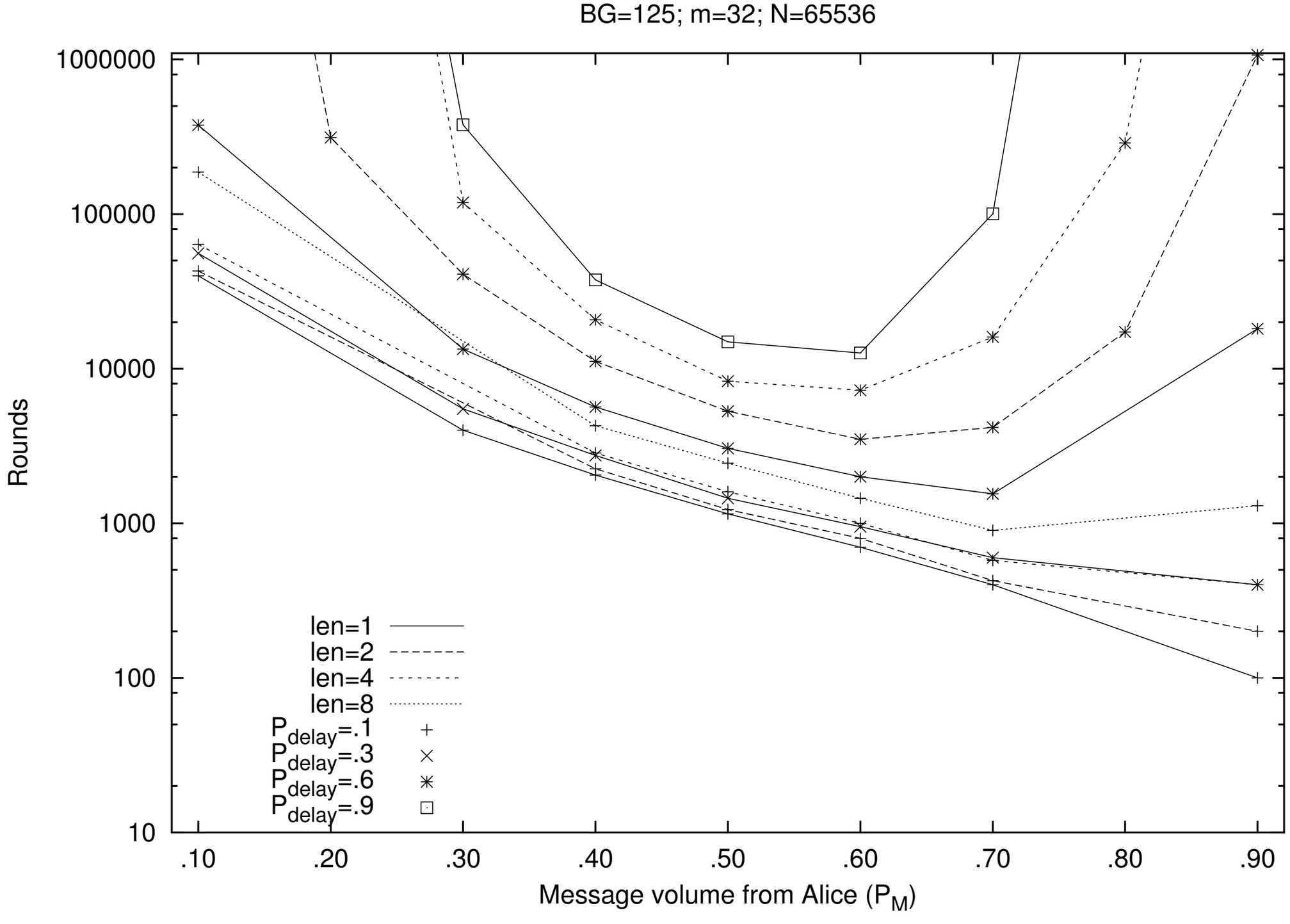
Unknown background

Method: estimate background by averaging rounds in which Alice is *not* sending.



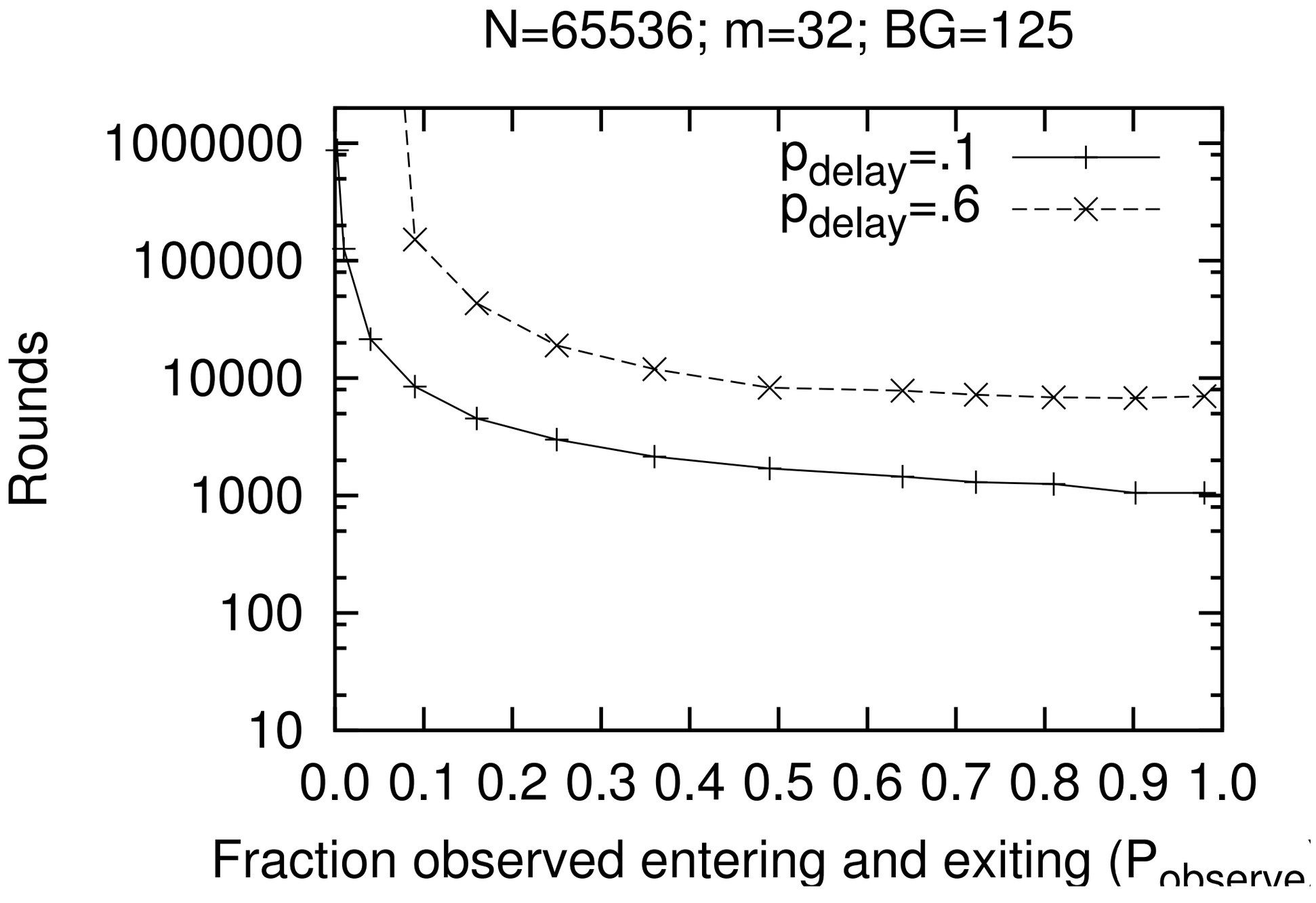
Pool mixes and mix-nets

Method: compute *expected* contribution of each message to subsequent rounds



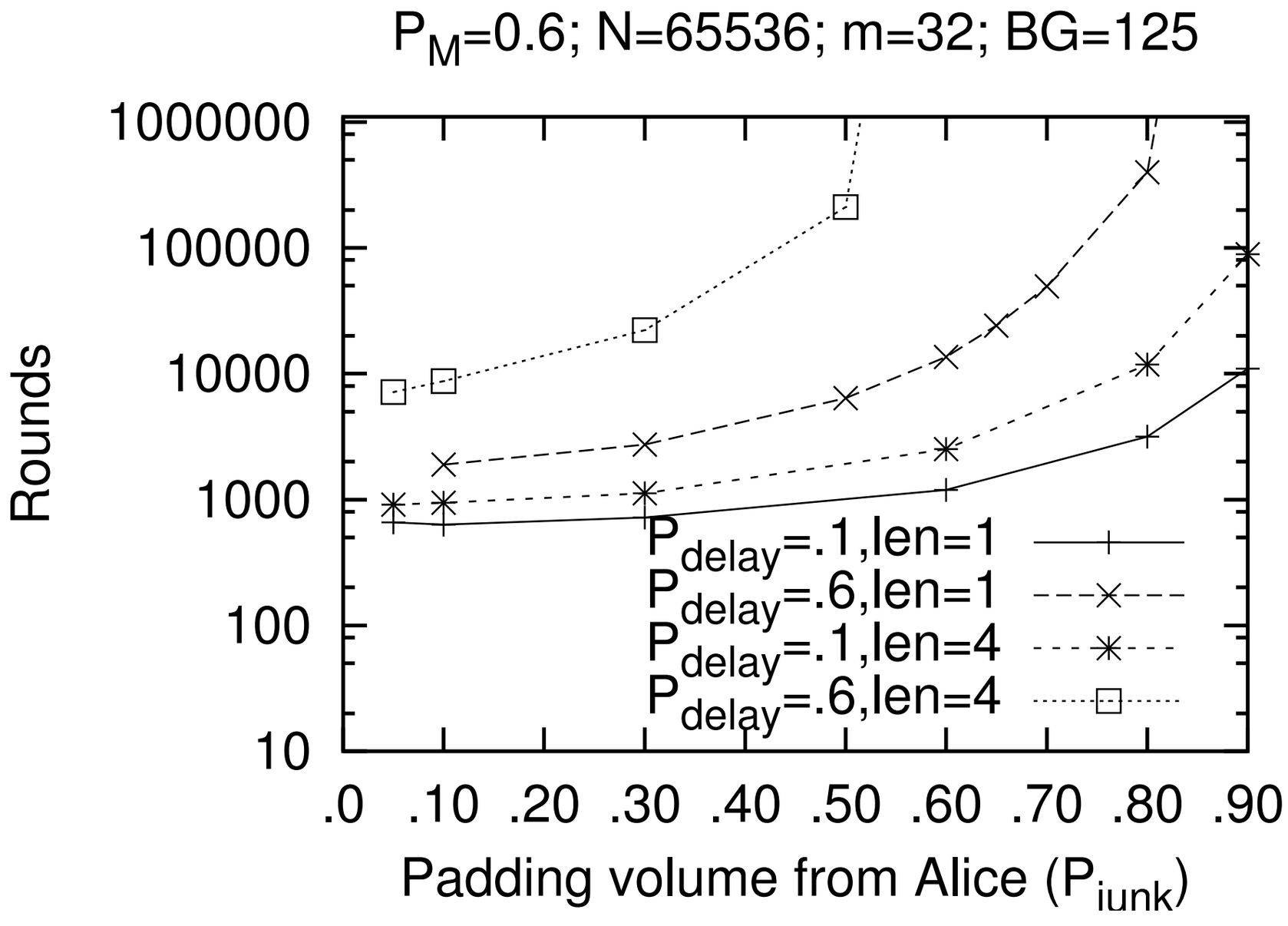
Non-global attackers

Method: Sample!



Independent Padding

No changes needed -- it's just more noise

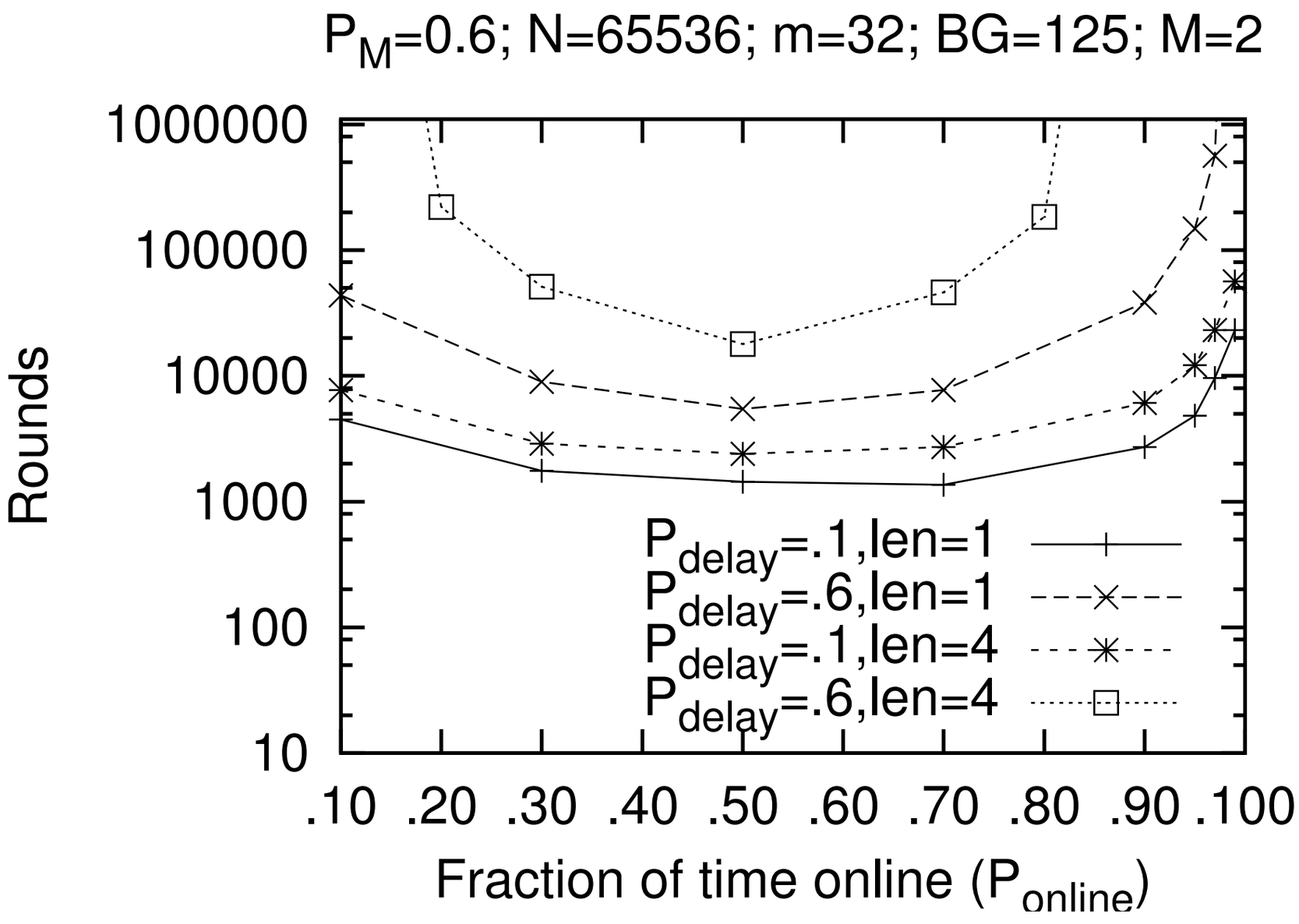


Perfect threshold padding

Alice wins.

But if Alice is unreliable...

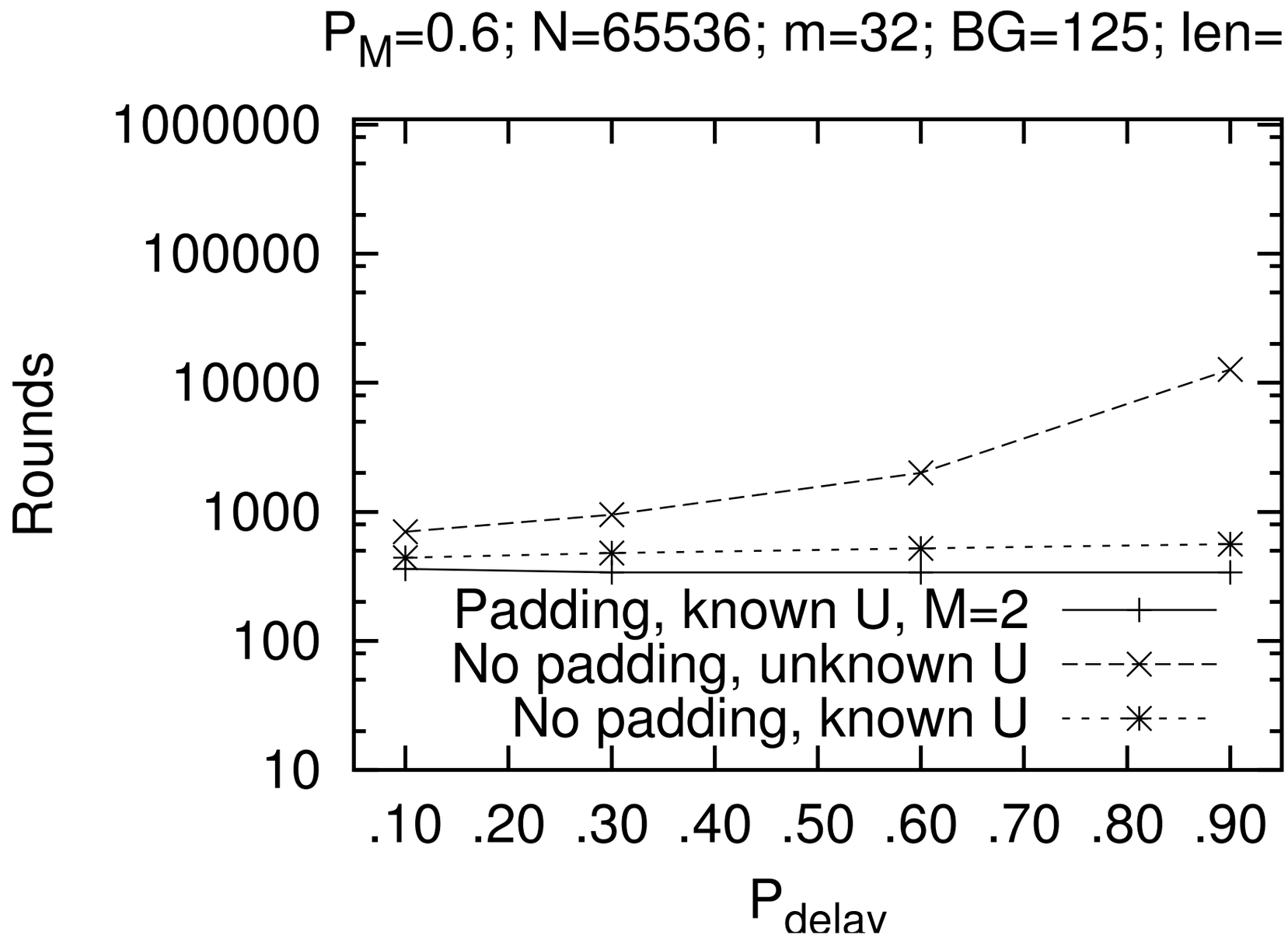
If Alice is sometimes offline, threshold padding can fail.



An active attacker can make this happen!

And if Alice must join/leave...

Threshold padding still doesn't help at all.



Other scenarios
(not simulated)

- Slowly changing cover traffic
- Attacks against recipients
- Exploiting message linkability
  - Pseudonyms
  - Message properties

Lessons (1)

- Intersection attacks may be feasible; being almost-global isn't necessary.

- Don't ask: "Is it categorically secure?"
  Ask: "How long does it secure whom?"

- Senders:
  Don't participate longer than necessary.

Lessons (2)

- It's hard to get padding perfect...
  ...and the imperfections matter.
  ...but padding can still help.
- High message delay variance is *essential*
  (It makes padding more effective and partial observation less effective.)

Model Limitations

In Alice's favor:

- User behavior changes over time.
- What if Alice runs a mix?

In attacker's favor:

- User behavior is not geometric, not quite scale-free-network.
  (Diaz, Sassaman, and Dewitte, [TR, submitted])
- Messages may be linkable.
- Attacker might be active.

Future work

- Better models for users
- Strengthen attacks
  (active attackers; linkable messages)
- Do “lessons” change when other attacks are considered?
- Closed-form solutions where possible.
- Link to other models of anonymity?
- Self-optimizing mix networks?

Q&A ?

- Simulation code available at http://freehaven.net/doc/e2e-traffic/